Rend. Istit. Mat. Univ. Trieste
Vol. 56 (2024), Art. No. 7, 20 pages
DOI: 10.13137/2464-8728/36848

A new version of projection method for variational inequalities problem with nonlinear constraints

Hassina Grar and Zahira Kebaili

Abstract. *We have proposed in [7], a new projection or extragradient method to solve many variational inequalities problem classes. The corresponding algorithm is established under continuity and pseudomonotonicity of the underlining mapping. The numerical implementation results express its remarkable efficiency. In this paper, we extend the application of this algorithm to the class of nonlinear constraints. The main idea is to linearize the constraints in the neighbourhood of each iterate, then we calculate the necessary projections. It is important to point out that most of the theoretical results already obtained in our previous work will be modified and justified according to the class of problems studied in this paper. The global convergence is proven under weak hypothesis. The numerical results are very encouraging and show that the method is also very efficient to solve this class of problems.*

Keywords: Variational inequalities problem, Nonlinear constraints, Linearized constraints, Projection methods, Fixed point problem.
MS Classification 2020: 65K05, 65K15, 90C33.

1. Introduction

Let C be a nonempty closed and convex set in $\mathbb{R}^n$ and F a continuous mapping from $\mathbb{R}^n$ to itself. The classical variational inequalities problem abbreviated $VIP\,(F, C)$ or simply (VIP) is to find a point $\overline{x} \in C$ such that

$$\langle F(\overline{x}), x - \overline{x} \rangle \geq 0, \textit{ for all } x \in C, \tag{1}$$

where $\langle \cdot, \cdot \rangle$ denotes the usual inner product in $\mathbb{R}^n$.

Without loss of generality, we assume that $C = \{x \in \mathbb{R}^n : g_i(x) \leq 0, i = 1, \ldots, m\}$, where $g_i : \mathbb{R}^n \to \mathbb{R}$, $(i = 1, \ldots, m)$ are continuously differentiable convex functions.

Throughout this paper, we denote by $\mathcal{S}$ the solutions set of $VIP(F, C)$.

In these last decades, the general variational inequalities problem, especially the class with nonlinear constraints, has become a very useful subject in many areas of mathematical programming. This development includes a rich theory, a variety of efficient resolution algorithms and lot of important applications in engineering, economy, transportation, signal processing, structural analysis... Interested reader may refer to the survey paper of Harker and Pang [8] and the paper of Ferris and Pang [2]. Many researchers have studied various numerical methods. The most popular ones are the projection methods which are mainly based on the reformulation of $VIP(F,C)$ as fixed point problem.

The first proposed projection methods suffered from significant theoretical and algorithmic difficulties [9, 11]. Later, several studies were completed, in particular [10, 15, 17] in order to overcome these difficulties. Consequently, many developments were brought to improve the algorithmic behavior of this type of methods. The principal achievement in this framework was to reduce to only two the number of projections onto C to be calculate at each iteration.

In the same spirit as the previous cited algorithms and under the same convergence assumptions (continuity and the pseudomonotonicity of F), we have proposed in [7] an algorithm with a new step size satisfying a certain condition. This last one ensures a faster convergence towards a solution. Our preliminary and comparative computational experience using this algorithm for many classes of $VIP(F,C)$ was very encouraging. This is particularly true for the class of linear constraints which is considered to be more difficult than the other cases of C. This is clear in examples as nonnegative orthant and boxes where the projection is given by an explicit expression. In this paper, we extend the applicability of the algorithm introduced in [7] to another class of (VIP) where we consider the constraints to be nonlinear. The problem that arises in this case is how to compute the projection onto the constraints set? We know that if C is given by nonlinear constraints, the expression of the projection is given explicitly only in the case of spheres. To overcome this numerical obstacle, we propose to approximate C at each iteration in the neighbourhood of x^k by the set of the linearized constraints at this point. Using this well-known mathematical programming technique, we can transform the computation of the projection onto the nonlinear constraints to their computation onto the linear ones; the case where our algorithm has proven its efficiency. The principal idea to establish the convergence for this type of algorithms is to construct a hyper-plane H_k of normal $F(y^k)$ passing through y^k and separating strictly x^k and the solutions set. This hyper-plane is defined by $H_k = \{x \in \mathbb{R}^n : \langle F(y^k), x - y^k \rangle = 0\}$. The computation of y^k depends essentially on the first step size α_k which is determined in turn by a line search step of Armijo type ensuring the separation condition. The next iterate x^{k+1} is then, the projection of the vector $x^k - \lambda_k F(y^k)$ onto C where the second step size λ_k is determined according to chosen algorithm. If we denote by $D_k = \{x \in \mathbb{R}^n : \langle F(y^k), x - y^k \rangle \leq 0\}$, the half-space containing the solutions set $\mathcal{S}$ and whose boundary is H_k, we remark that in all algorithms cited above, the iterate x^{k+1} belongs to the complement of D_k. Peculiarly, in the best case which coincides with Solodov's algorithm, this iterate is on the boundary H_k. To be closer to the set $\mathcal{S}$, we have proposed in [7] to introduce a condition that x^{k+1} must belong to D_k. This modification allowed us to establish a new version of projection algorithm with faster convergence compared to the ones of the same family. In this study, we are able to demonstrate that this algorithm which can be applied to any class of $VIP(F, C)$, converges under a condition that is weaker than the pseudomonotonicity as we will see later.

The rest of this paper is structured as follows. In Section 2, we summarize some basic notions and properties that are necessary in this work. In Section 3, we give a detailed presentation of the proposed algorithm, and we prove the global convergence results under weak assumptions on the mapping F. Furthermore, to test the efficiency of this new algorithm on the class of $VIP(F, C)$ with nonlinear constraints, we have carried out its implementation on many examples in Section 4. The results reported in this part are very promising. The last section draws an overall conclusion.

2. Preliminaries

$Proj_C : \mathbb{R}^n \longrightarrow C$, is the orthogonal projection mapping onto C, where $Proj_C(x) = \arg\min \{\|y - x\| \,/ y \in C\}$. We give some properties and results for projection mapping in the lemmas below.

LEMMA 2.1 ([16]). *Let D be a nonempty closed and convex subset in $\mathbb{R}^n$. Then, for any $x, y \in \mathbb{R}^n$ and $z \in D$, the following statements hold*

1. $\langle Proj_D(x) - x, z - Proj_D(x)\rangle \geq 0.$
2. $\|Proj_D(x) - z\|^2 \leq \|x - z\|^2 - \|Proj_D(x) - x\|^2.$

LEMMA 2.2 ([1, 3]). *Let D be a nonempty closed and convex subset in $\mathbb{R}^n$.*

1. *For $x, d \in \mathbb{R}^n$, and $\lambda \geq 0$, we define $x(\lambda) = Proj_D(x - \lambda d)$, then $\langle d, x - x(\lambda)\rangle$ is nondecreasing for $\lambda \geq 0$.*
2. *For $x \in D$, $d \in \mathbb{R}^n$, and $\lambda > 0$, we define*
$$\psi(\lambda) = min\left\{\|y - x + \lambda d\|^2 \,/\; y \in D\right\},$$
then
$$\psi'(\lambda) = 2\langle d, x(\lambda) - x + \lambda d\rangle.$$

It's well-known that

$$\overline{x} \in \mathcal{S} \;\; \textit{if and only if} \;\; -F(\overline{x}) \in N_C(\overline{x}). \tag{2}$$

We use $N_C(\overline{x})$ to denote the normal cone to C at $\overline{x}$. We have also,

$$\overline{x} \in \mathcal{S} \ \textit{ if and only if } \ \overline{x} = Proj_C\left(x - \beta F\left(x\right)\right), \forall \beta > 0\,. \tag{3}$$

The purpose of the following proposition is to show how we can transform the nonlinear constraints C to linear ones. This is done by linearizing them locally in the neighbourhood of the point $x \in C$.

Proposition 2.3. *Given any point $x \in \mathbb{R}^n$, we define the following polyhedral closed convex set*

$$C(x) = \{y \in \mathbb{R}^n : g_i(x) + \langle \nabla g_i(x), y - x\rangle \leq 0, i = 1, \dots, m\}. \tag{4}$$

Suppose that the Abadie's constraints qualification holds at $x \in C$, then the tangent cone to C at x, denoted by $T_C(x)$ can be written as

$$T_C(x) = \{v \in \mathbb{R}^n : \langle \nabla g_i(x), v\rangle \leq 0, i \in I(x)\}\,, \tag{5}$$

where $I(x)$ is the set of the active constraints g_i at x.
Then, we have

1. *$C \subset C(x)$.*

2. *$x \in C(x)$ if and only if $x \in C$.*

3. *Furthermore, we have $T_C(x) = T_{C(x)}(x)$.*

We recall that Abadie's constraints qualification holds at a feasible point x if

$$T_C(x) = L_C(x),$$

where $L_C(x)$ represents the linearized cone of C at x. Since the tangent cone is difficult to handle, we replace it with the linearized cone which is easier to use.

Proof. 1. We just use the definition of convexity of the functions g_i expressed by their gradients.

2. For the first implication, we have $x \in C(x)$, this means $g_i(x) + \langle \nabla g_i(x), x - x\rangle \leq 0, i = 1, \dots, m$, we get $x \in C$.

   The second implication follows from 1, in the particular case where $x \in C$.

3. It is not difficult to check this result. Indeed, we have $C(x) = \{y \in \mathbb{R}^n : h_i(y) = g_i(x) + \langle \nabla g_i(x), y - x\rangle \leq 0, i = 1, \dots, m\}$. Suppose that the constraints qualification holds at $y \in C(x)$, then $T_{C(x)}(y) = \{v \in \mathbb{R}^n : \langle \nabla h_i(y), v\rangle \leq 0, i \in J(y)\}$, where $J(y)$ is the set of the active constraints

h_i at y. Thus we get, $T_{C(x)}(y) = \{v \in \mathbb{R}^n : \langle \nabla g_i(x), v\rangle \leq 0, i \in J(y)\}$. (Because, when we calculate the gradient of ∇h_i with respect to y, we find $\nabla h_i(y) = \nabla g_i(x)$). In the particular case where $y = x \in C(x)$, $T_{C(x)}(x) = \{v \in \mathbb{R}^n : \langle \nabla g_i(x), v\rangle \leq 0, i \in J(x)\}$, and we have $h_i(x) = g_i(x)$, then $J(x) = I(x)$. Consequently, the active constraints in the two sets C and $C(x)$ at x are the same and at last we obtain the wanted result. □

Now, we state the assumptions which are necessary for our method.

$(A1)$ $\mathcal{S}$ is nonempty.

$(A2)$ For all $\overline{x} \in \mathcal{S}$, $y \in \mathbb{R}^n$, we have $\langle F(y), y - \overline{x}\rangle \geq 0$.

Remark 2.4. If the mapping F is pseudomonotone on $\mathbb{R}^n$ i.e.,

$$\langle F(x), y - x\rangle \geq 0 \Rightarrow \langle F(y), y - x\rangle \geq 0, \ \textit{for all} \ x, y \in \mathbb{R}^n.$$

If this property holds, then for any $\overline{x} \in \mathcal{S} \subset \mathbb{R}^n$, we have

$$\langle F(y), y - \overline{x}\rangle \geq 0, \text{ for all } y \in \mathbb{R}^n.$$

In our case, the assumption $(A2)$, is weaker than the condition of the pseudomonotonicity of F on $\mathbb{R}^n$.

3. Algorithm and its convergence

In this part, we will present our algorithm for solving $VIP(F, C)$ with nonlinear constraints. For $x \in C$ and $\beta > 0$, the projection residual function is defined as follows

$$r(x, \beta) = x - Proj_{C(\overline{x})}(x - \beta F(x)), \ \beta > 0.$$

It's clear that the algorithm terminates at a point x if this last one coincides with zeros of the function r

$$r(x, \beta) = 0, \ \forall \beta > 0. \tag{8}$$

Also, it is easy to see that (8) holds if and only if

$$-F(x) \in N_{C(x)}(x). \tag{9}$$

We denote by $N_{C(x)}(x)$ is the normal cone to the set $C(x)$ at the point x. The following proposition relates (8) to the problem (1), and shows that the algorithm terminates if and only if it finds a solution to this problem.

Algorithm 1 New projection algorithm

Require: $x^0 \in \mathbb{R}^n$, $\sigma, \gamma \in (0,1)$, $\beta > 0$ and let ϵ be a given tolerance.

$k = 0$, compute $z^k = Proj_{C(x^k)}\left(x^k - \beta F\left(x^k\right)\right)$.

while $\left\|r\left(x^k, \beta\right) = x^k - z^k\right\| > \epsilon$ **do**

$$y^k = (1 - \alpha_k)\, x^k + \alpha_k z^k,$$

where $\alpha_k = \gamma^j$ with j being the smallest nonnegative integer satisfying

$$\left\langle F(x^k - \gamma^j r(x^k, \beta)), r(x^k, \beta)\right\rangle \geq (\sigma/\beta)\left\|r(x^k, \beta)\right\|^2 \tag{6}$$

Let $x^{k+1} = Proj_{C(x^k)}\left(x^k - \lambda_k F\left(y^k\right)\right)$, where λ_k must satisfy

$$\left\langle F\left(y^k\right), Proj_{C(x^k)}\left(x^k - \lambda_k F\left(y^k\right)\right) - y^k\right\rangle \leq 0 \tag{7}$$

$k = k + 1$.

end while

PROPOSITION 3.1. *If $\overline{x}$ belongs to $\mathcal{S}$ and constraints qualification holds at this point, then (8) holds for any $\beta > 0$.*

Conversely, if a point $\overline{x} \in \mathbb{R}^n$ satisfies (8), then it is a solution for the problem (1).

Proof. First, we suppose that $\overline{x}$ is a solution for (1) at which constraints qualification holds. Using the first result or the second one of Proposition 1, $\overline{x} \in C$ implies that $\overline{x} \in C(\overline{x})$. From the third result of the same proposition, we have $T_C(\overline{x}) = T_{C(\overline{x})}(\overline{x})$, so it follows that $N_C(\overline{x}) = N_{C(\overline{x})}(\overline{x})$. Since $\overline{x}$ solves (1), the vector $-F(\overline{x}) \in N_C(\overline{x})$, then it belongs also to $N_{C(\overline{x})}(\overline{x})$. Then, $\overline{x}$ satisfies (9) and consequently (8).

For the converse direction, we suppose that $\overline{x}$ satisfies (8). This implies that $\overline{x}$ belongs to $C(\overline{x})$ since it is a projection onto $C(\overline{x})$, and from (9) we have $-F(\overline{x}) \in N_{C(\overline{x})}(\overline{x})$. The fact that $\overline{x} \in C(\overline{x})$ implies that $\overline{x} \in C$. Also, the fact that $C(\overline{x})$ contains C implies that $N_C(\overline{x})$ contains $N_{C(\overline{x})}(\overline{x})$, thus $-F(\overline{x}) \in N_C(\overline{x})$. Finally, from (2) we get $\overline{x} \in \mathcal{S}$. ☐

Next, we show that the line search step in Algorithm 1 is well-defined.

LEMMA 3.2. *If x^k is not a solution of problem (1), then there exists a smallest nonnegative integer j satisfying the line search step (7).*

Proof. Assume that for some k, the line search step (7) is not satisfied for any nonnegative integer j, i.e., that

$$\left\langle F(x^k - \gamma^j r(x^k, \beta)), r(x^k, \beta)\right\rangle < (\sigma/\beta)\left\|r(x^k, \beta)\right\|^2 . \tag{10}$$

Combining the fact that F is continuous and taking the limit in (10) as j goes to ∞, so since $\gamma \in (0,1)$, we get

$$\left\langle F(x^k), r(x^k,\beta)\right\rangle \leq (\sigma/\beta)\left\|r(x^k,\beta)\right\|^2 . \tag{11}$$

On the other hand, using the first result of Lemma 2.1, we obtain

$$\left\langle P_C\left(x^k-\beta F(x^k)\right)-\left(x^k-\beta F(x^k)\right), x^k-P_C\left(x^k-\beta F(x^k)\right)\right\rangle \geq 0,$$

which implies

$$\beta\left\langle t^k, r\left(x^k,\beta\right)\right\rangle \geq \left\|r\left(x^k,\beta\right)\right\|^2 . \tag{12}$$

From (11) and (12), we get that

$$(1/\beta)\left\|r\left(x^k,\beta\right)\right\|^2 \leq \left\langle F(x^k), r\left(x^k,\beta\right)\right\rangle \leq (\sigma/\beta)\left\|r\left(x^k,\beta\right)\right\|^2,$$

then,

$$(1/\beta)(1-\sigma)\left\|r\left(x^k,\beta\right)\right\|^2 \leq 0.$$

We get a contradiction because $\beta>0$, $0<\sigma<1$ and $\left\|r\left(x^k,\beta,t^k\right)\right\|>0$. □

In the lemma below, we will ensure that the hyper-plane H_k separates effectively the iterate x^k from the set $\mathcal{S}$.

LEMMA 3.3. *Assume that F satisfies $(A2)$ on C and $\left\{x^k\right\}$ be a sequence generated by Algorithm 1. If $D_k=\left\{x\in\mathbb{R}^n : \left\langle F(y^k), x-y^k\right\rangle \leq 0\right\}$ is the half-space whose boundary is H_k, then $\mathcal{S}\subset D_k$ and $x^k\notin D_k$.*

Proof. From the expression of y^k and the line search step used in Algorithm 1, we have

$$\begin{aligned}
&\langle F(y^k), x^k-z^k\rangle = \langle F(x^k-\gamma^j r(x^k,\beta)), r(x^k,\beta)\rangle \geq \sigma\|r(x^k,\beta)\|^2>0\\
&\qquad\qquad\qquad\qquad\qquad\text{(because } x^k \text{ isn't a solution for } VIP(F,C)\text{)},\\
&\langle F(y^k), \alpha_k x^k-\alpha_k z^k\rangle > 0\,,\\
&\langle F(y^k), x^k-((1-\alpha_k)x^k+\alpha_k z^k)\rangle > 0\,,\\
&\langle F(y^k), x^k-y^k\rangle > 0.
\end{aligned}$$

On the other hand, the second part of this separation follows directly from $(A2)$. □

In what follows, we will give the theoretical analysis of the convergence of this algorithm under the assumptions $(A1)$ and $(A2)$.

In this algorithm, if $\left\|r\left(x^k,\beta\right)\right\|=0$, then x^k is a solution of $VIP\left(F,C\right)$.

Otherwise, for any $\overline{x}\in\mathcal{S}$, and according to the iterative scheme of Algorithm 1, we have

$$\begin{aligned}\left\|x^{k+1}-\overline{x}\right\|^{2} &= \left\|Proj_{C(x^k)}\left(x^k-\lambda_k F\left(y^k\right)\right)-\overline{x}\right\|^{2}\\ &\leq \left\|x^k-\overline{x}-\lambda_k F\left(y^k\right)\right\|^{2}-\left\|x^k-x^{k+1}-\lambda_k F\left(y^k\right)\right\|^{2}\\ &\leq \left\|x^k-\overline{x}\right\|^{2}+\lambda_k^2\left\|F\left(y^k\right)\right\|^{2}-2\lambda_k\left\langle F\left(y^k\right),x^k-y^k\right\rangle\\ &\quad -\left\|x^k-x^{k+1}-\lambda_k F\left(y^k\right)\right\|^{2},\end{aligned}$$

where the first inequality follows from the second result of Lemma 2.1, and for the second inequality we use the assumption $(A2)$.

We apply the same idea used in [7], but we will apply it locally on the linearized constraints on the neighbourhood of the iterate x^k. We replace at each iteration, the projection region C by the set $C(x^k)$.

For any $\lambda \geq 0$, we define

$$x^{k+1}=x\left(\lambda\right)=Proj_{C(x^k)}\left(x^k-\lambda F\left(y^k\right)\right),$$

and the function

$$\phi_k\left(\lambda\right)=2\lambda\left\langle F\left(y^k\right),x^k-y^k\right\rangle+\left\|x^k-x^k\left(\lambda\right)-\lambda F\left(y^k\right)\right\|^{2}-\lambda^2\left\|F\left(y^k\right)\right\|^{2}, \quad (13)$$

where its derivative is

$$\phi_k'\left(\lambda\right)=2\left\langle F\left(y^k\right),x^k\left(\lambda\right)-y^k\right\rangle. \quad (14)$$

Using the second result of Lemma 2.2, we can find easily the expression of ϕ_k'.

We denote by λ_{k2} the step size associated to the Isuem's algorithm [10], given as follows

$$\lambda_{k2}=\frac{\langle F(y^k),x^k-y^k\rangle}{\|F(y^k)\|^2},$$

and λ_{k1} the step size associated to the Solodov's algorithm [15], where it is proven in [17] that

$$x^{k+1}=Proj_C\left(x^k-\lambda_{k1}F\left(y^k\right)\right)=Proj_{C\cap H_k}\left(x^k-\lambda_{k2}F\left(x^k\right)\right),$$

and we have that $\lambda_{k1}>\lambda_{k2}$.

Also from [17], we recall that ϕ_k is a positive function for all values $\lambda \in [0,\lambda_{k1}]$. We can see the same thing for the function ϕ_k' i.e.,

$$\phi_k'\left(\lambda\right)=2\left\langle F\left(y^k\right),x^k\left(\lambda\right)-y^k\right\rangle\geq 0,\ for\ \lambda\in\left[0,\lambda_{k1}\right], \quad (15)$$

and the function ϕ_k reaches its maximum on $[0,\lambda_{k1}]$ at λ_{k1}.

Then, for ϕ_k', we have

$$\phi_k'(\lambda_{k1}) = 2\left\langle F\left(y^k\right), x^k(\lambda_{k1}) - y^k\right\rangle = 0,$$

and we get

$$\left\langle F\left(y^k\right), x^k(\lambda_{k1}) - y^k\right\rangle = 0\,. \tag{16}$$

Geometrically, we see that the iterate x^{k+1} computed by the algorithm of Solodov is $x^{k+1} = Proj_{C(x^k)}\left(x^k - \lambda_{k1}F\left(y^k\right)\right) = x(\lambda_{k1})$ is on the boundary of D_k (belongs to H_k).

For our new algorithm, the step size must satisfy the following inequality

$$\left\langle F\left(y^k\right), Proj_{C(x^k)}\left(x^k - \lambda_k F\left(y^k\right)\right) - y^k\right\rangle \leq 0\,. \tag{17}$$

This condition ensures the properties below:

The iterate x^{k+1} computed by Algorithm 1 given by

$$x^{k+1} = Proj_{C(x^k)}\left(x^k - \lambda_k F\left(y^k\right)\right) = x(\lambda_k)$$

belongs to D_k and

$$\phi_k'(\lambda_k) = 2\left\langle F\left(y^k\right), x(\lambda_k) - y^k\right\rangle \leq 0.$$

Consequently, the step size $\lambda_k > \lambda_{k1}$, ($\lambda_k \notin [0, \lambda_{k1}]$) since the function ϕ_k' is positive on $[0, \lambda_{k1}]$.

To clarify the last characterization of λ_k, we give the proposition below to show that the algorithm corresponding to this step size guarantees a large decrease of the generated sequence towards the solutions set.

PROPOSITION 3.4. *Let $x^k(\lambda_{k1})$ and $x^k(\lambda_k)$ be the following iterates corresponding to the iteration $(k+1)$ computed by the Solodov's algorithm [15] and Algorithm 1, respectively. Then, we have*

1. $\left\|x^k - x^k(\lambda_k)\right\|^2 - \left\|x^k - x^k(\lambda_{k1})\right\|^2 \geq 0.$

2. *Furthermore, if*

$$\left\|x^k - x^k(\lambda_k)\right\|^2 - \left\|x^k - x^k(\lambda_{k1})\right\|^2 \geq 2\lambda_k\left\langle F(y^k), y^k - x^k(\lambda_k)\right\rangle,$$

then $\phi_k(\lambda_k) \geq \phi_k(\lambda_{k1})\,.$

Proof. In this paper, we are able to prove the first point of this proposition without needing the condition given in the second one as in the reference [7]. Indeed, we have

$$\begin{aligned}\|x^k - x^k(\lambda_k)\|^2 &= \|x^k - x^k(\lambda_k) - x^k(\lambda_{k1}) + x^k(\lambda_{k1})\|^2\\ &= \|x^k - x^k(\lambda_{k1})\|^2 + \|x^k(\lambda_k) - x^k(\lambda_{k1})\|^2\\ &\quad - 2\langle x^k - x^k(\lambda_{k1}), x^k(\lambda_k) - x^k(\lambda_{k1})\rangle\\ &= \|x^k - x^k(\lambda_{k1})\|^2 + \|x^k(\lambda_k) - x^k(\lambda_{k1})\|^2\\ &\quad - 2\langle x^k - \lambda_{k1}F(y^k) + \lambda_{k1}F(y^k) - x^k(\lambda_{k1}), x^k(\lambda_k) - x^k(\lambda_{k1})\rangle\\ &= \|x^k - x^k(\lambda_{k1})\|^2 + \|x^k(\lambda_k) - x^k(\lambda_{k1})\|^2\\ &\quad + 2\langle x^k(\lambda_{k1}) - (x^k - \lambda_{k1}F(y^k)), x^k(\lambda_k) - x^k(\lambda_{k1})\rangle\\ &\quad + 2\lambda_{k1}\langle F(y^k), y^k - x^k(\lambda_k)\rangle + 2\lambda_{k1}\langle F(y^k), x^k(\lambda_{k1}) - y^k\rangle\\ &\geq \|x^k - x^k(\lambda_{k1})\|^2.\end{aligned}$$

We get directly the last inequality using the first result of Lemma 2.1 and the two inequalities (16), (17) satisfied by the step sizes λ_{k1}, λ_k respectively.

For the second point, we use the definition of the function ϕ_k

$$\begin{aligned}\phi_k(\lambda_k) - \phi_k(\lambda_{k1}) &= 2\lambda_k \left\langle F\left(y^k\right), x^k - y^k\right\rangle + \left\|x^k - x^k(\lambda_k) - \lambda_k F\left(y^k\right)\right\|^2\\ &\quad - \lambda_k^2 \left\|F\left(y^k\right)\right\|^2 - 2\lambda_{k1}\left\langle F\left(y^k\right), x^k - y^k\right\rangle\\ &\quad - \left\|x^k - x^k(\lambda_{k1}) - \lambda_{k1}F\left(y^k\right)\right\|^2 + \lambda_{k1}^2\left\|F\left(y^k\right)\right\|^2\\ &= 2\lambda_k\left\langle F\left(y^k\right), x^k - y^k\right\rangle + \left\|x^k - x^k(\lambda_k)\right\|^2\\ &\quad + \lambda_k^2\left\|F\left(y^k\right)\right\|^2 - 2\lambda_k\left\langle F\left(y^k\right), x^k - x^k(\lambda_k)\right\rangle\\ &\quad - \lambda_k^2\left\|F\left(y^k\right)\right\|^2 - 2\lambda_{k1}\left\langle F\left(y^k\right), x^k - y^k\right\rangle\\ &\quad - \left\|x^k - x^k(\lambda_{k3})\right\|^2 - \lambda_{k1}^2\left\|F\left(y^k\right)\right\|^2\\ &\quad + 2\lambda_{k1}\left\langle F\left(y^k\right), x^k - x^k(\lambda_{k1})\right\rangle + \lambda_{k1}^2\left\|F\left(y^k\right)\right\|^2\\ &= \left\|x^k - x^k(\lambda_k)\right\|^2 - \left\|x^k - x^k(\lambda_{k1})\right\|^2\\ &\quad - 2\lambda_k\left\langle F\left(y^k\right), y^k - x^k(\lambda_k)\right\rangle\\ &\quad - 2\lambda_{k1}\left\langle F\left(y^k\right), x^k(\lambda_{k1}) - y^k\right\rangle.\end{aligned}$$

Using the given condition in second point and the inequality (16), we obtain the desired result. □

Now, let us give the proposition and the theorem establishing the convergence of Algorithm 1.

PROPOSITION 3.5. *Let* $\left\{x^k\right\}$ *be the sequence generated by Algorithm 1 and suppose that the assumptions* $(A1)$*and* $(A2)$ *are satisfied, then*

1. *The sequence* $\{x^k\}$ *is bounded.*

2. $\lim\limits_{k\to\infty} \langle F(y^k), x^k - y^k\rangle = 0.$

3. *If a cluster point of the sequence* $\{x^k\}$ *belongs to* $\mathcal{S}$, *then* $\{x^k\}$ *converges to a solution in* $\mathcal{S}$.

Proof. Let be λ_{k2} the step size used in the algorithm of Iusem [9], we consider here that the projection is computed on the set $C(x^k)$ at each iteration. From Proposition 3.4, we have $\phi_k(\lambda_k) \geq \phi_k(\lambda_{k1})$ and consequently $\phi_k(\lambda_k) \geq \phi_k(\lambda_{k2})$. This is because the function ϕ_k reaches its maximum on $[0, \lambda_{k1}]$ at λ_{k1} and we have $\lambda_{k2} \in]0, \lambda_{k1}]$.

1. Furthermore, for all $\overline{x} \in \mathcal{S}$ we have from what precedes the following inequality

$$
\begin{aligned}
\left\|x^{k+1} - \overline{x}\right\|^2 &\leq \left\|x^k - \overline{x}\right\|^2 - \phi_k(\lambda_k) \\
&\leq \left\|x^k - \overline{x}\right\|^2 - \phi_k(\lambda_{k2}) \\
&= \left\|x^k - \overline{x}\right\|^2 + \lambda_{k2}^2 \left\|F(y^k)\right\|^2 - 2\lambda_{k2} \left\langle F(y^k), x^k - y^k\right\rangle \\
&\quad - \left\|x^k - x^k(\lambda_{k2}) - \lambda_{k2} F(y^k)\right\|^2 \\
&= \left\|x^k - \overline{x}\right\|^2 + \frac{\left\langle F(y^k), x^k - y^k\right\rangle^2}{\|F(y^k)\|^2} - 2\frac{\left\langle F(y^k), x^k - y^k\right\rangle^2}{\|F(y^k)\|^2} \\
&\quad - \left\|x^k - x^k(\lambda_{k2}) - \lambda_{k2} F(y^k)\right\|^2 \\
&= \left\|x^k - \overline{x}\right\|^2 - \lambda_{k2}^2 \left\|F(y^k)\right\|^2 - \left\|x^k - x^k(\lambda_{k2}) - \lambda_{k2} F(y^k)\right\|^2 \\
&\leq \left\|x^k - \overline{x}\right\|^2 - \lambda_{k2}^2 \left\|F(y^k)\right\|^2 .
\end{aligned}
$$

It's obvious that the sequence $\left\{\left\|x^k - \overline{x}\right\|^2\right\}$ is nonincreasing. In addition, it is positive, then it converges and correspondingly $\{x^k\}$ is bounded.

2. Using the first point, we deduce that the sequence $\left\{\lambda_{k2}^2 \left\|F(y^k)\right\|^2\right\}$ converges to 0, when $k \to \infty$.

It yields

$$\lim_{k\to\infty} \frac{\left\langle F(y^k), x^k - y^k\right\rangle}{\|F(y^k)\|} = \lim_{k\to\infty} \lambda_{k2} \left\|F(y^k)\right\| = 0.$$

Since $\{x^k\}$ is bounded, the same for $\{y^k\}$ and F is a continuous mapping, then the sequence $\{F(y^k)\}$ is also bounded.

Using the boundedness of $\{F(y^k)\}$, we obtain $\lim\limits_{k\to\infty} \left\langle F(y^k), x^k - y^k\right\rangle = 0.$

3. Let $\overline{x}$ a cluster point of the sequence $\left\{x^k\right\}$ belonging to $\mathcal{S}$ and $\left\{x^{i_k}\right\}$ a subsequence $\left\{x^k\right\}$ such that $\lim_{k\to\infty} x^{i_k} = \overline{x}$, then $\lim_{k\to\infty} \left\|x^{i_k} - \overline{x}\right\| = 0$.

On the other hand, we have $\overline{x} \in \mathcal{S}$ and the whole sequence $\left\{\left\|x^k - \overline{x}\right\|\right\}$ converges to some limit from the first point, but since one of its subsequences converges to 0, we then get $\lim_{k\to\infty} \left\|x^k - \overline{x}\right\| = 0$, i.e., $\lim_{k\to\infty} x^k = \overline{x}$. □

THEOREM 3.6. *Suppose that F is a continuous mapping and the assumptions $(A1)$ and $(A2)$ are satisfied. Then, the sequence generated by Algorithm 1 converges to a solution of $VIP(F, C)$.*

Proof. The parameter β in Algorithm 1 is taken strictly positive and not only in $]0, 1]$ as in [7]. Numerically, this suggestion allows us to have a large interval for the values of this parameter in order to improve the results obtained by Algorithm 1. The steps of the convergence's proof remain the same as in [7], however, at each iteration we must change the projection onto the set C by the projection onto the set $C(x^k)$. □

REMARK 3.7. We notice that we are not sure that the sequence $\{x^k\}$ is entirely contained in C in fact its terms x^k are computed as projection onto $C(x^{k-1})$, however the solution $\overline{x}$ is certainly in C from (8) and the second point of Proposition 2.3. The same holds for the sequence $\{y^k\}$, which is not entirely contained in C, because its terms are computed as a combination of $x^k \in C(x^{k-1})$ and $z^k \in C(x^k)$. For this reason, in assumption $(A2)$, we have taken $y \in \mathbb{R}^n$.

3.1. Determination of the iterate x^{k+1}

As proposed in [7], x^{k+1} can be calculated as a convex combination of $x^k(\lambda_{k1}) \in C(x^k)$ and $z^k \in C(x^k)$,

$$x^{k+1} = \theta x^k(\lambda_{k1}) + (1-\theta) z^k, \qquad (\theta \in [0, 1]).$$

From this form, we see that x^{k+1} belongs to $C(x^k)$ and satisfies the condition (17), indeed we have at first

$$\begin{aligned}\langle F(y^k), z^k - y^k\rangle &= (1-\alpha_k)\langle F(y^k), z^k - x^k\rangle \\ &\leq -(1-\alpha_k)\sigma\|r(x^k,\beta)\|\,.\end{aligned}$$

Thus, we get

$$\langle F(y^k), z^k - y^k\rangle < 0\,. \tag{18}$$

We obtain,

$$\begin{aligned}&\langle F(y^k), \theta x^k(\lambda_{k1}) + (1-\theta) z^k - y^k\rangle \\ &\quad = \langle F(y^k), \theta x^k(\lambda_{k1}) + (1-\theta) z^k - (\theta + (1-\theta)) y^k\rangle \\ &\quad = \theta\langle F(y^k), x^k(\lambda_{k1}) - y^k\rangle + (1-\theta)\langle F(y^k), z^k - y^k\rangle \le 0.\end{aligned}$$

The last equality follows from (18) and the fact that λ_{k1} satisfies (16).

We note that Solodov's iterate can be obtained as a particular case of our iteration, when $\theta = 1$.

4. Computational experience

In this section, we provide some numerical results of Algorithm 1 denoted as $Alg1$ and tested on a set of problems described below. All codes are written in Matlab 2015 and run on a personal computer (with the following characteristics: Windows 11 professional, Intel(R) Core (TM)i7-9850CPU @2.60GHz and RAM 16,0 Go). For the $Alg1$, we choose $\gamma = 0.9, \sigma = 0.9$, we use also $\|r(x^k, \beta)\| \le 10^{-6}$ as the stopping criteria for all our numerical tests. To evaluate the effectiveness of the proposed method, a comparative numerical study was carried out with the method presented in [6]. This method is introduced by Ge & al. where its principle is to reformulate the variational inequality problem as a parameterized smooth system of equations based on Fischer-Burmeister function. The corresponding linear system is approximately solved at each iteration by an inexact Newton-GMRES method which aims to reduce the computational cost. The authors have been able to establish the global and local quadratic convergence of their algorithm under some mild conditions.

For their associated algorithm denoted as $Alg2$, we choose $\mu = 0.2, \gamma = 0.001$, and we denote also by $Iter$, the number of iterations, T, the computation time.

It is not easy to find proper test examples for the variational inequalities with nonlinear constraints. Hence, we applied our algorithm on some modified test examples available in the literature.

To better understand the inexact Newton method which includes GMRES (Generalized Minimal Residual method), used in particular to solve large-scale problems, let us first recall Newton's method.

Consider the following system of equations

$$F(x) = 0\,.$$

The iterative scheme of Newton's method, is given by

$$\nabla F(x^k) \triangle_x^k + F(x^k) = 0 \text{ and } x^{k+1} = x^k + \triangle_x^k.$$

where $\bigtriangledown F(x^k)$ is the jacobian matrix at the iterate x^k.

The basic idea of the inexact Newton-GMRES method is the same as for the classical one, the only difference is when solving the linear system, the vector of Newton's direction $\triangle_x^k$ is an approximation of the solution of this system under some conditions. The accuracy level of this approximation is controlled by the so called forcing term which links the norm of residual vector to the norm of the mapping F at the current iteration. Since the inexact direction $\triangle_x^k$ of the combined method is still a descent direction, a sufficient reduction of the merit function can be always guaranteed.

EXAMPLE 4.1 ([14]). Let $F : \mathbb{R}^2 \longrightarrow \mathbb{R}^2$

$$F(x) = \begin{pmatrix} 3x_1 + 4x_2 + 5 \\ 2x_1 + 5x_2 - 4 \end{pmatrix},$$

and let the set C be

$$C = \left\{ x \in \mathbb{R}^4 : \begin{array}{r} x_1^2 + 4x_2^2 \leq 4 \\ 2x_1^2 + x_2^2 \leq 6 \\ 2x_1 + x_2 \geq -1 \end{array} \right\}.$$

The solution of this problem is $\overline{x} = (-0.9412, 0.8824)$.

	*Alg*1		*Alg*2	
x^0	*Iter*	T	*Iter*	T
$(0, 0)$	2	0.0131	36	0.0056
$(1, 1)$	3	0.0148	40	0.0075
$(-1, -1)$	4	0.0355	41	0.0080
$(10, 10)$	4	0.0234	47	0.0092
$(100, 100)$	9	0.0633	1471	0.2475

Table 1: Numerical results for Example 4.1 with *Alg*1 and *Alg*2 with different initial points.

EXAMPLE 4.2 ([5]). The mapping $F : \mathbb{R}^3 \longrightarrow \mathbb{R}^3$ is as follows

$$F(x) = \begin{pmatrix} 2x_1 + 0.2{x_1}^3 - 0.5x_2 + 0.1x_3 - 4 \\ -0.5x_1 + x_2 + 0.1{x_2}^3 + 0.5 \\ 0.5x_1 - 0.2x_2 + 2x_3 - 0.5 \end{pmatrix}$$

and

$$C = \left\{ x \in \mathbb{R}^3 \,:\, 0.6x_1^2 + 0.4x_2^2 + x_3^2 \leq 1 \right\}.$$

The solution of this problem is $\overline{x} = (1, 0, 0)$.

Since the number of iterations obtained for this example is quite small, then we can see how the sequence of the iterates $\{x^k\}$ converges towards the solution

	*Alg*1			
k	x_1^k	x_2^k	x_3^k	$\left\|r\left(x^k,\beta\right)\right\|$
1	1.5246	0.4456	0.5495	0.6065
2	1.1699	0.2592	−0.1130	0.0644
3	1.0188	0.1034	0.0158	0.0060
4	1.0011	0.0304	−0.0005	5.1105×10^{-4}
5	1.0001	0.0082	0.0021	4.1359×10^{-5}
6	1.0000	0.0022	−0.0000	2.6660×10^{-6}
7	1.0000	0.0006	0.0002	2.1154×10^{-7}

Table 2: Numerical results for Example 4.2 with *Alg*1 and $x^0 = (0,1,1)^T$.

	*Alg*2			
k	x_1^k	x_2^k	x_3^k	$\left\|H\left(\mu^k,w^k\right)\right\|$
1	1.9655	0.6384	−0.1170	6.2950
2	1.7018	0.3748	−0.1120	2.5670
3	1.2760	0.0401	−0.0354	0.5497
4	1.0906	0.0207	−0.0097	0.1809
5	1.0174	0.0029	−0.0017	0.0417
6	1.0022	0.0003	−0.0002	0.0054
7	1.0002	0.0000	−0.0000	5.5887×10^{-4}
8	1.0000	0.0000	−0.0000	5.6082×10^{-5}
9	1.0000	0.0000	−0.0000	5.6102×10^{-6}
10	1.0000	0.0000	−0.0000	5.6104×10^{-7}

Table 3: Numerical results for Example 4.2 with *Alg*2 and $x^0 = (0,1,1)^T$.

$\overline{x}$, and also see how the error decreases towards 0 for both algorithms *Alg*1 and *Alg*2. We consider the same initial point chosen in [6], $x^0 = (0,1,1)$.

Next, we give in Table 4, the numerical results for Example 4.2 with *Alg*1 and *Alg*2 with different initial points to see the choice influence of this point.

Example 4.3. This example is derived from [4], where the original problem is an optimization one. We give its form of variational inequalities by writing the corresponding optimality conditions, i.e., let $F : \mathbb{R}^4 \longrightarrow \mathbb{R}^4$

$$F(x) = \begin{pmatrix} 2x_1 - 5 \\ 2x_2 - 5 \\ 4x_3 - 21 \\ 2x_4 + 7 \end{pmatrix},$$

x^0	*Alg*1 *Iter*	*Alg*1 T	*Alg*2 *Iter*	*Alg*2 T
$(0,0,0)$	4	0.0345	10	0.0028
$(1,1,1)$	5	0.0656	10	0.0032
$(-1,-1,-1)$	5	0.0353	10	0.0030
$(10,10,10)$	12	0.1446	12	0.0954

Table 4: Numerical results for Example 4.2 with *Alg*1 and *Alg*2 with different initial points.

and let the set C be

$$C = \left\{ x \in \mathbb{R}^2 : \begin{array}{l} x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 \leq 8 \\ x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 \leq 10 \\ 2x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 \leq 5 \end{array} \right\}.$$

The solution of this problem is $\overline{x} = (0, 1, 2, -1)$.

Similarly, we test both algorithms *Alg*1 and *Alg*2 for this example from the initial point $(0, 0, 0, 0)$, then we get the results shown in the Tables 5 and 6.

k	x_1^k	x_2^k	x_3^k	x_4^k	$\left\Vert r\left(x^k, \beta\right)\right\Vert$
	*Alg*1				
1	0.5004	0.5004	2.0968	−0.7006	0.3826
2	0.0590	0.8999	2.1593	−0.8557	0.0416
3	−0.0283	0.9842	2.0339	−0.9718	0.0023
4	−0.0017	0.9966	2.0029	−0.9977	1.9127×10^{-5}
5	0.0004	1.0004	1.9996	−1.0004	4.7834×10^{-7}

Table 5: Numerical results for Example 4.3 with *Alg*1 and $x^0 = (0, 0, 0.0)^T$

.

We give also in Table 7, the numerical results for Example 4.3 with *Alg*1 and *Alg*2 and with different initial points to see the influence of this point choice.

In order to more evaluate the efficiency of our algorithm, we give in the last part of this section two examples where the dimension of the problems is from 100 to 1000.

Example 4.4. This problem is derived from [12]. The original problem is based on the boxes constraints and in our case we have changed them by the nonlinear constraints written below.

	*Alg*2				
k	x_1^k	x_2^k	x_3^k	x_4^k	$\left\|H\left(\mu^k, w^k\right)\right\|$
1	2.4622	2.4769	5.1971	−3.4631	276.5066
2	1.8530	1.8430	4.7238	−2.7727	70.3578
3	0.4009	1.3817	3.1040	−1.1109	16.1805
4	0.2116	1.0152	2.1546	−1.0333	3.2871
5	−0.0106	1.0279	2.0411	−0.9511	0.7829
6	0.0003	0.9991	2.0009	−1.0007	0.0337
7	0.0000	1.0000	2.0000	−1.0000	3.4910×10^{-4}
8	0.0000	1.0000	2.0000	−1.0000	3.4924×10^{-6}
9	0.0000	1.0000	2.0000	−1.0000	3.4924×10^{-8}

Table 6: Numerical results for Example 4.3 with $Alg2$ and $x^0 = (0, 0, 0.0)^T$.

	*Alg*1		*Alg*2	
x^0	*Iter*	T	*Iter*	T
$(0, 0, 0, 0)$	5	0.0245	9	0.0022
$(1, 1, 1, 1)$	6	0.0455	8	0.0018
$(-1, -1 - 1, -1)$	6	0.0442	10	0.0027
$(10, 10, 10, 10)$	10	0.0693	12	0.0031

Table 7: Numerical results for Example 4.3 with $Alg1$ and $Alg1$ with different initial points.

Let $F(x) = Ax + b$ where A is the following $(n \times n)$ nonsymmetric matrix

$$A = \begin{pmatrix} 4 & -1 & & & & \\ -1 & 4 & -1 & & & \\ & & 4 & -1 & & \\ & & & \ddots & \ddots & \\ & & & & 4 & -1 \\ & & & & & 4 \end{pmatrix}, \quad b = \begin{pmatrix} -1 \\ -1 \\ -1 \\ \vdots \\ -1 \\ -1 \end{pmatrix},$$

and

$$C = \left\{ x \in \mathbb{R}^n : \begin{array}{l} \sum_{i=1}^n (x_i - 0.3)^2 - 25 \leq 0 \\ \sum_{i=1}^n (x_i - 0.5)^2 - 22 \leq 0 \end{array} \right\}.$$

All the results are obtained by taking the same initial point $x^0 = (1, 1, \cdots, 1)^T$ for different sizes.

Example 4.5 ([6]). Similarly as in the previous example, the author has transformed the original problem based on linear constraints given in [1] by adding some nonlinear constraints. In this example, $F(x) = Ax + b$ where A and b

	*Alg*1		*Alg*2	
n	*Iter*	T	*Iter*	T
100	6	0.0621	10	0.0781
200	6	0.1131	11	0.1599
400	7	0.2755	12	0.3879
600	8	0.5765	12	0.7619
800	6	0.6744	12	0.9597
1000	7	0.8897	11	1.2459

Table 8: Numerical results for Example 4.4 obtained by *Alg*1, *Alg*2 and with different sizes.

are given as follows

$$A = \begin{pmatrix} 4 & -2 & & & & \\ 1 & 4 & -2 & & & \\ & 1 & 4 & -2 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & -2 \\ & & & & 1 & 4 \end{pmatrix}, \quad b = \begin{pmatrix} -1 \\ -1 \\ -1 \\ \vdots \\ -1 \\ -1 \end{pmatrix},$$

and

$$C = \left\{ x \in \mathbb{R}^n, i = 1, \dots, n \;:\; \begin{array}{c} -x_i - 10 \leq 0 \\ -100 + x_i^2 \leq 0 \end{array} \right\}.$$

In this case also, we used the same initial point $x^0 = (5, 5, \cdots, 5)^T$ for all numerical tests of this example.

	*Alg*1		*Alg*2	
n	*Iter*	T	*Iter*	T
100	8	0.1349	8	0.3071
200	9	0.2860	8	0.4547
400	9	0.6030	9	1.8581
600	9	1.0601	9	4.3098
800	9	1.4200	9	7.0898
1000	9	1.8655	9	10.4849

Table 9: Numerical results for Example 4.5 obtained by *Alg*1, *Alg*2 and with different sizes.

4.1. Comments

For the Examples 4.1, 4.2 and 4.3, we notice that the number of iterations obtained by *Alg*1 is better than that of *Alg*2, but the computation time recorded for $Alg2$ is lower than that of $Alg1$. Concerning large-scale examples (Example 4.4 and 4.5), it is clear that $Alg1$ and $Alg2$ have very close number of iterations, but there is a significant reduction in the computation time obtained by $Alg1$ compared with $Alg2$. We observe also that $Alg1$ converges to a solution whatever the initial point is (feasible or not). On the other hand, for $Alg2$ which is characterized by a global and quadratic convergence; the number of iterations exceeds 1400 to be able to achieve an approximate solution for certain initial points. The numerical performance of our algorithm to solve different classes of (VIP) notably with nonlinear constraints, is largely due to the relevant role played by the imposed condition (17) and the choice of $\beta > 0$. This is not limited to only in the interval $]0, 1]$ as it is case in the other algorithms proposed in [7, 10, 15, 17]. We recall that these two assumptions represent the main algorithmic modifications introduced in this work.

5. Conclusion

Based on the framework of last projection methods as [10, 15, 17], we have proposed a new algorithm of the same type and we have tested it for variational inequalities with nonlinear constraints. Under the condition $(A2)$ that is weaker than the pseudomonotonicity of the underling mapping, we have established the theoretical results to prove the global convergence for this algorithm. Furthermore, we have also presented some preliminary numerical results which show and confirm the efficiency of our method for this class of problems. As future perspective, we plan to analyze new procedures to determine the step size λ_k in order to improve further the convergence of the proposed algorithm.

References

[1] B.H. Ahn, *Iterative methods for linear complementarity problems with upper-bounds on primary variables*, Math. Program. **26** (1983), 295–315.

[2] Ferris M. C. and Pang J. S, *Complementarity and variational problems*, no. 52, State of Art, SIAM Publications, Philadelphia, 1997.

[3] P. H. Calamai and J. J. Moré, *Projected gradient methods for linearly constrained problems*, Math. Program. **39** (1987), 93–116.

[4] C. Charalambous, *Nonlinear least pth optimization and nonlinear programming*, Math. Program. **12** (1977), 195–225.

[5] M. Fukushima, *Relaxed projection method for variational inequalities*, Math. Program. **35** (1986), 58–70.

[6] Z. Ge, Q. Ni, and X. Zhang, *A smoothing inexact Newton method for variational inequalities with nonlinear constraints*, J. Inequal. Appl. (2017), 1–12.

[7] H. Grar and D. Benterki, *New effective projection method for variational inequalities problem*, RAIRO Oper. Res. **49** (2015), 805–820.

[8] P.T. Harker and J.S Pang, *Finite dimensional variational inequality and nonlinear complementarity problem: A survey of theory, algorithms and applications*, Math. Program. **48** (1990), 161–220.

[9] A. N. Iusem, *An iterative algorithm for variational inequalities problem*, Computat. Appl. Math. **13** (1994), 103–114.

[10] A. N. Iusem and Svaiter B.F., *A variant of Korpolevich's method for variational inequalities with a new search strategy*, Optimization **42** (1997), 309–321.

[11] G. M. Korpolevich, *The extragradient method for finding saddle points and other problems*, Matecon **12** (1976), 747–756.

[12] M.A. Noor, Y. Wang, and N. Xiu, *Some new projection methods for variational inequalities*, Appl. Math. Comput. **137** (2003), 423–435.

[13] J. Rosen and S. Suzuki, *Construction of non-linear programming test problems*, Commun. ACM **8(2):113** (1965).

[14] S.S. Singh, *Projection based algorithms for variational inequalities*, Ph.D. thesis, University of North Carolina at Chapel Hill, 2010.

[15] M.V. Solodov and B.F. Svaiter, *New projection method for variational inequality problems*, SIAM J. Control Optim. **37** (1999), 765–776.

[16] C.Y. Wang and F. Zhao, *Directional derivatives of optimal value functions in mathematical programming*, J. Optim. Theory Appl. **82** (1994), 397–404.

[17] Y.J. Wang, N. Xiu, and C.Y. Wang, *Unified framework of extragradient-type method for pseudomonotone variational inequalities*, J. Optim. Theory Appl. **111** (2001), 641–656.

Authors' addresses:

Hassina Grar
Department of Mathematics, Faculty of Sciences
Ferhat Abbas Setif-1 University, Algeria
E-mail: hassina.grar@univ-setif.dz

Zahira Kebaili
Department of Mathematics, Faculty of Sciences
Ferhat Abbas Setif-1 University, Algeria
E-mail: zahira.kebaili@univ-setif.dz

Received August 29, 2023
Revised October 1, 2024
Accepted November 22, 2024