Xie et al. *Journal of Inequalities and Applications* (2021) 2021:132
https://doi.org/10.1186/s13660-021-02671-2

Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

A new semismooth Newton method for solving finite-dimensional quasi-variational inequalities

Shui-Lian Xie[1], Zhe Sun[2*] and Hong-Ru Xu[1]

*Correspondence: snzma@126.com
[2]College of Mathematics and Information Science, Jiangxi Normal University, Nanchang, China
Full list of author information is available at the end of the article

Abstract

In this paper, we consider the numerical method for solving finite-dimensional quasi-variational inequalities with both equality and inequality constraints. Firstly, we present a semismooth equation reformulation to the KKT system of a finite-dimensional quasi-variational inequality. Then we propose a semismooth Newton method to solve the equations and establish its global convergence. Finally, we report some numerical results to show the efficiency of the proposed method. Our method can obtain the solution to some problems that cannot be solved by the method proposed in (Facchinei et al. in Comput. Optim. Appl. 62:85–109, 2015). Besides, our method outperforms than the interior point method proposed in (Facchinei et al. in Math. Program. 144:369–412, 2014).

MSC: 65K10; 90C53; 90C33

Keywords: Quasi-variational inequality; KKT condition; Semismooth Newton method

1 Introduction

We consider the finite-dimensional quasi-variational inequality QVI(K, F): Find a vector $x^* \in K(x^*)$ such that

$$F(x^*)^T(y - x^*) \geq 0, \quad \forall y \in K(x^*), \tag{1.1}$$

where $F : R^n \to R^n$ is a point to point mapping and $K : R^n \rightrightarrows R^n$ is a point to set mapping with closed and convex images. Throughout the paper, we assume that F belongs to C^1 and for each $x \in R^n$, the feasible set mapping K is given by

$$K(x) \triangleq \{y \in R^n \mid g(y, x) \leq 0, h(y, x) = 0\}, \tag{1.2}$$

where $g : R^n \times R^n \to R^{m_1}$ belongs to C^2 and $g_i(\cdot, x)$ is convex on R^n for each $i = 1, 2, \ldots, m_1$ and for all $x \in R^n$, $h : R^n \times R^n \to R^{m_2}$ belongs to C^2 and $h_j(\cdot, x)$ is affine on R^n for each $j = 1, 2, \ldots, m_2$ and for all $x \in R^n$. When the set $K(x)$ is independent of x, (1.1) reduces to the famous variational inequality (VI). For VI, we refer the reader to [13] and the references therein.

QVI (1.1), which was first introduced by Bensoussan and Lions [2, 3], has important applications in many fields such as generalized Nash games, mechanics, economics, statistics, transportation and biology; see for example [1, 6, 10, 12] and the references therein. One interesting topic on QVI is to develop the efficient algorithms for the solution of QVI. Since QVI is nonsmooth and nonconvex, it is difficult to design effective methods for QVI, and by now, compared with VI, the numerical methods are still scarce. In this paper, we mainly focus on the numerical method based on the KKT conditions of QVI. This area attracts many people's attention and much progress has been made. In [12] an interior point approach was proposed to solve QVI and the convergence was established for several classes of interesting QVIs. Reference [8, 9] developed a so called LP-Newton method and the method can be successfully applied to nonsmooth systems of equations with non-isolated solutions. Reference [21] developed an efficient regularized smoothing Newton-type algorithm for QVI. The proposed algorithm takes the advantage of newly introduced smoothing functions and a non-monotone line search strategy. [10] proposed a semismooth Newton method for QVI. They obtained global convergence and locally superlinear/quadratic convergence result for some important classes of quasi-variational inequality problems. The numerical results show that the method performs well.

There are many ways to compute a numerical solution of the nonlinear complementarity problems (NCP), such as linearized projected relaxation methods [13], the modulus-based matrix splitting method [24] and the penalty method [7, 23, 25]. In the past two decades, the nonsmooth-equation-based method has been thoroughly studied to solve NCP; see for example [5, 14–19] and the references therein. A common way to reformulate the complementarity system is to use the so called NCP-function. A function $\phi : R^2 \to R$ is called an NCP-function if it satisfies

$$\phi(a,b) = 0 \quad \Leftrightarrow \quad a \geq 0, \quad b \geq 0, \quad ab = 0.$$

For example, the famous Fischer–Burmeister (FB) function takes the form

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

By the use of the NCP-function, nonlinear complementarity problem can be easily converted into a system of nonlinear equations. Most existing NCP-functions are generally nondifferentiable in the sense of Fréderivative but semismooth in the sense of Mifflin [20] and Qi and Sun [22]. In [17], the authors proposed a nonsmooth equation reformulation to the NCP. Their reformulation enjoys a nice property that it is continuous differentiable everywhere except at the solution. In this paper, we present a semismooth equation reformulation to the KKT system of a quasi-variational inequality and propose a semismooth Newton method to solve the equations.

The paper is organized as follows. In the next section, we describe a semismooth equation reformulation to the KKT system of a quasi-variational inequality, present the semismooth Newton method and establish the global convergence for the method. In Sect. 3, we compare the proposed method with some other methods on problems list in [11].

In the following, we introduce some notations that will be used in this paper. For a continuously differentiable function $F : R^n \to R^n$, we write $JF(x)$ for the Jacobian of F at a point $x \in R^n$, whereas $\nabla F(x)$ denotes the transposed Jacobian of F. Given a smooth mapping $g : R^n \times R^n \to R^m$, $(y, x) \mapsto g(y, x)$, $\nabla_y g(y, x)$ denotes the transpose of the partial Jacobian of g with respect to the y-variables. If F is locally Lipschitz continuous around x, then $\partial F(x)$ denotes Clarke's generalized Jacobian of F at x. For a vector $x \in R^n$ and a subset $I \subset \{1, 2, \dots, n\}$, we write x_I for the subvector consisting of the elements x_i, $i \in I$. For a matrix $A \in R^{n\times n}$ and two subsets $I, J \subset \{1, 2, \dots, n\}$, the symbol A_{IJ} stands for the submatrix with entries a_{ij} for $i \in I$, $j \in J$. The symbol $\operatorname{diag}(a_{11}, a_{22}, \dots, a_{nn})$ stands for a diagonal matrix with diagonal elements $a_{11}, a_{22}, \dots, a_{nn}$.

2 Semismooth equation reformulation and semismooth Newton method

Firstly, we give the following definition that will be used.

Definition 2.1 ([22]) A function $F : R^n \to R^n$ is semismooth at a point $x \in R^n$ if it is locally Lipschitzian at x and

$$\lim_{V\in\partial F(x+td'),d'\to d,t\downarrow 0} Vd'$$

exists for any $d \in R^n$, where $\partial F(x)$ is the generalized Jacobian of F at x. F is strongly semismooth at $x \in R^n$ if for any $d \to \mathbf{0}$ and any $V \in \partial F(x + d)$,

$$Vd - F'(x; d) = O\bigl(\|d\|^2\bigr),$$

where $F'(x; d)$ denotes the directional derivative of F at x along the direction d.

A point x is called a KKT point of QVI (1.1) if there exist Lagrange multipliers $\lambda \in R^{m_1}$ and $\nu \in R^{m_2}$ such that

$$\begin{cases} F(x) + \nabla_y g(x, x)\lambda + \nabla_y h(x, x)\nu = 0, \\ h(x, x) = 0, \\ \lambda \geq 0, \qquad g(x, x) \leq 0, \qquad \lambda^T g(x, x) = 0. \end{cases} \tag{2.1}$$

Similar to Theorem 1 of [12], we find that $x^* \in K(x^*)$ is a solution of (1.1) if there exist $\lambda^* \in R^{m_1}$ and $\nu^* \in R^{m_2}$ such that (x^*, λ^*, ν^*) satisfies the KKT conditions (2.1). Moreover, if $x^* \in K(x^*)$ is a solution of (1.1) and some suitable constraint qualification holds at x^*, then there exist $\lambda^* \in R^{m_1}$ and $\nu^* \in R^{m_2}$ such that (x^*, λ^*, ν^*) satisfies the KKT conditions (2.1). Based on the above relationship, our aim is to develop a numerical method for solving the KKT conditions (2.1). For convenience, let

$$L(x, \lambda, \nu) := F(x) + \nabla_y g(x, x)\lambda + \nabla_y h(x, x)\nu,$$

$$p(x) := g(x, x), \qquad q(x) := h(x, x),$$

and then (2.1) can be rewritten as

$$\begin{cases} L(x,\lambda,\nu) = 0, \\ q(x) = 0, \\ p(x) + w = 0, \\ \lambda \geq 0, \qquad w \geq 0, \qquad \lambda^T w = 0, \end{cases} \tag{2.2}$$

where the $w \in R^{m_1}$ are slack variables.

It is not easy to solve (2.2) directly since the fourth formula is a complementarity system. We replace the complementarity system by an NCP-function [17], which is called the smoothed form of FB function:

$$\phi(u,v,\varepsilon) = \sqrt{u^2 + v^2 + \varepsilon^2} - (u + v).$$

It is clear that, for each $\varepsilon \neq 0$, $\phi(u,v,\varepsilon)$ is continuously differentiable. We use it to construct an almost smooth equation reformulation to the fourth formula.

Let $\Phi_{FB}(\lambda,w) = (\phi_1^{FB}(\lambda_1,w_1),\ldots,\phi_{m_1}^{FB}(\lambda_{m_1},w_{m_1}))^T$ and $S(\lambda,w) = (S_1(\lambda,w),\ldots,S_{m_1}(\lambda,w))^T$, where for each $i = 1,2,\ldots,m_1$, the elements $\phi_i^{FB}(\lambda_i,w_i)$ and $S_i(\lambda,w)$ are given by

$$\phi_i^{FB}(\lambda_i,w_i) = \sqrt{\lambda_i^2 + w_i^2} - \lambda_i - w_i$$

and

$$S_i(\lambda,w) = \phi\bigl(\lambda_i,w_i,\mu^{\frac{1}{2}}\bigl\|\Phi_{FB}(\lambda,w)\bigr\|\bigr) = \sqrt{\lambda_i^2 + w_i^2 + 2\mu\theta(\lambda,w)} - \lambda_i - w_i, \tag{2.3}$$

respectively, where $0 < \mu < \frac{(\sqrt{2}+1)^2}{m_1}$ is a parameter, $\|\cdot\|$ is the Euclidean norm, and

$$\theta(\lambda,w) = \frac{1}{2}\bigl\|\Phi_{FB}(\lambda,w)\bigr\|^2.$$

It is obvious that, for each $i = 1,2,\ldots,m_1$, $S_i(\lambda,w)$ is differentiable everywhere except at the degenerate point (λ,w) which satisfies $\theta(\lambda,w) = 0$ and $\lambda_i = w_i = 0$ for some $i = 1,2,\ldots,m_1$. Moreover, we can obtain from Theorem 2.3 of [17] that $S(\lambda,w) = 0$ is equivalent to $\lambda \geq 0$, $w \geq 0$, $\lambda^T w = 0$. This means that (x^*,λ^*,ν^*) is a KKT point of the QVI if and only if $(x^*,\lambda^*,\nu^*,w^*)$ with $w^* = -p(x^*)$ is a solution of the nonsmooth system of equations

$$H(x,\lambda,\nu,w) = 0, \quad \text{with } H(x,\lambda,\nu,w) := \begin{pmatrix} L(x,\lambda,\nu) \\ q(x) \\ p(x) + w \\ S(\lambda,w) \end{pmatrix}. \tag{2.4}$$

Associated with the system of $H(x,\lambda,\nu,w) = 0$, we consider its natural merit function

$$\Psi(z) := \frac{1}{2}\bigl\|H(z)\bigr\|^2, \tag{2.5}$$

where we set $z := (x,\lambda,\nu,w)$.

By a direct calculation, we find that the gradient $\nabla\theta(\lambda,w)$ of $\theta(\cdot,\cdot)$ at (λ,w) can be expressed as follows:

$$\nabla\theta(\lambda,w) = \bigl(\partial\theta(\lambda,w)/\partial\lambda_1,\dots,\partial\theta(\lambda,w)/\partial\lambda_{m_1},\partial\theta(\lambda,w)/\partial w_1,\dots,\partial\theta(\lambda,w)/\partial w_{m_1}\bigr)^T,$$

where

$$\partial\theta(\lambda,w)/\partial\lambda_i = \phi_i^{FB}(\lambda_i,w_i)\nu_{\lambda_i}, \quad \text{with } \nu_{\lambda_i} \in \partial_{\lambda_i}\phi_i^{FB}(\lambda_i,w_i),$$

and

$$\partial\theta(\lambda,w)/\partial w_i = \phi_i^{FB}(\lambda_i,w_i)\nu_{w_i}, \quad \text{with } \nu_{w_i} \in \partial_{w_i}\phi_i^{FB}(\lambda_i,w_i),$$

which means that

$$\nabla\theta(\lambda,w) = \bigl[\underbrace{\operatorname{diag}(\nu_{\lambda_1},\nu_{\lambda_2},\dots,\nu_{\lambda_{m_1}})}_{V_\lambda}\underbrace{\operatorname{diag}(\nu_{w_1},\nu_{w_2},\dots,\nu_{w_{m_1}})}_{V_w}\bigr]^T\Phi_{FB}(\lambda,w).$$

Here, $\partial_{\lambda_i}\phi_i^{FB}(\lambda_i,w_i)$ denotes the partial generalized gradient of $\phi_i^{FB}(\cdot,w_i)$ at λ_i and $\partial_{w_i}\phi_i^{FB}(\lambda_i,w_i)$ denotes the partial generalized gradient of $\phi_i^{FB}(\cdot,w_i)$ at w_i, respectively. In particular, if $\theta(\lambda,w)=0$, then $\nabla\theta(\lambda,w)=0$.

If $\theta(\lambda,w)\neq 0$, we can get by a direct calculation

$$\begin{aligned}
\nabla S_i(\lambda,w) &= \left(\frac{\partial S_i}{\partial\lambda_1},\dots,\frac{\partial S_i}{\partial\lambda_{m_1}},\frac{\partial S_i}{\partial w_1},\dots,\frac{\partial S_i}{\partial w_{m_1}}\right)^T \\
&= \left[\left(\frac{\lambda_i}{\sqrt{\lambda_i^2+w_i^2+2\mu\theta}}-1\right)e_i^T,\left(\frac{w_i}{\sqrt{\lambda_i^2+w_i^2+2\mu\theta}}-1\right)e_i^T\right]^T \\
&\quad + \frac{\mu\nabla\theta(\lambda,w)}{\sqrt{\lambda_i^2+w_i^2+2\mu\theta}} \\
&= \Biggl[\Biggl(\underbrace{\frac{\lambda_i}{\sqrt{\lambda_i^2+w_i^2+2\mu\theta}}}_{a_i(\lambda,w)}-1\Biggr)e_i^T,\Biggl(\underbrace{\frac{w_i}{\sqrt{\lambda_i^2+w_i^2+2\mu\theta}}}_{b_i(\lambda,w)}-1\Biggr)e_i^T\Biggr]^T \\
&\quad + \underbrace{\frac{\sqrt{2\mu\theta}}{\sqrt{\lambda_i^2+w_i^2+2\mu\theta}}}_{c_i(\lambda,w)}\sqrt{\mu}\begin{bmatrix}V_\lambda\\ V_w\end{bmatrix}\frac{\Phi_{FB}(\lambda,w)}{\|\Phi_{FB}(\lambda,w)\|},
\end{aligned}$$

where

$$a_i^2(\lambda,w)+b_i^2(\lambda,w)+c_i^2(\lambda,w)=1.$$

Otherwise, we have $\lambda_i \geq 0$, $w_i \geq 0$ and $\lambda_i w_i = 0$ for any i, which means that, if $\lambda_i^2 + w_i^2 \neq 0$, then

$$\nabla S_i(\lambda, w) = \left(\frac{\partial S_i}{\partial \lambda_1}, \ldots, \frac{\partial S_i}{\partial \lambda_{m_1}}, \frac{\partial S_i}{\partial w_1}, \ldots, \frac{\partial S_i}{\partial w_{m_1}}\right)^T = \left[\left(\frac{\lambda_i}{\sqrt{\lambda_i^2 + w_i^2}} - 1\right)e_i^T, \left(\frac{w_i}{\sqrt{\lambda_i^2 + w_i^2}} - 1\right)e_i^T\right]^T,$$

and, if $\lambda_i^2 + w_i^2 = 0$, then the element in $\partial_C S_i(\lambda, w)$ takes the form

$$\left[(a_i - 1)e_i^T, (b_i - 1)e_i^T\right]^T + c_i\sqrt{\mu}\left[\operatorname{diag}(\bar{v}_{\lambda_1}, \bar{v}_{\lambda_2}, \ldots, \bar{v}_{\lambda_{m_1}}) \operatorname{diag}(\bar{v}_{w_1}, \bar{v}_{w_2}, \ldots, \bar{v}_{w_{m_1}})\right]^T u, \quad (2.6)$$

where $\bar{v}_{\lambda_i} \in \partial_{\lambda_i}\phi_i^{FB}(\lambda_i, w_i)$, $\bar{v}_{w_i} \in \partial_{w_i}\phi_i^{FB}(\lambda_i, w_i)$, $\|u\| = 1$, and

$$a_i^2 + b_i^2 + c_i^2 \leq 1.$$

Therefore, the partial generalized derivatives $S(\lambda, w)$ can be expressed in the form of

$$U_\lambda = \operatorname{diag}(a_1 - 1, a_2 - 1, \ldots, a_{m_1} - 1) + \sqrt{\mu}\operatorname{diag}(c_1, c_2, \ldots, c_{m_1})EV_\lambda \operatorname{diag}(u) \quad (2.7)$$

and

$$U_w = \operatorname{diag}(b_1 - 1, b_2 - 1, \ldots, b_{m_1} - 1) + \sqrt{\mu}\operatorname{diag}(c_1, c_2, \ldots, c_{m_1})EV_w \operatorname{diag}(u), \quad (2.8)$$

where $a_i^2 + b_i^2 + c_i^2 \leq 1$, $u \in R^{m_1}$ satisfies $\|u\| = 1$, E is a matrix whose elements are one, V_λ and V_w are diagonal matrices whose diagonal elements belong to $\partial_{\lambda_i}\phi_i^{FB}(\lambda_i, w_i)$ and $\partial_{w_i}\phi_i^{FB}(\lambda_i, w_i)$, respectively.

On the basis of the above calculations, we have the following proposition.

Proposition 2.2 *Let the mapping H be defined by (2.4). Then the following statements hold*:

(a) *If F is continuously differentiable and g, h are twice continuously differentiable, then H is semismooth and*

$$\partial H(x, \lambda, \nu, w) \subseteq \left\{\begin{pmatrix} J_x L(x, \lambda, \nu) & \nabla_y g(x, x) & \nabla_y h(x, x) & 0 \\ Jq(x) & 0 & 0 & 0 \\ Jp(x) & 0 & 0 & I \\ 0 & U_\lambda & 0 & U_w \end{pmatrix}\right\},$$

where U_λ, U_w is defined by (2.7) and (2.8), respectively.

(b) *If, in addition, JF, $\nabla^2 g_i$ $(i = 1, \ldots, m_1)$ and $\nabla^2 h_j$ $(j = 1, \ldots, m_2)$ are locally Lipschitz, then H is strongly semismooth.*

(c) *Let the merit function Ψ be defined by (2.5). If F is continuously differentiable and g, h are twice continuously differentiable, then Ψ is continuously differentiable, and its* *gradient is given by*

$$\nabla\Psi(z) = V^T H(z)$$

for an arbitrary element $V \in \partial H(z)$.

Remark 2.1 Consider QVI$(\tilde{K}, F)$, where

$$\tilde{K}(x) \triangleq \{y \in R^n \mid g(y,x) \le 0\}. \tag{2.9}$$

That is, there are no equality constraints in QVI (1.1). Similarly, we can formulate the above problem in terms of the nonsmooth system of equations

$$\tilde{H}(x,\lambda,w) = 0, \quad \text{with } \tilde{H}(x,\lambda,w) := \begin{pmatrix} \tilde{L}(x,\lambda) \\ p(x) + w \\ S(\lambda,w) \end{pmatrix}, \tag{2.10}$$

where $\tilde{L}(x,\lambda) := F(x) + \nabla_y g(x,x)\lambda$. Similar to the Proposition 2.2, if F is continuously differentiable and g is twice continuously differentiable, then $\tilde{H}$ is semismooth and

$$\partial\tilde{H}(x,\lambda,w) \subseteq \left\{ \begin{pmatrix} J_x\tilde{L}(x,\lambda) & \nabla_y g(x,x) & 0 \\ Jp(x) & 0 & I \\ 0 & U_\lambda & U_w \end{pmatrix} \right\},$$

where U_λ and U_w are the same as in Proposition 2.2.

Now, we present the semismooth Newton method for (1.1).

Algorithm 1 (Semismooth Newton Method)

Step 0. Choose $z^0 = (x^0, \lambda^0, \nu^0, w^0) \in R^n \times R^{m_1} \times R^{m_2} \times R^{m_1}$, $\rho > 0$, $\beta \in (0,1)$, $\sigma \in (0, \frac{1}{2})$, $p > 2$, $\varepsilon \ge 0$, and set $k := 0$.

Step 1. If $\|\nabla\Psi(z^k)\| \le \varepsilon$, stop.

Step 2. Choose an arbitrary element $V_k \in \partial H(z^k)$, and compute d^k as a solution of the linear system of equations

$$V_k d = -H(z^k). \tag{2.11}$$

If either this system is not solvable or the sufficient decrease condition

$$\nabla\Psi(z^k)^T d^k \le -\rho\|d^k\|^p \tag{2.12}$$

is not satisfied, then take $d^k := -\nabla\Psi(z^k)$.

Step 3. Compute a stepsize t_k as the maximum of the numbers β^{l_k}, $l_k = 0, 1, 2, \ldots$, such that the following Armijo condition holds:

$$\Psi(z^k + t_k d^k) \le \Psi(z^k) + \sigma t_k \nabla\Psi(z^k)^T d^k. \tag{2.13}$$

Step 4. Set $z^{k+1} := z^k + t_k d^k$, $k \leftarrow k+1$, and go to Step 1.

End.

Below, we establish the following global convergence theorem for Algorithm 1.

Theorem 2.3 *Let* $\{z^k\} = \{(x^k, \lambda^k, \nu^k, w^k)\}$ *be a sequence of iterates generated by Algorithm* 1. *Then every accumulation point of the sequence* $\{z^k\}$ *is a stationary point of the merit function* Ψ.

Proof We prove it by contradiction. Firstly, if for an infinite set of indices N, $d^k = -\nabla\Psi(z^k)$ for all $k \in N$, then, by [4] Proposition 1.16, we see that any limit point z^* of z^k satisfies $\nabla\Psi(z^*)$.

In the following, we suppose the direction is always given by (2.11). Suppose $\{z^k\} \to z^*$ and $\nabla\Psi(z^*) \neq 0$, by (2.11), we have

$$\|H(z^k)\| = \|V_k d^k\| \leq \|V_k\| \times \|d^k\|.$$

Noting that $\|V_k\|$ cannot be 0, otherwise $H(z^k) = 0$ and z^k would be a stationary point. Hence, we have

$$\|d^k\| \geq \frac{\|H(z^k)\|}{\|V_k\|}. \tag{2.14}$$

If for some subsequence N, $\{d^k\}_N \to 0$, we have by (2.14), $\{H(z^k)\}_N \to 0$, and z^* is a solution of the QVI (1.1). Hence, there exists a $m > 0$ such that $\|d^k\| \geq m$. Noting that $\{\nabla\Psi(z^k)\}_N$ is bounded and $p > 2$, there exists $M > 0$ such that $\|d^k\| \leq M$. Otherwise, it would contradict (2.12).

By (2.13) and $\{z^k\}$ is a bounded sequence, $\Psi(z^k)$ is bounded from below and $\{\Psi(z^{k+1}) - \Psi(z^k)\} \to 0$, which implies

$$\{\beta^{l_k} \nabla\Psi(z^k)^T d^k\} \to 0. \tag{2.15}$$

Suppose, subsequencing if necessary, we have $\{\beta^{l_k}\} \to 0$. By (2.13), we have

$$\frac{\Psi(z^k + \beta^{l_k - 1} d^k) - \Psi(z^k)}{\beta^{l_k - 1}} > \sigma \nabla\Psi(z^k)^T d^k. \tag{2.16}$$

By $m \leq \|d^k\| \leq M$, we can assume, subsequencing if necessary, that $\{d^k\} \to \bar{d} \neq 0$. By passing to the limit in (2.16), we get

$$\nabla\Psi(z^k)^T \bar{d} \geq \sigma \nabla\Psi(z^k)^T \bar{d}. \tag{2.17}$$

On the other hand, by (2.12), we have $\nabla\Psi(z^k)^T \bar{d} \leq -\rho\|\bar{d}\|^p < 0$, which contradicts (2.17). Hence β^{l_k} is bounded away from 0. (2.15) and (2.12) imply that $\{d^k\} \to 0$, thus contradicting $0 < m \leq \|d^k\|$, so that $\nabla\Psi(z^*) = 0$. This completes the proof. □

Remark 2.2 The method proposed in [10] only considers the case of inequality constraints, while our method can solve QVI with both equality and inequality constraints.

Besides, as we will see in the next section, our method can solve some problems in QVILIB [11], which cannot be solved by the method proposed by [10].

3 Numerical experiments

In this section, we report the results obtained by Algorithm 1 on problems list in QVILIB. All the computations in this paper were done using Matlab 2014a on a computer with 8.00 GB RAM and 2.5 GHz CPU. We solved all 55 test problems whose detailed description can be found in [11]. For each problem we list

- the x-part of the starting point (the number reported is the value of all components of the x-part of the starting point);
- the number of iterations;
- the number of evaluations of Ψ;
- the value of $Y(x, \lambda, \nu)$ at the termination.

In order to perform the linear algebra involved, we used Matlab's linear system solver *mldivide*. If any entry of the solution given by *mldivide* is a NaN or it is equal to $\pm\infty$ or the sufficient decrease condition is not satisfied, then an anti gradient direction is used. We take $\mu = 10^{-5}$, $\beta = 0.5$, $\rho = 10^{-10}$, $\sigma = 0.01$ and $p = 2.1$. We choose $\lambda^0 = 0$, $\nu^0 = 0$ and $w^0 = 0$ for all problems. For (2.6), we choose $a_i = b_i = c_i = 0$ when $(\lambda_i, w_i) = (0, 0)$ and $\theta = 0$. Our aim is mainly to verify the reliability of the method, and compare the iteration numbers

Table 1 Test results for Algorithm 1 and SSN

Problem	x^0	Algorithm 1			SSN		
		iter	Ψ	$Y(x,\lambda,\nu)$	iter	Ψ	$Y(x,\lambda,\nu)$
Box2A	10	14	17	2.8478e–05	24	77	4.1771e–06
Box2B	10	Failure			26	89	8.1684e–06
Box3A	10	10	14	2.0397e–05	10	14	5.7049e–07
Box3B	10	Failure			Failure		
KunR11	0	Failure			14	26	7.1161e–05
KunR12	0	Failure			20	53	5.1480e–05
KunR21	0	Failure			5	5	2.8047e–05
KunR22	0	Failure			5	5	2.4567e–05
KunR31	0	Failure			Failure		
KunR32	0	Failure			Failure		
MoveSet3A1	0	Failure			Failure		
MoveSet3A2	0	Failure			Failure		
MoveSet3B1	0	Failure			Failure		
MoveSet3B2	0	Failure			Failure		
MoveSet4A1	0	11	27	5.2971e–10	11	27	2.6602e–06
MoveSet4A2	0	13	41	2.0149e–09	13	43	4.0769e–07
MoveSet4B1	0	11	28	3.0273e–08	11	28	8.0836e–07
MoveSet4B2	0	13	40	1.2611e–08	13	42	8.7163e–07
OutKZ31	0	8	11	2.3605e–09	7	10	2.6128e–06
OutKZ41	0	11	20	6.2950e–06	11	21	1.7394e–05
OutZ40	0	5	5	2.5263e–08	5	5	2.5270e–08
OutZ41	0	5	5	2.8060e–08	5	5	8.6320e–07e–08
OutZ42	0	6	7	7.7261e–07	6	7	9.2896e–07
OutZ43	0	4	4	5.2452e–05	4	4	5.2459e–05
OutZ44	0	4	4	4.8112e–05	4	4	4.9362e–05
RHS1A1	0	35	242	2.5881e–08	Failure		
RHS1A1	10	35	242	2.5880e–08	Failure		
RHS2B1	0	70	559	2.3303e–08	Failure		
RHS2B1	10	70	559	2.3303e–08	Failure		
Scrim22	0	10	19	2.0003e–05	10	20	3.5249e–06
Wal2	0	17	63	8.0835e–07	20	78	2.2062e–07
Wal3	0	13	41	5.0018e–05	Failure		

Table 2 Test results for Algorithm 1 and IP

Problem	x^0	Algorithm 1				IP			
		iter	Ψ	$Y(x,\lambda,\nu)$	CPU	iter	Ψ	$Y(x,\lambda,\nu)$	CPU
BiLin1A	0	12	21	3.3093e–07	0.267200	30	90	5.4887e–05	0.172522
Box1A	0	5	5	1.3892e–08	0.235864	7	7	9.3897e–05	0.044991
Box1B	0	Failure				Failure			
Box2B	0	14	24	2.1673e–05	16.392773	19	19	1.5653e–05	190.318205
MoveSet1A	0	9	12	8.9800e–08	0.056921	24	252	3.5404e–05	0.208420
MoveSet1B	10	18	24	6.0742e–06	0.222965	Failure			
MoveSet2A	0	11	23	2.0779e–07	0.216374	56	466	9.7848e–05	0.430626
MoveSet2B	0	Failure				Failure			
MoveSet4A1	0	11	27	5.2971e–10	30.333752	10	10	9.1223e–05	177.94647
MoveSet4A2	0	13	41	2.01419e–09	359.196863	11	11	1.6374e–05	1322.551713

Table 3 Test results for rest QVIs in QVILIB

Problem	x^0	iter	Ψ	$Y(x,\lambda,\nu)$	Problem	x^0	iter	Ψ	$Y(x,\lambda,\nu)$
BiLin1B	0	9	15	6.5200e–09	Box3A	0	9	10	8.2844e–06
RHS2A1	0	39	315	2.5649e–08	RHS2A1	10	39	315	2.5649e–08
RHS1B1	10	70	349	6.9449e–09	Scrim21	0	10	19	2.1008e–06
Wal5	5	17	54	5.0645e–05	LunSS1	0	Failure		
LunSS2	0	Failure			LunSS3	0	Failure		
LunSSVI1	0	11	11	2.5630e–05	LunSSVI2	0	11	11	1.4596e–05
LunSSVI3	0	12	13	9.6398e–05	WalEq1	0	10	20	4.0512e–06
WalEq2	0	54	301	4.6316e–06	WalEq3	0	Failure		
WalEq4	0	Failure			WalEq5	0	Failure		
Scrim11	0	6	6	9.6075e–05	Scrim12	0	7	7	2.7352e–07

with the results presented in [10]. In order to perform a fair computation with the results in [10], we choose the same stopping criterion, i.e., let

$$Y(x,\lambda,\nu)=\left\|\begin{pmatrix}L(x,\lambda,\nu)\\ S(\lambda,-p(x))\end{pmatrix}\right\|_\infty,$$

and choose the termination criterion to be $Y(x^k,\lambda^k,\nu^k)\le 10^{-4}$. The iteration is also stopped if the number of iterations exceeds 500 or the stepsize t_k computed at Step 3 is less than 10^{-6}.

We denote Algorithm 2.2 proposed in [10] by SSN, and compare our method with SSN. The results are list in Table 1. From Table 1, for problems that can be solved by SSN, they can also be solved by our method with almost the same iteration numbers except problems Box2B, Box3A, KunR11, KunR12, KunR21 and KunR22. However, our method can solve the problems RHS1A1, RHS1B1, RHS2A1, RHS2B1 and Wal3, which cannot be solved by SSN.

We also compare our method with the interior point method (denoted by IP) proposed in [12] from the iteration number and CPU time. For IP, we use the same parameters presented in [12] and the results are list in Table 2. From Table 2, we can see that our method is much more effective than IP for most problems.

We also consider other problems in QVILIB which are not test in Tables 1 and 2, including the QVIs with equality constraints, that is, Problems LunSS1 to Scrim12 in Table 3. As we can see from the table, Algorithm 1 can solve over half of those problems effectively.

We tried to make some modifications to the algorithm for those problems that cannot be solved. Specifically, when calculating the Jacobian of $\tilde{H}(\tilde{z})$, we use $JF(x)$ to approximate

Table 4 Test results for modified algorithm

Problem	x^0	iter	Ψ	$Y(x,\lambda,\nu)$	Problem	x^0	iter	Ψ	$Y(x,\lambda,\nu)$
MoveSet3A1	10	41	92	7.5378e–05	MoveSet3A2	10	43	96	6.9322e–05
MoveSet3B1	10	29	60	5.1761e–05	MoveSet3B2	10	32	67	3.3559e–05

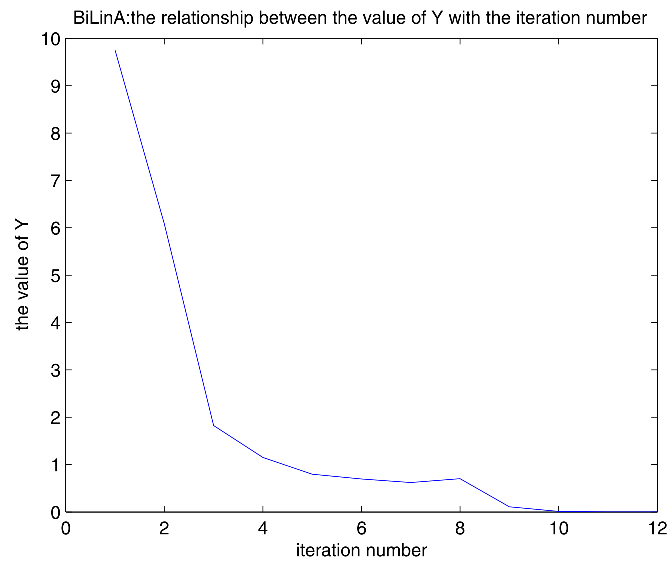

Figure 1 Algorithm 1 for BiLin1A

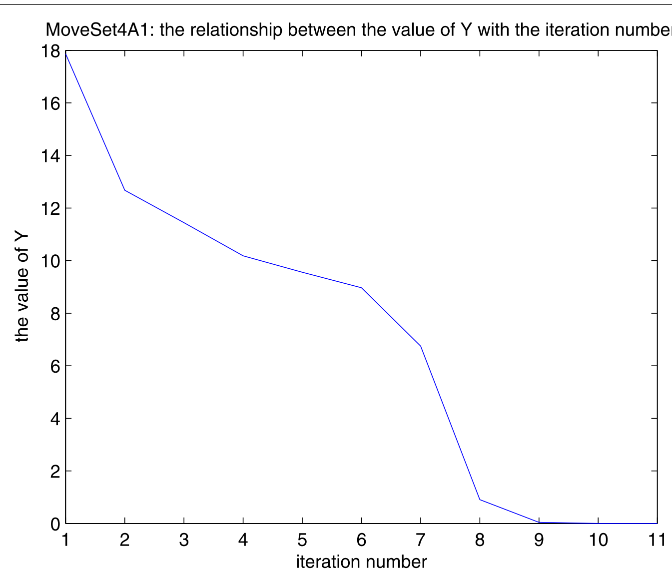

Figure 2 Algorithm 1 for Moveset4A1

$J\tilde{L}$. For now, we cannot prove the convergence of the modified algorithm. However, it is interesting to find that the modified algorithm can find a solution for some problems, such as MoveSet3A1, MoveSet3A2, MoveSet3B1 and MoveSet3B2. The results are presented in Table 4.

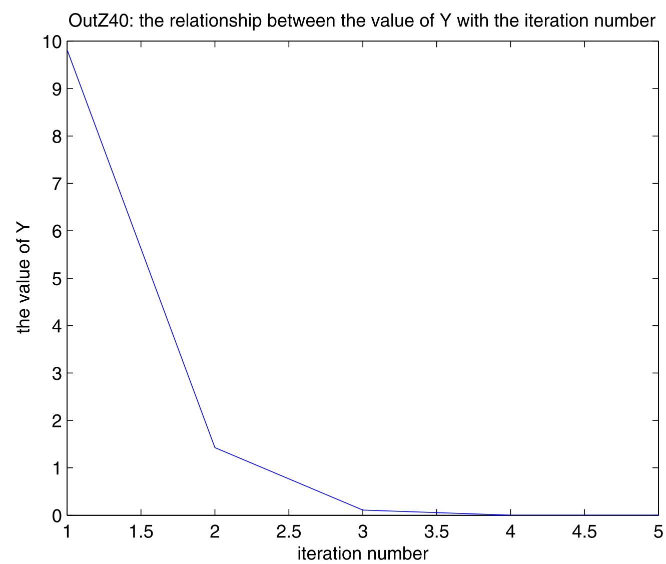

Figure 3 Algorithm 1 for OutZ40

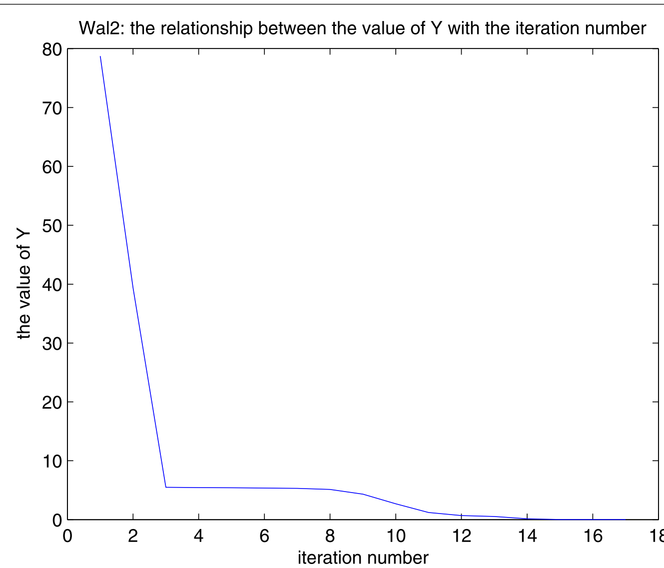

Figure 4 Algorithm 1 for Wal2

Figures 1–4 display the performance of our method on the problems BiLin1A, Movset 4A1, OutZ40 and Wal2. The vertical axis in those figures represents the value of Y and the horizontal axis represents the iteration number. As we can see from the figures, with the increase of the iteration numbers, the value of Y decrease.

Conclusion Remarks In this paper, we have studied the numerical solution of QVI. We obtain the KKT system of a QVI and present a semismooth Newton method to solve the equations. We also establish its global convergence. Numerical results show that the performance of the proposed algorithm is promising.

Acknowledgements
Not applicable.

Funding
The work was supported by the Natural Science Foundation of China (grant No. 11601188, 11761037), by the Educational Commission of Guangdong Province, China (grant No. 2019KTSCX172), by the Natural Science Foundation of Jiangxi Province (grant No. 20181BAB201009).

Availability of data and materials
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
All authors jointly worked on the results. All authors read and approved the final manuscript.

Author details
[1]School of Mathematics, Jiaying University, Meizhou, 514015, China. [2]College of Mathematics and Information Science, Jiangxi Normal University, Nanchang, China.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 22 September 2020 Accepted: 15 July 2021 Published online: 28 July 2021

References

1. Baiocchi, C., Capelo, A.: Variational and Quasivariational Inequalities: Applications to Free Boundary Problems. Wiley, New York (1984)
2. Bensoussan, A., Lions, J.L.: Nouvelle formulation de problèmes de contrôle implusionnel et applications. C. R. Acad. Sci. Paris, Sér. A **276**, 1189–1192 (1973)
3. Bensoussan, A., Lions, J.L.: Nouvelles méthodes en contrôle impulsionnel. Appl. Math. Optim. **1**, 289–312 (1975)
4. Bertserkas, D.P.: Constrained Optimization and Lagrange Multiplier Methods. Academic Press, New York (1982)
5. Chen, B., Chen, X., Kanzow, C.: A penalized Fischer–Burmeister NCP-function. Math. Program., Ser. A **88**, 211–216 (2000)
6. Dreves, A., Facchinei, F., Fischer, A., Herrich, M.: A new error bound result for generalized Nash equilibrium problems and its algorithmic application. Comput. Optim. Appl. **59**, 63–84 (2014)
7. Duan, Y., Wang, S., Zhou, Y.: A power penalty approach to a mixed quasilinear elliptic complementarity problem. J. Glob. Optim. (2021). https://doi.org/10.1007/s10898-021-01000-7
8. Facchinei, F., Fischer, A., Herrich, M.: A family of Newton methods for nonsmooth constrained systems with nonisolated solutions. Math. Methods Oper. Res. **77**, 433–443 (2013)
9. Facchinei, F., Fischer, A., Herrich, M.: An LP-Newton method: nonsmooth equations, KKT systems, and nonisolated solutions. Math. Program. **146**, 1–36 (2014)
10. Facchinei, F., Kanzow, C., Karl, S., Sagratella, S.: The semismooth Newton method for the solution of quasi-variational inequalities. Comput. Optim. Appl. **62**, 85–109 (2015)
11. Facchinei, F., Kanzow, C., Sagratella, S.: QVILIB: a library of quasi-variational inequality test problems. Pac. J. Optim. **9**, 225–250 (2013)
12. Facchinei, F., Kanzow, C., Sagratella, S.: Solving quasi-variational inequalities via their KKT conditions. Math. Program. **144**, 369–412 (2014)
13. Facchinei, F., Pang, J.S.: Finite-Dimensional Variational Inequalities and Complementarity Problems. Springer, New York (2003)
14. Jiang, H., Qi, L.: A new nonsmooth equations approach to noninear complementarity problem. SIAM J. Control Optim. **35**, 178–193 (1997)
15. Kanzow, C., Kleinmichel, H.: A new class of semismooth Newton-type methods for nonlinear complementarity problems. Comput. Optim. Appl. **11**, 227–251 (1998)
16. Li, D.H., Fukushima, M.: Globally convergent Broyden-like methods for semismooth equations and applications to VIP, NCP and MCP. Ann. Oper. Res. **103**, 71–97 (2001)
17. Li, D.H., Li, Q., Xu, H.: An almost smooth equation reformulation to the nonlinear complementarity problem and Newton's method. Optim. Methods Softw. **27**, 1–13 (2011)
18. Luca, T.D., Facchinei, F., Kanzow, C.: A semismooth equation approach to the solution of noninear complementarity problems. Math. Program. **75**, 407–439 (1996)
19. Mangasarian, O.L., Solodov, M.V.: Nonlinear complementarity as unconstrained and constrained minization. Math. Program. **62**, 277–297 (1993)
20. Mifflin, R.: Semismooth and semiconvex functions in constrained optimization. SIAM J. Control Optim. **15**, 957–972 (1997)
21. Ni, T., Zhai, J.: A regularized smoothing Newton-type algorithm for quasi-variational inequalities. Comput. Math. Appl. **68**, 1312–1324 (2014)
22. Qi, L., Sun, J.: A nonsmooth version of Newton's method. Math. Program. **58**, 353–367 (1993)
23. Wang, S.: An interior penalty method for a large-scale finite-dimensional nonlinear double obstacle problem. Appl. Math. Model. **58**, 217–228 (2018)

24. Xie, S.L., Xu, H.R., Zeng, J.P.: Two-step modulus-based matrix splitting iteration method for a class of nonlinear complementarity problems. Linear Algebra Appl. **494**, 1–10 (2016)
25. Zhao, J.X., Wang, S.: A power penalty approach to a discretized obstacle problem with nonlinear constraints. Optim. Lett. **13**, 1483–1504 (2019)