A derivative-free descent method for nonsmooth variational inequalities *

Igor V. Konnov[1], Barbara Panicucci[2], and Mauro Passacantando[3]
[1] Faculty of Computational Mathematics
and Cybernetics, Kazan University, ul. Kremlevskaya, 18,
Kazan 420008, Russia
ikonnov@ksu.ru
[2]Department of Applied Mathematics,
University of Pisa, via Buonarroti 1/c,
56127 Pisa, Italy
panicucc@mail.dm.unipi.it
[3]Department of Applied Mathematics,
University of Pisa, via Buonarroti 1/c,
56127 Pisa, Italy
passacantando@dma.unipi.it

March 19, 2008

Abstract

A descent method with gap function is proposed for a finite-dimensional variational inequality with non-integrable and nonsmooth mapping. The convergence of the method with exact line search is established under strong monotonicity conditions on the underlying mapping. Preliminary numerical results are also reported.

AMS Subject Classification: 65K10, 47J20.
Key Words and Phrases: Variational inequality, nonsmooth mapping, gap function, descent method.

1 Introduction

Let X a nonempty, closed and convex subset of $\mathbb{R}^n$ and $F : X \to \mathbb{R}^n$ a single-valued mapping. The variational inequality problem (VIP) is to find a point $x^* \in X$ such that

$$\langle F(x^*), x - x^* \rangle \geq 0 \qquad \forall\, x \in X, \tag{1}$$

*In this work, the first author was supported in part by RFBR Grant No. 07-01-00499; the second and third authors have been supported by the National Research Program PRIN/2005017083 "Innovative Problems and Methods in Nonlinear Optimization".

where $\langle \cdot, \cdot \rangle$ denotes the inner product in $\mathbb{R}^n$. Variational inequality problems occur in a number of applications, such as traffic assignment problems and other equilibrium problems. For insightful reviews of VIPs and associated solution methods, we refer the reader to recent books [2], [6], [9] and references therein. It is well known that if F is integrable, i.e. F is the gradient of a function ϕ, then (1) represents the first order optimality conditions for the problem

$$\min_{x \in X} \phi(x).$$

However, there are many VIPs in which the integrability condition for F is inappropriate. In these cases the concept of a gap function allows to reformulate a VIP as an equivalent optimization problem. For instance, Fukushima [3] proposed a gap function which turns out to be smooth if F is so, and developed a derivative-free descent method for solving problem (1). However, it is worth considering VIPs in which the mapping F is nonsmooth. In such situations, the usual Jacobians can be replaced by their multi-valued extensions. For instance, an extension of the Newton method having a local convergence in the case of the semismooth mapping F and a splitting type method for the case when F is the sum of locally Lipschitz and smooth mappings was proposed in [5]. It is well known from optimization that the descent methods are usually robust and provide the global convergence, and there exist various iterative schemes for constructing such methods; see e.g. [4], [9]. Nevertheless, the descent direction in iterative methods for VIPs usually involves a computation of some gap function, rather than an analogue of the gradient, which forces one to utilize different techniques.

In [10], a descent method for the D-gap function was proposed. A descent method with Armijo-type line search with the usual gap function was proposed in [7].

In this paper, we propose a derivative-free descent method with exact line search, which is also associated with the usual gap function for solving problem (1) under strong monotonicity and local Lipschitz continuity of the mapping F, thus extending the Fukushima's method to the nonsmooth setting. A notable feature of the algorithm to be presented is that it does not require the computation of any element of the Clarke generalized Jacobian of F. Moreover, its substantiation is essentially simpler than that of the method from [7].

2 Preliminaries

Given a symmetric positive definite matrix G, we denote by $\| \cdot \|_G$ the norm in $\mathbb{R}^n$ defined by $\|x\|_G = \sqrt{\langle x, G\,x \rangle}$. In particular, $\| \cdot \|$ denotes the classical euclidean norm induced by unit matrix I. The projection of a point $x \in \mathbb{R}^n$ onto the closed convex set X with respect to $\| \cdot \|_G$, denoted by $\Pi_{X,G}(x)$, is defined as the unique solution of the problem

$$\min_{y \in X} \|y - x\|_G.$$

It is well known that $\Pi_{X,G}(x)$ is characterized by the following condition:

$$\langle x - \Pi_{X,G}(x), G\,(z - \Pi_{X,G}(x)) \rangle \leq 0 \qquad \forall\, z \in X.$$

For a function $f : \mathbb{R}^n \to \mathbb{R}$ Lipschitz near a given point $x \in \mathbb{R}^n$, Clarke [1] proposed a natural generalization of the notion of a gradient. We recall that the *generalized directional derivative* of f at x in the direction $d \in \mathbb{R}^n$ is

$$f^\circ(x; d) = \limsup_{\substack{y \to x \\ \lambda \downarrow 0}} \frac{f(y + \lambda\, d) - f(y)}{\lambda},$$

and the *generalized gradient* of f at x is

$$\partial f(x) = \{\xi \in \mathbb{R}^n : \; f^\circ(x; d) \geq \langle \xi, d \rangle \qquad \forall\, d \in \mathbb{R}^n\}.$$

It is known [1, Proposition 2.1.2] that

$$f^\circ(x; d) = \max_{\xi \in \partial f(x)} \langle \xi, d \rangle \qquad \forall\, d \in \mathbb{R}^n. \tag{2}$$

We recall that the mapping $F : X \to \mathbb{R}^n$ is said to be:

- *monotone* if
$$\langle F(x) - F(y), x - y \rangle \geq 0 \qquad \forall\, x, y \in X;$$
- *strictly monotone* if
$$\langle F(x) - F(y), x - y \rangle > 0 \qquad \forall\, x, y \in X, \quad x \neq y;$$
- *strongly monotone* with constant $\tau > 0$ if
$$\langle F(x) - F(y), x - y \rangle \geq \tau\, \|x - y\|^2 \qquad \forall\, x, y \in X.$$

In the rest of the paper we will utilize the following assumptions.
(A1) *The set $X \subseteq \mathbb{R}^n$ is nonempty, closed, and convex.*
(A2) *The mapping $F : Y \to \mathbb{R}^n$ is locally Lipschitz at each point of an open convex set Y such that $X \subset Y$.*

Under assumption (A2), Clarke [1] introduced the *generalized Jacobian* of F at any point $x \in Y$, which is defined as

$$\partial F(x) = \operatorname{conv} \left\{ \lim_{x_i \to x} \nabla F(x_i) : \; F \text{ is differentiable at } x_i \right\}.$$

The mapping $x \longmapsto \partial F(x)$ is closed and upper semicontinuous, with nonempty convex compact values.

Monotonicity properties of F are closely related to positive semidefiniteness (or definiteness) of its generalized Jacobians.

Proposition 1. (see [5, Proposition 2.3])
Let assumptions (A1) and (A2) be fulfilled. Then:

(a) F is monotone if and only if each generalized Jacobian $V \in \partial F(x)$ is positive semidefinite for each $x \in Y$.

(b) If each generalized Jacobian $V \in \partial F(x)$ is positive definite for each $x \in Y$, then F is strictly monotone.

(c) F is strongly monotone with constant $\tau > 0$ if and only if for each $x \in Y$ and for each generalized Jacobian $V \in \partial F(x)$ one has

$$\langle d, V\, d \rangle \geq \tau\, \|d\|^2 \qquad \forall\, d \in \mathbb{R}^n.$$

3 Gap functions

In the rest of the paper we suppose that a symmetric positive definite matrix G is given and define

$$\Phi(x,y) = \langle F(x), x-y\rangle - \frac{1}{2}\|y-x\|_G^2.$$

For each $x \in X$ the function $\Phi(x,\cdot)$ is continuous and strongly concave on X, thus the problem

$$\max_{y\in X} \Phi(x,y) \tag{3}$$

has a unique solution denoted by $y(x)$. The following proposition contains several properties of the point $y(x)$.

Proposition 2. *Let assumptions* $(A1)$ *and* $(A2)$ *be fulfilled and let* $y(x)$ *be the optimal solution of problem (3). Then:*

(a)

$$\langle F(x) + G\,(y(x)-x), z-y(x)\rangle \geq 0 \qquad \forall\; z \in X. \tag{4}$$

(b) $y(x) = \Pi_{X,G}(x - G^{-1}F(x))$.

(c) the mapping $x \longmapsto y(x)$ *is continuous on* X.

(d) x^* *solves problem (1) if and only if* $x^* = y(x^*)$.

Proof. (a) It is clear that (4) is a necessary and sufficient optimality condition for the optimization problem (3).

(b) Condition (4) can be written in the form:

$$\langle G\,[y(x) - (x - G^{-1}\,F(x))], z-y(x)\rangle \geq 0 \qquad \forall\; z \in X,$$

i.e.

$$\langle [x - G^{-1}\,F(x)] - y(x), G\,[z-y(x)]\rangle \leq 0 \qquad \forall\; z \in X,$$

which is equivalent to $y(x) = \Pi_{X,G}(x - G^{-1}F(x))$.

(c) It follows from (b) and from the continuity of the operator F and projection mapping $\Pi_{X,G}$.

(d) x^* solves (1) if and only if

$$\langle x^* - F(x^*) - x^*, z - x^*\rangle \leq 0 \qquad \forall\; z \in X,$$

i.e.

$$\langle G\,[(x^* - G^{-1}\,F(x^*)) - x^*], z - x^*\rangle \leq 0 \qquad \forall\; z \in X,$$

which is equivalent to $x^* = y(x^*)$.

□

Now we define the function

$$\varphi(x) = \max_{y \in X} \Phi(x, y) = \langle F(x), x - y(x) \rangle - \frac{1}{2} \|x - y(x)\|_G^2 \tag{5}$$

which was proposed in [3] as a gap function for problem (1).

Proposition 3. *Let assumptions* (A1) *and* (A2) *be fulfilled. Then:*

(a) $\varphi(x) \geq \frac{1}{2} \|x - y(x)\|_G^2 \geq 0 \qquad \forall\, x \in X$.

(b) x^* *solves problem (1) if and only if* $x^* \in X$ *and* $\varphi(x^*) = 0$.

Proof. (a) From Proposition 2(a), with $z = x$, one has

$$\langle F(x), x - y(x) \rangle - \|x - y(x)\|_G^2 \geq 0,$$

thus

$$\varphi(x) = \langle F(x), x - y(x) \rangle - \frac{1}{2} \|x - y(x)\|_G^2 \geq \frac{1}{2} \|x - y(x)\|_G^2 \geq 0.$$

(b) If $\varphi(x^*) = 0$ with $x^* \in X$, then assertion (a) gives $x^* = y(x^*)$ which is equivalent, by Proposition 2(d), to say that x^* solves (1). Viceversa, if x^* solves (1), then $x^* = y(x^*)$ and hence, by definition of φ, we obtain $\varphi(x^*) = 0$.

□

It follows from Proposition 3 that if problem (1) has a solution, then it is equivalent to the optimization problem

$$\min_{x \in X} \ \varphi(x). \tag{6}$$

Proposition 4. (see [10, Lemma 3.2])
Let assumptions (A1) *and* (A2) *be fulfilled. Then the gap function* φ *is locally Lipschitz at any point* $x \in X$ *and its generalized gradient is given by the formula:*

$$\partial\varphi(x) = \left\{ F(x) - (V^T - G)\,(y(x) - x), \qquad V \in \partial F(x) \right\}.$$

Observe that, in general, the gap function φ is not convex and problem (6) may have local minima or stationary points which differ from the global minima. However, under the additional strong monotonicity assumption on F, problem (6) is equivalent to the stationarity condition. In the rest of the paper we will consider the following assumption:

(A3) *The mapping* $F : Y \to \mathbb{R}^n$ *is strongly monotone with constant* $\tau > 0$.

Proposition 5. (Stationary point characterization of x^*)
Let assumptions $(A1)-(A3)$ be fulfilled. Then x^* solves problem (1) if and only if x^* is a stationary point of the optimization problem (6), i.e. there exists $V \in \partial F(x^*)$ such that

$$\langle F(x^*) - (V^T - G)\,(y(x^*) - x^*), x - x^* \rangle \geq 0 \qquad \forall\, x \in X.$$

The above proposition generalizes Theorem 3.1 from [10] to the case in which G is not required to be the identity matrix. The proof is omitted as Proposition 5 can be easily proved by slightly modifying the proof of such theorem.

Proposition 5 implies that, under assumptions $(A1)-(A3)$, problem (1) can be solved by applying any of the existing subgradient algorithms to solve the optimization problem (6). Observe also that under assumptions $(A1)-(A3)$, problem (1) has a unique solution (see e.g. [2, Lemma 2.3.3]).

The following result establishes an a posteriori error bound, measured in terms of the gap function φ.

Proposition 6. (A posteriori error bound)
Let assumptions $(A1)-(A3)$ be fulfilled and let x^* be the unique solution of problem (1). Then there exists a constant $\sigma > 0$ such that

$$\varphi(x) \geq \sigma\, \|x - x^*\|^2 \qquad \forall\, x \in X. \tag{7}$$

Proof. It is well known that

$$\max_{\|x\|=1} \langle x, G\,x \rangle = \lambda_{\max}(G),$$

where $\lambda_{\max}(G)$ is the maximum eigenvalue of G. Hence, for each $x \in \mathbb{R}^n$ one has

$$\|x\|_G^2 = \langle x, G\,x \rangle \leq \lambda_{\max}(G)\, \|x\|^2. \tag{8}$$

Choose an arbitrary point $x \in X$ and $\mu \in (0,1]$, we set $x(\mu) = \mu\, x^* + (1-\mu)\, x$. Taking into account of assumption $(A3)$ and relation (8), we obtain:

$$\begin{array}{rcl} \varphi(x) & \geq & \Phi(x, x(\mu)) \\ & = & \langle F(x), x - x(\mu) \rangle - \frac{1}{2}\, \|x(\mu) - x\|_G^2 \\ & = & \mu\, \langle F(x), x - x^* \rangle - \frac{1}{2}\, \mu^2\, \|x - x^*\|_G^2 \\ & \geq & \mu\, \left[\langle F(x^*), x - x^* \rangle + \tau\, \|x - x^*\|^2 \right] - \frac{1}{2}\, \mu^2\, \|x - x^*\|_G^2 \\ & \geq & \mu\, \tau\, \|x - x^*\|^2 - \frac{1}{2}\, \mu^2\, \|x - x^*\|_G^2 \\ & \geq & \left[\mu\, \tau - \frac{1}{2}\, \mu^2\, \lambda_{\max}(G) \right]\, \|x - x^*\|^2. \end{array}$$

Therefore

$$\varphi(x) \geq \max_{\mu \in (0,1]} \left[\mu\, \tau - \frac{1}{2}\, \mu^2\, \lambda_{\max}(G) \right]\, \|x - x^*\|^2 = \sigma\, \|x - x^*\|^2,$$

where

$$\sigma = \begin{cases} \tau - \frac{1}{2}\, \lambda_{\max}(G) & \text{if } \tau \geq \lambda_{\max}(G), \\ \frac{\tau^2}{2\, \lambda_{\max}(G)} & \text{if } \tau < \lambda_{\max}(G). \end{cases}$$

□

Corollary 7. *Under the assumptions* $(A1)-(A3)$ *the gap function* φ *has compact level sets.*

Proof. This result follows directly from (7). □

4 A descent method

In this section we develop a descent algorithm based on the stationary point characterization of Proposition 5.

Proposition 8. *Let assumptions* $(A1)-(A3)$ *be fulfilled. Then for each* $x \in X$ *the vector* $d = y(x) - x$ *satisfies the following descent condition:*

$$\varphi^{\circ}(x; d) \leq -\tau \, \|d\|^2. \tag{9}$$

Proof. From Proposition 4 it follows that

$$\varphi^{\circ}(x; d) = \max_{V \in \partial F(x)} \langle F(x) - (V^T - G) \, d, d \rangle.$$

Using Proposition 2(a) with $z = x$ and Proposition 1, we obtain that for each $V \in \partial F(x)$ one has

$$\langle F(x) - (V^T - G) \, d, d \rangle \leq -\langle V^T d, d \rangle = -\langle d, V \, d \rangle \leq -\tau \, \|d\|^2,$$

hence inequality (9) is true. □

Proposition 8 implies that if a point $x \in X$ does not solve problem (1), then the vector $d = y(x) - x \neq 0$ is a descent direction at x for the gap function φ, indeed:

$$\limsup_{t \downarrow 0} \frac{\varphi(x + t \, d) - \varphi(x)}{t} \leq \varphi^{\circ}(x; d) \leq -\tau \, \|d\|^2 < 0.$$

Now, we propose a descent method based on the direction $d = y(x) - x$.

Descent algorithm

Step 0. Choose a point $x^0 \in X$ and a symmetric definite positive matrix G. Set $k = 0$.

Step 1. Find $y(x^k) = \Pi_{X,G}(x^k - G^{-1} \, F(x^k))$ and set $d^k = y(x^k) - x^k$.

Step 2. If $d^k = 0$, then STOP; otherwise find

$$t_k \in \arg \min_{t \in [0,1]} \varphi(x^k + t \, d^k), \tag{10}$$

set $x^{k+1} = x^k + t_k \, d^k$, set $k := k + 1$ and go to step 1.

We next show that, using the exact line search rule (10), from any feasible starting point the sequence of iteration points converges towards the unique solution of problem (1).

Theorem 9. (Global convergence)
Let assumptions $(A1) - (A3)$ be fulfilled. The descent algorithm either gives the solution of problem (1) after a finite number of iterations, or generates an infinite sequence $\{x^k\}$ converging to the solution of problem (1).

Proof. By Proposition 2(d) the stopping criterion $d^k = 0$ means that x^k solves problem (1), hence we will consider the case when the algorithm generates an infinite sequence $\{x^k\}$. To prove the global convergence of the algorithm we show that the assumptions of Zangwill's Convergence Theorem [11, p. 91] are fulfilled:

- the sequence $\{x^k\}$ is contained in a compact subset of X because φ has compact level sets by Corollary 7.
- $\varphi(x^{k+1}) < \varphi(x^k)$ by Proposition 8 and the choice of the step length.
- The direction $d = y(x) - x$ is continuous with respect to x because F and $\Pi_{X,G}$ are continuous. It is well known that the point-to-set map assigning to each pair (x, d) the points determined by the exact line-search rule (10) is closed, hence the point-to-set algorithmic map associated with the descent algorithm is closed.

Therefore, by Zangwill's Theorem and Proposition 5 any cluster point of the sequence $\{x^k\}$ is a solution of problem (1). Since there is a unique solution of (1), the entire sequence $\{x^k\}$ converges to the solution of problem (1). □

It should be remarked that the implementation of the line search can utilize any known one-dimensional algorithm within a certain tolerance, if necessary. Global convergence with an Armijo step length rule can also be established (see [7]).

5 Computational experiments

To give some insight into the behavior of the proposed descent algorithm we implemented it in MATLAB 7.0.4. In the implementation we set the matrix G as the identity matrix. The convergence criterion was $\|d^k\| \leq 10^{-4}$.

Note that at every step of the algorithm we need to perform a projection and a line search. We performed the projection using the quadratic-program solver QUADPROG and we performed the line search using the function FMINBND, both from the MATLAB Optimization Toolbox. The test problems we considered are from [10]. In the following we report the data of the examples for the reader's convenience. The tables summarize the numerical results of the algorithm: the first column indicates the test number, the second one reports the starting point, the third one shows the number of iterations computed, the fourth and fifth columns give, respectively, the value of the gap function φ at x and the norm of mapping F at x. In the bottom, the approximated solution x where the algorithm terminated is indicated for each test data.

Example 1. Consider problem (1) where components of $F(x)$ are given as follows:

$$\begin{aligned}
F_1(x) = \;& 0.726\,x_1 - 0.949\,x_2 + 0.266\,x_3 - 1.193\,x_4 - 0.504\,x_5 \\
& +10\,\max\{\arctan(x_1-2), \arctan(2\,x_1-4)\} + 5.308 \\
F_2(x) = \;& 1.645\,x_1 + 0.678\,x_2 + 0.333\,x_3 - 0.217\,x_4 - 1.443\,x_5 \\
& +10\,\arctan(x_2-2) + 0.008 \\
F_3(x) = \;& -1.016\,x_1 - 0.225\,x_2 + 0.769\,x_3 + 0.934\,x_4 + 1.007\,x_5 \\
& +10\,\arctan(x_3-2) - 0.938 \\
F_4(x) = \;& 1.063\,x_1 + 0.567\,x_2 - 1.144\,x_3 + 0.550\,x_4 - 0.548\,x_5 \\
& +10\,\arctan(x_4-2) + 1.024 \\
F_5(x) = \;& -0.259\,x_1 + 1.453\,x_2 - 1.073\,x_3 + 0.509\,x_4 + 1.026\,x_5 \\
& +10\,\arctan(x_5-2) - 1.312
\end{aligned}$$

The function F is strongly monotone and piecewise continuously differentiable. First, we set $X = [1,6] \times \cdots \times [1,6]$ and we applied the algorithm to solve this problem with several vertices of X as initial points. The results are displayed in Table 1. It is worth noting that the solution is in the interior of X.

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1,1,1,1,1)$	8	6.68E-10	3.66E-05
2	$(1,1,1,6,6)$	11	2.59E-09	7.20E-05
3	$(1,1,6,6,1)$	10	1.71E-09	5.86E-05
4	$(1,6,1,1,6)$	9	3.47E-09	8.33E-05
5	$(1,6,6,1,1)$	10	3.24E-10	2.55E-05
6	$(1,6,6,6,6)$	12	3.49E-09	8.35E-05
7	$(6,1,1,6,1)$	13	8.90E-10	4.22E-05
8	$(6,1,6,1,6)$	11	9.12E-10	4.27E-05
9	$(6,6,1,1,1)$	11	7.71E-10	3.93E-05
10	$(6,6,1,6,6)$	12	4.95E-09	9.95E-05
11	$(6,6,6,6,6)$	8	2.65E-10	2.30E-05

No.	x
1	$(1.769783, 1.824792, 1.819678, 1.812395, 1.825833)$
2	$(1.769784, 1.824795, 1.819674, 1.812398, 1.825833)$
3	$(1.769784, 1.824794, 1.819676, 1.812396, 1.825832)$
4	$(1.769785, 1.824790, 1.819679, 1.812392, 1.825830)$
5	$(1.769783, 1.824792, 1.819677, 1.812396, 1.825834)$
6	$(1.769778, 1.824787, 1.819682, 1.812395, 1.825838)$
7	$(1.769780, 1.824790, 1.819679, 1.812396, 1.825837)$
8	$(1.769783, 1.824791, 1.819678, 1.812394, 1.825833)$
9	$(1.769783, 1.824792, 1.819677, 1.812395, 1.825833)$
10	$(1.769778, 1.824786, 1.819682, 1.812395, 1.825839)$
11	$(1.769780, 1.824791, 1.819678, 1.812397, 1.825837)$

Table 1: Results for Example 1 with $X = [1,6] \times \cdots \times [1,6]$.

Then, we considered $X = [1,6] \times [2,6] \times [3,6] \times [4,6] \times [5,6]$ so that the solution is not in the interior of X. Applying the algorithm with several vertices of X as starting points, we obtained the results displayed in Table 2.

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1,2,3,4,5)$	14	4.14E-10	2.62E+01
2	$(1,2,3,6,6)$	39	9.60E-07	2.62E+01
3	$(1,2,6,6,5)$	44	2.15E-06	2.62E+01
4	$(1,6,3,4,6)$	46	6.67E-09	2.62E+01
5	$(1,6,6,4,5)$	31	4.90E-07	2.62E+01
6	$(1,6,6,6,6)$	38	2.68E-07	2.62E+01
7	$(6,2,3,6,5)$	25	9.33E-07	2.62E+01
8	$(6,2,6,4,6)$	41	2.42E-07	2.62E+01
9	$(6,6,3,4,5)$	14	4.14E-10	2.62E+01
10	$(6,6,3,6,6)$	56	5.29E-09	2.62E+01
11	$(6,6,6,6,6)$	43	3.80E-07	2.62E+01

No.	x
1	$(2.089579, 2.216870, 3.000000, 4.000000, 5.000000)$
2	$(2.089580, 2.216868, 3.000000, 4.000000, 5.000000)$
3	$(2.089581, 2.216868, 3.000000, 4.000000, 5.000000)$
4	$(2.089579, 2.216868, 3.000000, 4.000000, 5.000000)$
5	$(2.089583, 2.216868, 3.000000, 4.000000, 5.000000)$
6	$(2.089583, 2.216868, 3.000000, 4.000000, 5.000000)$
7	$(2.089579, 2.216870, 3.000000, 4.000000, 5.000000)$
8	$(2.089584, 2.216868, 3.000000, 4.000000, 5.000000)$
9	$(2.089579, 2.216870, 3.000000, 4.000000, 5.000000)$
10	$(2.089579, 2.216868, 3.000000, 4.000000, 5.000000)$
11	$(2.089575, 2.216867, 3.000000, 4.000000, 5.000000)$

Table 2: Results for Example 1 with $X = [1,6] \times [2,6] \times [3,6] \times [4,6] \times [5,6]$.

Example 2. Consider problem (1) where components of $F(x)$ are given as follows:

$$\begin{aligned}
F_1(x) = {} & 0.726\,x_1 - 0.949\,x_2 + 0.266\,x_3 - 1.193\,x_4 - 0.504\,x_5 \\
& +10\ \max\{\arctan(x_1 - 2), \arctan(x_1 + x_2 - 4)\} + 5.308 \\
F_2(x) = {} & 1.645\,x_1 + 0.678\,x_2 + 0.333\,x_3 - 0.217\,x_4 - 1.443\,x_5 \\
& +10\ \arctan(x_2 - 2) + 0.008 \\
F_3(x) = {} & -1.016\,x_1 - 0.225\,x_2 + 0.769\,x_3 + 0.934\,x_4 + 1.007\,x_5 \\
& +10\ \arctan(x_3 - 2) - 0.938 \\
F_4(x) = {} & 1.063\,x_1 + 0.567\,x_2 - 1.144\,x_3 + 0.550\,x_4 - 0.548\,x_5 \\
& +10\ \arctan(x_4 - 2) + 1.024 \\
F_5(x) = {} & -0.259\,x_1 + 1.453\,x_2 - 1.073\,x_3 + 0.509\,x_4 + 1.026\,x_5 \\
& +10\ \arctan(x_5 - 2) - 1.312;
\end{aligned}$$

The function F is strongly monotone and piecewise continuously differentiable. First, we set $X = [1,6] \times \cdots \times [1,6]$ and we applied the algorithm to solve this problem with several vertices of X as initial points. The results are displayed in Table 3.

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1,1,1,1,1)$	8	6.68E-10	3.66E-05
2	$(1,1,1,6,6)$	11	2.49E-09	7.06E-05
3	$(1,1,6,6,1)$	10	1.71E-09	5.86E-05
4	$(1,6,1,1,6)$	10	8.93E-10	4.23E-05
5	$(1,6,6,1,1)$	10	4.56E-09	9.55E-05
6	$(1,6,6,6,6)$	10	1.08E-09	4.64E-05
7	$(6,1,1,6,1)$	13	9.78E-10	4.42E-05
8	$(6,1,6,1,6)$	11	2.16E-09	6.57E-05
9	$(6,6,1,1,1)$	11	7.71E-10	3.93E-05
10	$(6,6,1,6,6)$	13	4.63E-10	3.04E-05
11	$(6,6,6,6,6)$	8	2.65E-10	2.30E-05

No.	x
1	$(1.769783, 1.824792, 1.819677, 1.812395, 1.825833)$
2	$(1.769784, 1.824795, 1.819674, 1.812398, 1.825833)$
3	$(1.769784, 1.824794, 1.819676, 1.812396, 1.825832)$
4	$(1.769781, 1.824789, 1.819680, 1.812395, 1.825836)$
5	$(1.769779, 1.824795, 1.819675, 1.812401, 1.825840)$
6	$(1.769779, 1.824789, 1.819679, 1.812396, 1.825838)$
7	$(1.769782, 1.824794, 1.819676, 1.812398, 1.825835)$
8	$(1.769784, 1.824795, 1.819675, 1.812397, 1.825833)$
9	$(1.769783, 1.824792, 1.819677, 1.812395, 1.825833)$
10	$(1.769780, 1.824792, 1.819677, 1.812398, 1.825837)$
11	$(1.769780, 1.824791, 1.819678, 1.812397, 1.825837)$

Table 3: Results for Example 2 with $X = [1,6] \times \cdots \times [1,6]$.

Then, we considered $X = [1,6] \times [2,6] \times [3,6] \times [4,6] \times [5,6]$ so that the solution is not in the interior of X. In Table 4 are shown the results obtained applying the algorithm with several vertices of X as starting points.

Example 3. Consider problem (1) where components of $F(x)$ are given as follows:

$$\begin{aligned}
F_1(x) = {} & 0.726\,x_1 - 0.949\,x_2 + 0.266\,x_3 - 1.193\,x_4 - 0.504\,x_5 \\
& +10 \max\{\arctan(x_1 - 2), \arctan(x_1 + x_2 - 4)\} + 5.308 \\
F_2(x) = {} & 1.645\,x_1 + 0.678\,x_2 + 0.333\,x_3 - 0.217\,x_4 - 1.443\,x_5 \\
& +10 \max\{\arctan(x_2 - 2), \arctan(x_2 + x_3 - 4)\} + 0.008 \\
F_3(x) = {} & -1.016\,x_1 - 0.225\,x_2 + 0.769\,x_3 + 0.934\,x_4 + 1.007\,x_5 \\
& +10 \arctan(x_3 - 2) - 0.938 \\
F_4(x) = {} & 1.063\,x_1 + 0.567\,x_2 - 1.144\,x_3 + 0.550\,x_4 - 0.548\,x_5 \\
& +10 \arctan(x_4 - 2) + 1.024 \\
F_5(x) = {} & -0.259\,x_1 + 1.453\,x_2 - 1.073\,x_3 + 0.509\,x_4 + 1.026\,x_5 \\
& +10 \arctan(x_5 - 2) - 1.312
\end{aligned}$$

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1, 2, 3, 4, 5)$	9	2.09E-09	2.63E+01
2	$(1, 2, 3, 6, 6)$	30	4.06E-10	2.63E+01
3	$(1, 2, 6, 6, 5)$	38	8.85E-07	2.63E+01
4	$(1, 6, 3, 4, 6)$	21	7.35E-07	2.63E+01
5	$(1, 6, 6, 4, 5)$	23	1.47E-06	2.63E+01
6	$(1, 6, 6, 6, 6)$	23	4.12E-06	2.63E+01
7	$(6, 2, 3, 6, 5)$	28	6.05E-07	2.63E+01
8	$(6, 2, 6, 4, 6)$	25	1.96E-09	2.63E+01
9	$(6, 6, 3, 4, 5)$	9	2.09E-09	2.63E+01
10	$(6, 6, 3, 6, 6)$	24	6.22E-07	2.63E+01
11	$(6, 6, 6, 6, 6)$	31	3.19E-08	2.63E+01

No.	x
1	$(1.952634, 2.238983, 3.000000, 4.000000, 5.000000)$
2	$(1.952624, 2.238990, 3.000000, 4.000000, 5.000000)$
3	$(1.952631, 2.238993, 3.000000, 4.000000, 5.000000)$
4	$(1.952624, 2.238985, 3.000000, 4.000000, 5.000000)$
5	$(1.952621, 2.238994, 3.000000, 4.000000, 5.000000)$
6	$(1.952617, 2.238996, 3.000000, 4.000000, 5.000000)$
7	$(1.952623, 2.238992, 3.000000, 4.000000, 5.000000)$
8	$(1.952630, 2.238984, 3.000000, 4.000000, 5.000000)$
9	$(1.952634, 2.238983, 3.000000, 4.000000, 5.000000)$
10	$(1.952623, 2.238992, 3.000000, 4.000000, 5.000000)$
11	$(1.952628, 2.238983, 3.000000, 4.000000, 5.000000)$

Table 4: Results for Example 2 with $X = [1, 6] \times [2, 6] \times [3, 6] \times [4, 6] \times [5, 6]$.

The function F is strongly monotone and piecewise continuously differentiable. First, we set $X = [1, 6] \times \cdots \times [1, 6]$ and we applied the algorithm to solve this problem with several vertices of X as initial points. The results are shown in Table 5.

Then, we considered $X = [1, 6] \times [2, 6] \times [3, 6] \times [4, 6] \times [5, 6]$ so that the solution is not in the interior of X. We applied the algorithm with several vertices of X as starting points and we obtained the results shown in Table 6.

Example 4. Consider problem (1) where components of $F(x)$ are given

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1,1,1,1,1)$	8	6.68E-10	3.66E-05
2	$(1,1,1,6,6)$	11	2.55E-09	7.14E-05
3	$(1,1,6,6,1)$	10	4.66E-09	9.66E-05
4	$(1,6,1,1,6)$	10	2.08E-09	6.45E-05
5	$(1,6,6,1,1)$	11	9.10E-10	4.27E-05
6	$(1,6,6,6,6)$	10	1.08E-09	4.64E-05
7	$(6,1,1,6,1)$	11	8.66E-10	4.16E-05
8	$(6,1,6,1,6)$	12	8.14E-10	4.04E-05
9	$(6,6,1,1,1)$	12	6.47E-10	3.60E-05
10	$(6,6,1,6,6)$	11	1.01E-09	4.49E-05
11	$(6,6,6,6,6)$	8	2.65E-10	2.30E-05

No.	x
1	$(1.769783, 1.824792, 1.819677, 1.812395, 1.825833)$
2	$(1.769784, 1.824795, 1.819674, 1.812397, 1.825833)$
3	$(1.769780, 1.824785, 1.819683, 1.812392, 1.825835)$
4	$(1.769781, 1.824787, 1.819681, 1.812393, 1.825835)$
5	$(1.769779, 1.824791, 1.819678, 1.812398, 1.825838)$
6	$(1.769779, 1.824789, 1.819679, 1.812396, 1.825838)$
7	$(1.769784, 1.824792, 1.819677, 1.812395, 1.825833)$
8	$(1.769780, 1.824789, 1.819680, 1.812396, 1.825837)$
9	$(1.769783, 1.824792, 1.819678, 1.812395, 1.825833)$
10	$(1.769781, 1.824794, 1.819676, 1.812398, 1.825836)$
11	$(1.769780, 1.824791, 1.819678, 1.812397, 1.825837)$

Table 5: Results for Example 3 with $X = [1,6] \times \cdots \times [1,6]$.

as follows:

$$\begin{aligned}
F_1(x) = \;& 0.726\,x_1 - 0.949\,x_2 + 0.266\,x_3 - 1.193\,x_4 - 0.504\,x_5 \\
& +10\,\max\{\arctan(|x_1| - 2), \arctan(|x_1 + x_2| - 4)\} + 5.308 \\
F_2(x) = \;& 1.645\,x_1 + 0.678\,x_2 + 0.333\,x_3 - 0.217\,x_4 - 1.443\,x_5 \\
& +10\,\max\{\arctan(|x_2| - 2), \arctan(|x_2 + x_3| - 4)\} + 0.008 \\
F_3(x) = \;& -1.016\,x_1 - 0.225\,x_2 + 0.769\,x_3 + 0.934\,x_4 + 1.007\,x_5 \\
& +10\,\max\{\arctan(|x_3| - 2), \arctan(|x_3 + x_4| - 4)\} - 0.938 \\
F_4(x) = \;& 1.063\,x_1 + 0.567\,x_2 - 1.144\,x_3 + 0.550\,x_4 - 0.548\,x_5 \\
& +10\,\max\{\arctan(|x_4| - 2), \arctan(|x_4 + x_5| - 4)\} + 1.024 \\
F_5(x) = \;& -0.259\,x_1 + 1.453\,x_2 - 1.073\,x_3 + 0.509\,x_4 + 1.026\,x_5 \\
& +10\,\max\{\arctan(|x_5| - 2), \arctan(|x_1 + x_5| - 4)\} - 1.312
\end{aligned}$$

The function F is strongly monotone and piecewise continuously differentiable. First, we set $X = [1,6] \times \cdots \times [1,6]$ and we applied the algorithm to solve this problem with several vertices of X as initial points. The results are displayed in Table 7.

Then, we considered $X = [1,6] \times [2,6] \times [3,6] \times [4,6] \times [5,6]$ so that the solution is not in the interior of X. We applied the algorithm with several vertices of X as starting points and the results are displayed in Table 8.

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1, 2, 3, 4, 5)$	2	7.77E-17	2.66E+01
2	$(1, 2, 3, 6, 6)$	5	8.49E-08	2.66E+01
3	$(1, 2, 6, 6, 5)$	10	2.05E-05	2.66E+01
4	$(1, 6, 3, 4, 6)$	5	5.63E-05	2.66E+01
5	$(1, 6, 6, 4, 5)$	7	9.56E-05	2.66E+01
6	$(1, 6, 6, 6, 6)$	9	1.49E-04	2.66E+01
7	$(6, 2, 3, 6, 5)$	22	9.10E-08	2.66E+01
8	$(6, 2, 6, 4, 6)$	15	2.81E-05	2.66E+01
9	$(6, 6, 3, 4, 5)$	5	1.08E-08	2.66E+01
10	$(6, 6, 3, 6, 6)$	5	9.72E-05	2.66E+01
11	$(6, 6, 6, 6, 6)$	7	1.15E-04	2.66E+01

No.	x
1	$(2.153257, 2.000000, 3.000000, 4.000000, 5.000000)$
2	$(2.153256, 2.000000, 3.000000, 4.000000, 5.000000)$
3	$(2.153256, 2.000000, 3.000001, 4.000001, 5.000000)$
4	$(2.153246, 2.000006, 3.000000, 4.000000, 5.000001)$
5	$(2.153251, 2.000006, 3.000004, 4.000000, 5.000000)$
6	$(2.153258, 2.000005, 3.000004, 4.000003, 5.000001)$
7	$(2.153257, 2.000000, 3.000000, 4.000000, 5.000000)$
8	$(2.153255, 2.000000, 3.000001, 4.000000, 5.000000)$
9	$(2.153257, 2.000000, 3.000000, 4.000000, 5.000000)$
10	$(2.153251, 2.000006, 3.000000, 4.000003, 5.000002)$
11	$(2.153252, 2.000004, 3.000003, 4.000002, 5.000001)$

Table 6: Results for Example 3 with $X = [1, 6] \times [2, 6] \times [3, 6] \times [4, 6] \times [5, 6]$.

We remark that computational tests were also carried out using further vertices as starting points: such tests provide similar results which were omitted for brevity. Computational results show that the proposed algorithm appears to be rather robust. In fact no failure was noticed and the number of iteration does not change very much with respect to the chosen starting point. Furthermore, its convergence is rather fast.

It is worth noting that our algorithm requires feasible starting points, while in [10] computational tests on the descent method based on D-gap function are carried out starting from unfeasible points. However, it is not difficult to find such a feasible point within simple constrained sets. Our convergence results seem rather good even in comparison with those obtained in [10] on the same test examples, although with different starting points, as our method needs on the average less iterations and no failure was noticed. Moreover, our method requires the computation of one gap function per iteration, while the method based on D-gap function requires the evaluation of two gap functions per iteration.

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1,1,1,1,1)$	8	6.68E-10	3.66E-05
2	$(1,1,1,6,6)$	11	6.79E-10	3.68E-05
3	$(1,1,6,6,1)$	11	1.00E-09	4.47E-05
4	$(1,6,1,1,6)$	10	8.21E-10	4.05E-05
5	$(1,6,6,1,1)$	11	4.18E-10	2.89E-05
6	$(1,6,6,6,6)$	10	1.08E-09	4.64E-05
7	$(6,1,1,6,1)$	11	6.33E-10	3.56E-05
8	$(6,1,6,1,6)$	10	2.13E-09	6.52E-05
9	$(6,6,1,1,1)$	13	5.68E-10	3.37E-05
10	$(6,6,1,6,6)$	9	2.77E-09	7.44E-05
11	$(6,6,6,6,6)$	8	2.65E-10	2.30E-05

No.	x
1	$(1.769783, 1.824792, 1.819677, 1.812395, 1.825833)$
2	$(1.769780, 1.824792, 1.819677, 1.812398, 1.825837)$
3	$(1.769780, 1.824792, 1.819677, 1.812398, 1.825838)$
4	$(1.769781, 1.824789, 1.819680, 1.812394, 1.825835)$
5	$(1.769781, 1.824790, 1.819679, 1.812395, 1.825835)$
6	$(1.769779, 1.824789, 1.819679, 1.812396, 1.825838)$
7	$(1.769783, 1.824793, 1.819676, 1.812397, 1.825834)$
8	$(1.769784, 1.824795, 1.819675, 1.812398, 1.825833)$
9	$(1.769780, 1.824791, 1.819678, 1.812397, 1.825837)$
10	$(1.769778, 1.824791, 1.819678, 1.812399, 1.825840)$
11	$(1.769780, 1.824791, 1.819678, 1.812397, 1.825837)$

Table 7: Results for Example 4 with $X = [1,6] \times \cdots \times [1,6]$.

No.	x^0	iter	$\varphi(x)$	$\|F(x)\|$
1	$(1,2,3,4,5)$	2	7.77E-17	3.07E+01
2	$(1,2,3,6,6)$	5	9.42E-08	3.07E+01
3	$(1,2,6,6,5)$	4	7.33E-05	3.07E+01
4	$(1,6,3,4,6)$	5	5.43E-05	3.07E+01
5	$(1,6,6,4,5)$	7	1.20E-04	3.07E+01
6	$(1,6,6,6,6)$	4	3.22E-07	3.07E+01
7	$(6,2,3,6,5)$	22	1.68E-06	3.07E+01
8	$(6,2,6,4,6)$	19	3.91E-06	3.07E+01
9	$(6,6,3,4,5)$	5	1.08E-08	3.07E+01
10	$(6,6,3,6,6)$	5	1.17E-04	3.07E+01
11	$(6,6,6,6,6)$	7	1.40E-04	3.07E+01

No.	x
1	$(2.153257, 2.000000, 3.000000, 4.000000, 5.000000)$
2	$(2.153255, 2.000000, 3.000000, 4.000000, 5.000000)$
3	$(2.153250, 2.000000, 3.000002, 4.000002, 5.000000)$
4	$(2.153247, 2.000005, 3.000000, 4.000000, 5.000001)$
5	$(2.153250, 2.000006, 3.000004, 4.000000, 5.000000)$
6	$(2.153261, 2.000000, 3.000000, 4.000000, 5.000000)$
7	$(2.153258, 2.000000, 3.000000, 4.000000, 5.000000)$
8	$(2.153256, 2.000000, 3.000000, 4.000000, 5.000000)$
9	$(2.153257, 2.000000, 3.000000, 4.000000, 5.000000)$
10	$(2.153246, 2.000007, 3.000000, 4.000003, 5.000002)$
11	$(2.153253, 2.000004, 3.000003, 4.000002, 5.000001)$

Table 8: Results for Example 4 with $X = [1,6] \times [2,6] \times [3,6] \times [4,6] \times [5,6]$.

References

[1] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley and Sons, New York (1983).

[2] F. Facchinei, J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer, Berlin, (2003).

[3] M. Fukushima, Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems, *Math. Progr.*, **53** (1992), 99–110.

[4] U.M. Garcia-Palomares, J.F. Rodriguez, New sequential and parallel derivative-free algorithms for unconstrained optimization, *SIAM J. Optimiz.*, **13** (2002), 79–96.

[5] H. Jiang, L. Qi, Local uniqueness and convergence of iterative methods for nonsmooth variational inequalities, *J. Mathem. Anal. Appl.*, **196** (1995), 314–331.

[6] I. Konnov, *Combined Relaxation Methods for Variational Inequalities*, Springer, Berlin (2001).

[7] I. Konnov, Descent methods for nonsmooth variational inequalities, *Comput. Maths. Math. Phys.*, **46** (2001), 1186–1192.

[8] I. Konnov, O. Pinyagina, D-gap functions and descent methods for a class of monotone equilibrium problems, *Lobachevskii J. Mathem.*, **13** (2003), 57–65.

[9] M. Patriksson, *Nonlinear Programming and Variational Inequality Problems: A Unified Approach*, Kluwer Academic Publishers, Dordrecht (1998).

[10] H. Xu, Regularized gap function and D-gap function for nonsmooth variational inequalities, In: *Optimization and Related Topics*, Kluwer, Dordrecht (2001), 153–176.

[11] W. I. Zangwill, *Nonlinear Programming: A Unified Approach*, Prentice-Hall, Englewood Cliffs (1969).