Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 17, No. 2, pp. 269–288, 2021

LINESEARCH-FREE ALGORITHMS FOR SOLVING PSEUDOMONOTONE VARIATIONAL INEQUALITIES

Trinh Ngoc Hai

Abstract: Based on the recent works by Censor et al. [Censor, Y., Gibali, A., Reich, S.: The subgradient extragradient method for solving variational inequalities in Hilbert space. J. Optim. Theory and Appl., 148, 318-335 (2011)] and Tseng [Tseng, P.: A modified forward-backward splitting method for maximal monotone mapping. SIAM J. Control. Optim. 38, 431-446 (2000)], we propose two new algorithms for solving pseudomonotone and non-Lipschitz continuous variational inequalities. Our algorithms improve the recent one announced by Censor and Tseng in that they do not require the Lipschitz continuity of the involving mapping. Also, compared to the Armijo line-search, which uses the same assumption, our algorithms are more advanced: they only require to compute one projection at each iteration and do not use the line-search procedure. Some numerical experiments and applications to the traffic network are presented to test the effectiveness of the proposed algorithms.

Key words: *variational inequality, pseudomonotonicity, extragradient algorithm*

Mathematics Subject Classification: *47J20, 49J40, 49M30*

---

1 Introduction

Let $C \subset \mathbb{R}^m$ be a nonempty, closed and convex set, and $A : \mathbb{R}^m \to \mathbb{R}^m$ be a mapping. The variational inequality [19] of A on C is

$$\text{find } x^* \in C \text{ such that } \langle A(x^*), y - x^* \rangle \geq 0 \quad \forall y \in C. \tag{VI(A,C)}$$

Denote by $Sol(A,C)$ the solution set of VI(A,C). This problem is a classical subject in nonlinear analysis, optimization and operations research. Many important problems, such as fixed points problem, saddle point problem, optimization problem, Nash equilibrium problem, can be formulated in the form of variational inequalities [12]. A lot of algorithms have been proposed for solving VI(A,C) [4, 5, 8, 9, 10, 11, 15, 16, 17, 28, 30, 31, 32, 1, 3, 18, 29, 26, 25, 14, 2, 20, 21, 22], among them, the extension of the projection gradient algorithm seems to be the simplest:

$$\begin{cases} x^0 \in C, \\ x^{k+1} = P_C\left(x^k - \lambda A(x^k)\right). \end{cases} \tag{1.1}$$

Under the assumption that A is γ-strongly pseudomonotone and L-Lipschitz continuous, $\lambda \in (0, 2\gamma/L^2)$, the sequences $\{x^k\}$ generated by (1.1) linearly converge to the unique solution of VI(A, C) [17]. To reduce the convergence conditions, Korpelevich [24] introduced the extragradient algorithm:

$$\begin{cases} x^0 \in C \\ y^k = P_C\left(x^k - \lambda A(x^k)\right) \\ x^{k+1} = P_C\left(x^k - \lambda A(y^k)\right). \end{cases} \tag{1.2}$$

This algorithm is convergent under the assumption that A is pseudomonotone and Lipschitz continuous. One drawback of the extragradient algorithm is that it requires to compute the projection onto C two times per iteration. When the feasible set C has complicated form and its projection is computationally expensive, this drawback may seriously affect to the performance of the algorithm. To over come this drawback, Censor et al. [10] proposed the subgradient extragradient algorithm, replacing the second projection onto C in (1.2) by the one onto a half space:

$$\begin{cases} x^0 \in C \\ y^k = P_C\left(x^k - \lambda A(x^k)\right) \\ T_k = \left\{z \in \mathbb{R}^m : \left\langle x^k - \lambda A(x^k) - y^k, z - y^k \right\rangle \leq 0\right\} \\ x^{k+1} = P_{T_k}\left(x^k - \lambda A(y^k)\right). \end{cases} \tag{1.3}$$

Since the projection onto T_k has an explicit form and can be computed easily, (1.3) can be considered as an one-projection algorithm. When A is monotone and Lipschitz continuous on $\mathbb{R}^m$, the sequence $\{x^k\}$ is convergent to a desired solution. Another one-projection algorithm with the same assumption is the Tseng's one [30]:

$$\begin{cases} x^0 \in C \\ y^k = P_C\left(x^k - \lambda A(x^k)\right) \\ x^{k+1} = y^k + \lambda\left[A(x^k) - A(y^k)\right]. \end{cases} \tag{1.4}$$

Compared to (1.2), both the algorithms (1.3) and (1.4) are more advanced from a computational point of view. However, these algorithms have a disadvantage: they require the mapping A being Lipschitz continuous on the whole space. This is a rather strict condition.

On the other hand, in [28] Solodov and Svaiter proposed the Armijo line search algorithm, which does not require the Lipschitz continuity of the mapping A. But at each step of this algorithm, we need to check an inequality many times (the line-search procedure). This make the algorithm very computationally expensive.

Motivated by the Censor et al. [10], Tseng [30] and Solodov et al. [28], in this paper we introduce two new algorithms for solving VI(A, C). Our algorithms preserve the advantages and overcome the disadvantage of the works in [10, 30, 28]. On the one hand, convergence conditions of our algorithms are as mild as the ones of the Armijo line-search algorithm, i.e., we only need the mapping A being pseudomonotone but is not necessarily Lipschitz continuous. On the other hand, the new algorithms are computationally cheap: they require to compute the projection onto C only one time per iteration and do not use the line-search procedure.

The rest of the article is organized as follows. Section 2 is devoted to some definitions and preliminary results. Our algorithms and their convergence analysis are described in Section 3. Some further analysis on the convergence conditions of the proposed algorithms are presented in Section 4. In the last section, we compare the new algorithms with the existing ones and present an application to the traffic network.

2 Preliminaries

In this section, we recall some basic properties and definitions that will be used in the subsequent sections. We refer the reader to [7, 19, 27] for more details.

Let $C \subset \mathbb{R}^m$ be a nonempty, closed and convex set, and $x \in \mathbb{R}^m$ be an arbitrary point. There exists a uniquely point in C, denoted by $P_C(x)$, satisfying $\|x - P_C(x)\| \leq \|x - y\|$ for all $y \in C$. The mapping $x \mapsto P_C(x)$ is called the projection onto C.

Proposition 2.1 ([7])**.** *For all $x, y \in \mathbb{R}^m$, we have*

(i) $\|P_C(x) - P_C(y)\| \leq \|x - y\|$,

(ii) $\langle y - P_C(x), x - P_C(x)\rangle \leq 0$.

Definition 2.2. A mapping $A : C \to \mathbb{R}^m$ is said to be

1. pseudomonotone on C, iff for all $x, y \in C$, we have

$$\langle A(y), x - y\rangle \geq 0 \Rightarrow \langle A(x), x - y\rangle \geq 0.$$

2. Lipschitz continuous on C with modulus $L > 0$, iff for all $x, y \in C$, we have

$$\|A(x) - A(y)\| \leq L\|x - y\|.$$

Lemma 2.3 ([6])**.** *Let $\{a_k\}$, $\{b_k\} \subset (0, \infty)$ be two sequences satisfying*

$$a_{k+1} \leq a_k + b_k \ \forall k \geq 0 \ \text{and} \ \sum_{k=0}^{\infty} b_k < \infty.$$

Then, the sequence $\{a_k\}$ is convergent.

Lemma 2.4. [13] *Suppose that the mapping $A : C \to \mathbb{R}^m$ is continuous. Then, for all bounded sequences $\{x^k\}$, $\{y^k\} \subset C$ satisfying $\|x^k - y^k\| \to 0$, it holds that $\|A(x^k) - A(y^k)\| \to 0$.*

Let $I \subset \mathbb{N}$, denote by $|I|$ the number of elements in I.

3 Main Results

Assumption 3.1. We investigate problem VI(A, C) under the following conditions:

(A1) The mapping A is pseudomonotone on C;

(A2) There exist constants μ, $\delta > 0$ such that for all $x \in H$, $y \in C$, we have

$$\|x - y\| \leq \delta \Rightarrow \|A(x) - A(y)\| \leq \mu;$$

(A3) The mapping A is continuous on $\mathbb{R}^m$;

(A4) $Sol(A, C) \neq \emptyset$.

Assumption 3.1 is rather mild. Some practical models satisfying this assumption can be found in [1, 3, 29, 2, 16, 23].

Algorithm 3.1 (Subgradient extragradient algorithm for VI(A, C) without Lipschitz continuity)**.**
Step 0. Choose $x^0 \in C$, $\xi, \rho \in (0,1)$, $\alpha_0 \in \left(0, \frac{\delta}{1+\mu}\right)$. Set $k = 0$.
Step 1. Compute:

$$\begin{aligned}
&\lambda_k = \frac{\alpha_k}{\max\{1; \|A(x^k)\|\}}\\
&y^{k+1} = P_C\left(x^k - \lambda_k A(x^k)\right)\\
&x^{k+1} = P_{T_k}\left(x^k - \lambda_k A(y^{k+1})\right),\\
&\text{where } T_k = \left\{z \in \mathbb{R}^m : \left\langle x^k - \lambda_k A(x^k) - y^{k+1}, z - y^{k+1}\right\rangle \le 0\right\}.
\end{aligned}$$

If $\lambda_k \|A(x^k) - A(y^{k+1})\| \le \rho \|x^k - y^{k+1}\|$ **then** set $\alpha_{k+1} = \alpha_k$ **else** set $\alpha_{k+1} = \xi \alpha_k$.
Step 2. **If** $y^{k+1} = x^k$, **then** STOP, **else** update $k := k+1$ and GOTO **Step 1**.

Theorem 3.2. *Suppose that Assumption 3.1 is satisfied. Then, the sequence $\{x^k\}$ generated by Algorithm 3.1 converges to a solution of VI(A, C).*

Proof. Since

$$y^{k+1} = P_C\left(x^k - \lambda_k A(x^k)\right),$$

we have

$$\left\langle x^k - \lambda_k A(x^k) - y^{k+1}, z - y^{k+1}\right\rangle \le 0 \quad \forall z \in C. \tag{3.1}$$

Hence $C \subset T_k$. Since

$$x^{k+1} = P_{T_k}\left(x^k - \lambda_k A(y^{k+1})\right),$$

it implies that

$$\left\langle x^k - \lambda_k A(y^{k+1}) - x^{k+1}, z - x^{k+1}\right\rangle \le 0 \quad \forall z \in C \subset T_k$$

or

$$\left\langle x^k - x^{k+1}, z - x^{k+1}\right\rangle \le \lambda_k \left\langle A(y^{k+1}), z - x^{k+1}\right\rangle \quad \forall z \in C. \tag{3.2}$$

On the other hand, noting that $x^{k+1} \in T_k$, we get

$$\left\langle x^k - \lambda_k A(x^k) - y^{k+1}, x^{k+1} - y^{k+1}\right\rangle \le 0$$

or

$$\left\langle x^k - y^{k+1}, x^{k+1} - y^{k+1}\right\rangle \le \lambda_k \left\langle A(x^k), x^{k+1} - y^{k+1}\right\rangle. \tag{3.3}$$

From (3.2) and (3.3), it follows that for any $z \in C$,

$$\begin{aligned}
\|x^{k+1} - z\|^2 &= \|x^k - z\|^2 - \|x^{k+1} - y^{k+1}\|^2 - \|y^{k+1} - x^k\|^2 +\\
&\quad + 2\left\langle x^{k+1} - x^k, x^{k+1} - z\right\rangle + 2\left\langle y^{k+1} - x^k, y^{k+1} - x^{k+1}\right\rangle\\
&\le \|x^k - z\|^2 - \|x^{k+1} - y^{k+1}\|^2 - \|y^{k+1} - x^k\|^2 +\\
&\quad + 2\lambda_k \left\langle A(y^{k+1}), z - x^{k+1}\right\rangle + 2\lambda_k \left\langle A(x^k), x^{k+1} - y^{k+1}\right\rangle\\
&= \|x^k - z\|^2 - \|x^{k+1} - y^{k+1}\|^2 - \|y^{k+1} - x^k\|^2 +\\
&\quad + 2\lambda_k \left\langle A(y^{k+1}), z - y^{k+1}\right\rangle + 2\lambda_k \left\langle A(x^k) - A(y^{k+1}), x^{k+1} - y^{k+1}\right\rangle.
\end{aligned} \tag{3.4}$$

Let $z = x^* \in Sol(A, C)$, we have

$$\begin{aligned}\|x^{k+1} - x^*\|^2 \le \|x^k - x^*\|^2 - \|x^{k+1} - y^{k+1}\|^2 - \|y^{k+1} - x^k\|^2 \\ + 2\lambda_k \left\langle A(x^k) - A(y^{k+1}), x^{k+1} - y^{k+1}\right\rangle\end{aligned} \tag{3.5}$$

Denote

$$I := \left\{k \in \mathbb{N} : \lambda_k \|A(x^k) - A(y^{k+1})\| > \rho\|x^k - y^{k+1}\|\right\}.$$

Consider two cases:

Case 1: $|I| = \infty$. Let $I = \{k_n\}_{n=1}^{\infty} \subset \mathbb{N}$. We have

$$\lambda_{k_n}\|A(x^{k_n}) - A(y^{k_n+1})\| > \rho\|x^{k_n} - y^{k_n+1}\| \quad \forall n \ge 0.$$

For $x \in \mathbb{R}^m$, denote $\mathrm{dist}(x, C) := \|x - P_C(x)\|$.

Claim 1: $\lim_{k\to\infty}\mathrm{dist}(x^k, C) = 0$.

Since $|I| = \infty$, we have $\lim_{k\to\infty} \alpha_k = 0$. From the definition of T_k, it is easy seen that $y^{k+1} = P_{T_k}(x^k - \lambda_k A(x^k))$. For all $k \ge 0$, we have

$$\begin{aligned}\mathrm{dist}(x^{k+1}, C) &\le \|x^{k+1} - y^{k+1}\| \\ &= \|P_{T_k}\left(x^k - \lambda_k A(y^{k+1})\right) - P_{T_k}\left(x^k - \lambda_k A(x^k)\right)\| \\ &\le \lambda_k\|A(x^k) - A(y^{k+1})\| \\ &\le \alpha_k\|A(x^k) - A(y^{k+1})\|.\end{aligned} \tag{3.6}$$

On the other hand, it holds that

$$\begin{aligned}\|x^k - y^{k+1}\| &\le \|x^k - P_C(x^k)\| + \|P_C(x^k) - P_C(x^k - \lambda_k A(x^k))\| \\ &\le \mathrm{dist}(x^k, C) + \lambda_k\|A(x^k)\| \\ &\le \mathrm{dist}(x^k, C) + \alpha_k.\end{aligned} \tag{3.7}$$

We will prove that $\|x^k - y^{k+1}\| \le \delta$ for all $k \ge 0$. Indeed, from (3.7), we have $\|x^0 - y^1\| \le \alpha_0 < \delta$. Suppose that $\|x^j - y^{j+1}\| \le \delta$ for some $j \ge 0$. Using assumption (A2), we get $\|A(x^j) - A(y^{j+1})\| \le \mu$, and hence (3.6) implies that

$$\mathrm{dist}(x^{j+1}, C) \le \alpha_j\|A(x^j) - A(y^{j+1})\| \le \alpha_0\mu < \frac{\mu\delta}{1+\mu},$$

where the last inequality follows from the definition of α_0, i.e., $\alpha_0 < \frac{\delta}{1+\mu}$. From (3.7), we have

$$\begin{aligned}\|x^{j+1} - y^{j+2}\| &\le \mathrm{dist}(x^{j+1}, C) + \alpha_{j+1} \\ &< \frac{\mu\delta}{1+\mu} + \frac{\delta}{\mu+1} = \delta.\end{aligned}$$

By induction, it implies that $\|x^k - y^{k+1}\| \le \delta$ for all $k \ge 0$. Using assumption (A2) and (3.6), we get $\mathrm{dist}(x^k, C) \le \alpha_{k-1}\mu \to 0$ and hence

$$\|x^k - y^{k+1}\| \le \alpha_k + \mathrm{dist}(x^k, C) \to 0. \tag{3.8}$$

Claim 2: *The sequence* $\{x^k\}$ *is bounded.*

Take $k \ge 0$ arbitrarily. If $k \notin I$, then from (3.19), we get

$$\|x^{k+1} - x^*\| \le \|x^k - x^*\|. \tag{3.9}$$

If $k \in I$, from (3.6) and (3.7), we have

$$\begin{aligned}
\|x^{k+1} - x^*\| &\leq \|x^k - x^*\| + \|x^{k+1} - x^k\| \\
&\leq \|x^k - x^*\| + \|x^{k+1} - y^{k+1}\| + \|x^k - y^{k+1}\| \\
&\leq \|x^k - x^*\| + \alpha_k \|A(x^k) - A(y^{k+1})\| + \operatorname{dist}(x^k, C) + \alpha_k \\
&\leq \|x^k - x^*\| + \alpha_k \mu + \alpha_{k-1} \|A(x^{k-1}) - A(y^k)\| + \alpha_k \\
&\leq \|x^k - x^*\| + \alpha_k \mu + \frac{\alpha_k}{\xi} \mu + \alpha_k. \qquad (3.10)
\end{aligned}$$

Combinning (3.9) and (3.10), we have

$$\|x^{k+1} - x^*\| \leq \|x^k - x^*\| + \tau_k,$$

where

$$\tau_k := \begin{cases} 0 & \text{if } k \notin I, \\ \alpha_0 \left(\mu + \frac{\mu}{\xi} + 1\right) \xi^{n-1} & \text{if } k = k_n \in I. \end{cases}$$

Hence, $\sum_{k=0}^{\infty} \tau_k = \sum_{k \in I} \tau_k = \alpha_0 \left(\mu + \frac{\mu}{\xi} + 1\right) \sum_{n=0}^{\infty} \xi^n < \infty$. Using Lemma 2.3, we deduce that the sequence $\{\|x^k - x^*\|\}$ is convergent, and hence, the sequence $\{x^k\}$ is bounded. Since $\{\|x^k - y^{k+1}\|\} \to 0$, it implies that the sequence $\{y^k\}$ is also bounded.

Claim 3: The sequence $\{x^k\}$ converges to a solution of VI(A, C).

We have

$$\lambda_{k_n} \|A(x^{k_n}) - A(y^{k_n+1})\| > \rho \|x^{k_n} - y^{k_n+1}\| \quad \forall n \geq 0. \qquad (3.11)$$

The sequence $\{x^{k_n}\}$ is bounded, hence there exists a subsequence $\{x^{k_{n_t}}\} \subset \{x^{k_n}\}$ such that $x^{k_{n_t}} \to \bar{x}$. Since $\lim_{k\to\infty} \operatorname{dist}(x^k, C) = 0$, it implies that $\bar{x} \in C$. From (3.1), we have

$$\left\langle x^k - y^{k+1}, z - y^{k+1} \right\rangle \leq \lambda_k \left\langle A(x^k), z - y^{k+1} \right\rangle \quad \forall z \in C. \qquad (3.12)$$

From (3.11) and (3.12), we get

$$\begin{aligned}
\left\langle A(x^k), z - y^{k+1} \right\rangle &\geq -\frac{1}{\lambda_k} \|x^k - y^{k+1}\| \|z - y^{k+1}\| \\
&\geq -\frac{1}{\rho} \|A(x^k) - A(y^{k+1})\| \|z - y^{k+1}\| \quad \forall z \in C, \ k \in I.
\end{aligned}$$

In the last inequality, letting $k = k_{n_t}$, $t \to \infty$, using the continuity of A, the boundedness of $\{y^k\}$, the fact that $\|x^k - y^{k+1}\| \to 0$ and Lemma 2.4, we obtain

$$\langle A(\bar{x}), z - \bar{x} \rangle \geq 0 \quad \forall z \in C.$$

It implies that $\bar{x} \in Sol(A, C)$, and hence, the sequence $\|x^k - \bar{x}\|$ is convergent. Thus,

$$\lim_{k\to\infty} \|x^k - \bar{x}\| = \lim_{t\to\infty} \|x^{k_{n_t}} - \bar{x}\| = 0.$$

Case 2: $|I| < \infty$. Let $i_0 = \max\{i + 1 : i \in I\}$. It implies that $\alpha_k = \alpha_{i_0}$ for all $k \geq i_0$. From (3.19), we have

$$\|x^{k+1} - x^*\|^2 \leq \|x^k - x^*\|^2 - (1 - \rho) \|x^{k+1} - y^{k+1}\|^2 - (1 - \rho) \|y^{k+1} - x^k\|^2 \quad \forall k \geq i_0. \quad (3.13)$$

It follows that the sequence $\left\{\|x^k - x^*\|\right\}$ is convergent and

$$\lim_{k\to\infty} \|x^k - y^k\| = \lim_{k\to\infty} \|y^{k+1} - x^k\| = 0. \tag{3.14}$$

The sequence $\{x^k\}$ is bounded, and hence, there exists a subsequence $\{x^{k_m}\} \subset \{x^k\}$ such that $\lim_{m\to\infty} x^{k_m} = \bar{x}$. Since $\{y^k\} \subset C$ and $\|x^k - y^k\| \to 0$, it implies that $\bar{x} \in C$. From (3.2), for all $z \in C$ we have

$$\left\langle x^k - x^{k+1}, z - x^{k+1}\right\rangle \leq \frac{\alpha_k}{\max\{1; \|A(x^k)\|\}} \left(\left\langle A(y^{k+1}), z - y^{k+1}\right\rangle + \left\langle A(y^{k+1}), y^{k+1} - x^{k+1}\right\rangle\right).$$

Letting $k = k_m \to \infty$, using (3.14) and the continuity of A, we get

$$\frac{\alpha_{i_0}}{\max\{1; \|A(\bar{x})\|\}} \langle A(\bar{x}), z - \bar{x}\rangle \geq 0 \quad \forall z \in C,$$

or $\bar{x} \in Sol(A, C)$. Since the sequence $\{\|x^k - \bar{x}\|\}$ is convergent, we obtain

$$\lim_{k\to\infty} \|x^k - \bar{x}\| = \lim_{m\to\infty} \|x^{k_m} - \bar{x}\| = 0.$$

□

Algorithm 3.3 (Tseng-type algorithm for VI(A, C) without Lipschitz continuity)**.**
Step 0. Choose $x^0 \in C$, $\xi, \rho \in (0, 1)$, $\alpha_0 \in \left(0, \frac{\delta}{1+\mu}\right)$. Set $k = 0$.
Step 1. Compute:

$$\begin{aligned}
\lambda_k &= \frac{\alpha_k}{\max\{1; \|A(x^k)\|\}} \\
y^k &= P_C\left(x^k - \lambda_k A(x^k)\right) \\
x^{k+1} &= y^k + \lambda_k\left[A(x^k) - A(y^k)\right],
\end{aligned}$$

If $\lambda_k\|A(x^k) - A(y^k)\| \leq \rho\|x^k - y^k\|$ **then** set $\alpha_{k+1} = \alpha_k$ **else** set $\alpha_{k+1} = \xi\alpha_k$.
Step 2. **If** $y^{k+1} = x^k$, **then** STOP, **else** update $k := k + 1$ and GOTO **Step 1**.

Theorem 3.4. *Suppose that Assumption 3.1 is satisfied. Then, the sequence $\{x^k\}$ generated by Algorithm 3.1 converges to a solution of VI(A, C).*

Proof. Since $y^k = P_C(x^k - \lambda A(x^k))$, using the property of the projection, we have

$$\left\langle x - y^k, y^k - x^k + \lambda_k A(x^k)\right\rangle \geq 0 \quad \forall x \in C. \tag{3.15}$$

Let $x^* \in Sol(A, C)$, it holds that

$$\left\langle x^* - y^k, y^k - x^k + \lambda_k A(x^k)\right\rangle \geq 0. \tag{3.16}$$

On the other hand, since F is pseudomonotone, then

$$\lambda_k \left\langle y^k - x^*, A(y^k)\right\rangle \geq 0 \tag{3.17}$$

Adding (3.16) and (3.17), we get

$$\left\langle x^* - y^k, y^k - x^k + \lambda_k\left[A(x^k) - A(y^k)\right]\right\rangle \geq 0.$$

From the definition of x^{k+1}, it implies that

$$\left\langle x^* - y^k, x^{k+1} - x^k \right\rangle \geq 0.$$

Thus,

$$\begin{aligned}\left\langle x^{k+1} - x^*, x^{k+1} - x^k \right\rangle &\leq \left\langle x^{k+1} - y^k, x^{k+1} - x^k \right\rangle \\ &= \|x^{k+1} - x^k\|^2 + \left\langle x^k - y^k, x^{k+1} - x^k \right\rangle \\ &= \|x^{k+1} - x^k\|^2 + \left\langle x^k - y^k, y^k + \lambda_k [A(x^k) - A(y^k)] - x^k \right\rangle \\ &= \|x^{k+1} - x^k\|^2 - \|y^k - x^k\|^2 + \lambda_k \left\langle x^k - y^k, A(x^k) - A(y^k) \right\rangle.\end{aligned} \tag{3.18}$$

Applying the equality $a^2 + b^2 - (a-b)^2 = 2\langle a, b \rangle$, we get

$$\|x^{k+1} - x^*\|^2 - \|x^k - x^*\|^2 + \|x^{k+1} - x^k\|^2 = 2\left\langle x^{k+1} - x^*, x^{k+1} - x^k \right\rangle. \tag{3.19}$$

From (3.18) and (3.19), we have

$$\begin{aligned}\|x^{k+1} - x^*\|^2 &= \|x^k - x^*\|^2 - \|x^{k+1} - x^k\|^2 + 2\left\langle x^{k+1} - x^*, x^{k+1} - x^k \right\rangle \\ &\leq \|x^k - x^*\|^2 + \|x^{k+1} - x^k\|^2 - 2\|y^k - x^k\|^2 \\ &\quad + 2\lambda_k \left\langle x^k - y^k, A(x^k) - A(y^k) \right\rangle \\ &= \|x^k - x^*\|^2 + \|y^k + \lambda_k (A(x^k) - A(y^k)) - x^k\|^2 - 2\|y^k - x^k\|^2 \\ &\quad + 2\lambda_k \left\langle x^k - y^k, A(x^k) - A(y^k) \right\rangle \\ &= \|x^k - x^*\|^2 - \|y^k - x^k\|^2 + \lambda_k^2 \|A(x^k) - A(y^k)\|^2.\end{aligned} \tag{3.20}$$

Denote

$$I := \{k \in \mathbb{N} : \lambda_k \|A(x^k) - A(y^k)\| > \rho \|x^k - y^k\|\}.$$

Consider two cases:
Case 1: $|I| < \infty$. There exists a number $k_0 \in \mathbb{N}$ such that

$$\lambda_k \|A(x^k) - A(y^k)\| \leq \rho \|x^k - y^k\| \text{ and } \alpha_k = \alpha_{k_0} \quad \forall k \geq k_0.$$

From (3.20), it follows that

$$\|x^{k+1} - x^*\|^2 \leq \|x^k - x^*\|^2 - (1 - \rho^2)\|x^k - y^k\|^2 \quad \forall k \geq k_0.$$

So, the sequence $\{\|x^k - x^*\|\}$ is convergent for all $x^* \in Sol(F, C)$. It implies that the sequence $\{x^k\}$ is bounded and

$$\lim_{k \to \infty} \|x^k - y^k\| = 0.$$

There exists a subsequence $\{x^{k_i}\} \subset \{x^k\}$ such that $x^{k_i} \to \bar{x} \in \mathbb{R}^m$. Since $\{y^k\} \subset C$ and $\|x^k - y^k\| \to 0$, we obtain $\bar{x} \in C$. Next, we will prove that $\bar{x}$ is a solution of VI(A, C). From (3.15), we have

$$\left\langle A(x^k), x - y^k \right\rangle \geq -\frac{1}{\lambda_k} \|x - y^k\| \|y^k - x^k\| \quad \forall x \in C. \tag{3.21}$$

Since $\lambda_k = \frac{\alpha_k}{\max\{1; \|A(x^k)\|\}}$, $\alpha_k = \alpha_{k_0}$ for all $k \geq k_0$ and the sequence $\{A(x^k)\}$ is bounded, we deduce that the sequence $\{\lambda_k\}$ is bounded away from zero. In (3.21), letting $k = k_i$, $i \to \infty$, using the continuity of A, the boundedness of $\{x^k\}$ and the fact $\|y^k - x^k\| \to 0$, we obtain

$$\langle A(\bar{x}), x - \bar{x} \rangle \geq 0 \quad \forall x \in C,$$

or $\bar{x} \in Sol(F, C)$. The sequence $\{\|x^k - \bar{x}\|\}$ is convergent, hence

$$\lim_{k\to\infty} \|x^k - \bar{x}\| = \lim_{i\to\infty} \|x^{k_i} - \bar{x}\| = 0.$$

Case 2: $|I| = \infty$. Let $I = \{k_j\}_{j=1}^{n} \subset \mathbb{N}$, then

$$\lambda_{k_j}\|A(x^{k_j}) - A(y^{k_j})\| > \rho\|x^{k_j} - y^{k_j}\| \quad \forall j \geq 0.$$

It is easy seen that the sequence $\{\alpha_k\}$ is decreasing and tending to zero. Denote $\operatorname{dist}(x, C) := \|x - P_C(x)\|$ for all $x \in \mathbb{R}^m$. We will prove that $\operatorname{dist}(x^k, C) \to 0$ and $\|x^k - y^k\| \to 0$. For all $k \geq 0$, we have

$$\begin{aligned}\operatorname{dist}(x^{k+1}, C) &\leq \|x^{k+1} - y^k\| \\ &= \lambda_k\|A(x^k) - A(y^k)\| \\ &\leq \alpha_k\|A(x^k) - A(y^k)\| \end{aligned} \tag{3.22}$$

and

$$\begin{aligned}\|x^k - y^k\| &\leq \|x^k - P_C(x^k)\| + \|P_C(x^k) - y^k\| \\ &= \operatorname{dist}(x^k, C) + \|P_C(x^k) - P_C(x^k - \lambda_k A(x^k))\| \\ &\leq \operatorname{dist}(x^k, C) + \lambda_k\|A(x^k)\| \\ &\leq \operatorname{dist}(x^k, C) + \alpha_k. \end{aligned} \tag{3.23}$$

Analogously to the proof of Theorem 3.2, from (3.22) and (3.23), by induction, we can prove that $\|x^k - y^k\| \leq \delta$ for all $k \geq 0$. Using condition (A2) and (3.22), we get $\operatorname{dist}(x^k, C) \leq \alpha_{k-1}\mu \to 0$ and hence

$$\|x^k - y^k\| \leq \alpha_k + \operatorname{dist}(x^k, C) \to 0. \tag{3.24}$$

The rest of the proof is similar to the one of Theorem 3.2 and hence, is omitted. □

Remark 3.5. (a) Algorithm 3.1 and Algorithm 3.3 are improved from the Censor's algorithm [10] and Tseng's algorithm [30] for solving non-Lipschitz variational inequalities but it can solve Lipschitz variational inequalities as efficiently as the original ones do. Indeed, when the mapping A is Lipschitz continuous, there exists a number $k_0 > 0$ such that $\alpha_k = \alpha_{k_0}$ for all $k \geq k_0$, i.e., the steps-sizes of the new algorithms are bounded away from zero as in the original ones. Moreover, in this case, compared the algorithms in [10, 30], the new ones have a clear advantage: it does not require to know the Lipschitz constant of the mapping A.

(b) When the mapping A is pseudomonotone but is not Lipschitz continuous, the Armijo line search algorithm (ALA) [28] is often referred to as a typical method for solving $\mathrm{VI}(A, C)$. In comparison with (ALA), our algorithms have the following advantages. First, to compute the value of the step size λ_k, we only have to check an inequality one time instead of many time as in (ALA). This means, our algorithms do not use the line-search procedure, which is very computationally expensive. Second, at each step of our algorithms, we only have to perform one projection onto the feasible set instead of two projection as in (ALA). This feature also helps to reduce the computational cost of our algorithms.

(c) Condition (A2) and (A3) can be replaced by the uniform continuity of A. Moreover, from the proofs of Theorems 3.2 and 3.4, it is easy seen that Condition (A2) can be omitted if C is bounded or $\{x^k\} \subset C$.

$\boxed{4}$ Further Analysis

In this section, we study the possibility of eliminating the conditions in the proposed algorithms.

$\boxed{4.1}$ The non-emptiness of the solution set

To prove the convergence of Algorithm 3.1 and Algorithm 3.3, we need to assume that the solution set is not empty. In practice, this condition is rather difficult to verify. Thanks to the following corollary, we can implement our algorithms without checking this condition.

Corollary 4.1. *Suppose that Conditions (A1), (A2) and (A3) are satisfied, $\{x^k\}$ is the sequence generated by Algorithm 3.1 or Algorithm 3.3. Then, the solution set $Sol(A, C)$ is not empty if and only if the sequence $\{x^k\}$ is convergent.*

Proof. We will prove for the case of Algorithm 3.1. The rest case is proved similarly. Obviously, it is sufficient to show that

$$\text{the sequence } \{x^k\} \text{ is convergent} \Rightarrow Sol(A, C) \neq \emptyset.$$

Define

$$I := \left\{ i \in \mathbb{N} : \lambda_i \|A(x^i) - A(y^{i+1})\| > \rho \|x^i - y^{i+1}\| \right\}.$$

Let $\lim_{k\to\infty} x^k = \bar{x}$. Consider two cases:
Case 1: $|I| < \infty$. There exists $i_0 > 0$ such that

$$\begin{aligned} \|x^{k+1} - x\|^2 \le \|x^k - x\|^2 - (1-\rho)\|x^{k+1} - y^{k+1}\|^2 - (1-\rho)\|y^{k+1} - x^k\|^2 + \\ + 2\frac{\alpha_{i_0}}{\max\{1; \|A(x^k)\|\}} \left\langle A(y^{k+1}), x - y^{k+1} \right\rangle \quad \forall k \ge i_0, x \in C. \end{aligned} \tag{4.1}$$

Similarly to the proof of Theorem 3.2, it holds that $\|x^k - y^{k+1}\| \le \delta$ for all $k \ge 0$. Combinning this and the fact $\lim_{k\to\infty} x^k = \bar{x}$, we have that the sequence $\{y^k\}$ is bounded. There exists a subsequence $\{y^{k_n}\} \subset \{y^k\}$ such that $y^{k_n} \to \bar{y} \in C$. In (4.1), let $k = k_n - 1 \to \infty$, notting that $\{x^k\}$ is convergent and using the continuity of A, we get

$$\frac{\alpha_{i_0}}{\max\{1; \|A(\bar{x})\|\}} \langle A(\bar{y}), x - \bar{y} \rangle \ge 0 \quad \forall x \in C.$$

Hence, $\bar{y} \in Sol(A, C)$. In (4.1), let $x = \bar{y}$, $k = k_n - 1 \to \infty$, we obtain

$$2(1-\rho)\|\bar{x} - \bar{y}\|^2 \le \frac{\alpha_{i_0}}{\max\{1; \|A(\bar{x})\|\}} \langle A(\bar{y}), \bar{y} - \bar{y} \rangle = 0.$$

Hence, $\lim_{k\to\infty} x^k = \bar{x} \in Sol(A, C)$.
Case 2: $|I| = \infty$. There exists a subsequence $\{y^{k_n}\} \subset \{y^k\}$ such that

$$\lambda_{k_n} \|A(x^{k_n}) - A(y^{k_n+1})\| > \rho \|x^{k_n} - y^{k_n+1}\| \quad \forall n \ge 0. \tag{4.2}$$

On the other hand, from the definition of y^{k+1}, we have

$$\left\langle x^k - y^{k+1}, x - y^{k+1} \right\rangle \le \lambda_k \left\langle A(x^k), x - y^{k+1} \right\rangle \quad \forall x \in C, k \ge 0. \tag{4.3}$$

Combinning (4.2) and (4.3), we get

$$-\frac{\rho+1}{\rho} \|A(x^{k_n}) - A(y^{k_n+1})\| \|x - y^{k_n+1}\| \le \left\langle A(y^{k_n+1}), x - y^{k_n+1} \right\rangle. \tag{4.4}$$

Analogously to (3.24), we get

$$\|x^k - y^{k+1}\| \le \operatorname{dist}(x^k, C) + \alpha_k \to 0.$$

Hence, $\lim_{k\to\infty} x^k = \lim_{k\to\infty} y^k = \bar{x}$. From Lemma 2.4, it follows that $\|A(x^k) - A(y^{k+1})\| \to 0$. In (4.4), let $n \to \infty$, using the continuity of A, we obtain

$$\langle A(\bar{x}), x - \bar{x}\rangle \ge 0 \quad \forall x \in C.$$

This means $\lim_{k\to\infty} x^k = \bar{x} \in Sol(A, C)$. □

Remark 4.2. Thanks to Corollary 4.1, we can implement Algorithm 3.1 and Algorithm 3.3 without checking the solvability of VI(A, C) (Condition (A4)). If the generated sequence $\{x^k\}$ is convergent, we can claim that its limit is a desired solution. Otherwise, we claim that the solution set is empty.

4.2 The pseudomonotonicity of A

To prove the convergence of the proposed algorithms, the pseudomonotonicity of A cannot be omitted. Considered the following example: $C = \mathbb{R}^m$, $A : \mathbb{R}^m \to \mathbb{R}^m$, $A(x) = -x$ for all $x \in \mathbb{R}^m$. Then, A is Lipschitz continuous but is not pseudomonotone. It is easy seen that $Sol(A, C) = \{0\}$. Algorithm 3.1 now becomes

$$\begin{aligned} y^{k+1} &= P_C\left(x^k - \lambda_k A(x^k)\right) = (1 + \lambda_k)\, x^k, \\ x^{k+1} &= P_{T_k}\left(x^k - \lambda_k A(y^{k+1})\right) = x^k + \lambda_k y^{k+1} = x^k\left(\lambda_k^2 + \lambda_k + 1\right). \end{aligned}$$

Since $\lambda_k > 0$, we have $\|x^{k+1}\| > \|x^k\|$ for all $k \ge 0$, and hence, the sequence $\{x^k\}$ does not converge to the unique solution $x^* = 0$ of VI(A, C).

We have shown that the pseudomonotonicity condition in Theorem 3.2 and Theorem 3.4 cannot be omitted. However, it may happen that A is not pseudomonotone but the sequence $\{x^k\}$ generated by the proposed algorithms is convergent. From the proof of Corollary 4.1, it follows that in this case, the limit of $\{x^k\}$ is a solution of VI(A, C).

Corollary 4.3. *Suppose that Conditions (A2) and (A3) in Assumption 3.1 are satisfied and the sequence $\{x^k\}$ generated by Algorithm 3.1 or Algorithm 3.3 converges to $\bar{x}$. Then, $\bar{x}$ is a solution of VI(A, C).*

4.3 Closedness and convexity of the feasible set

From the proof of Theorem 3.2 and Theorem 3.4, it is easy seen that we can change the projection onto C in Step 1 of the proposed algorithms by the one onto any close, convex subset $D \subset C$ containing a solution. One of ways to construct D is:

$$D = \bigcap_{z\in K} \{x \in C : \langle A(z), x - z\rangle \le 0\}, \tag{4.5}$$

where K is an arbitrary subset of C. Note that we can choose the set K such that the projection onto D is calculated easier than the one onto C. For example, let

$$C = \{x \in \mathbb{R}^2 : x_1, x_2 \ge 0, x_1x_2 < 2\},$$

$A : C \to C$ defined by $A(x) = x$ for all $x \in C$. It is easy seen that C is neither closed nor convex, and hence, the projection onto C cannot be computed. Choose $K = \{(1, 0); (0, 1)\} \subset C$. According to (4.5), the set D is

$$D = \{x \in \mathbb{R}^2 : 0 \le x_1, x_2 \le 1\}.$$

We see that the projection onto D can be calculated explicitly.

5 Numerical Experiments

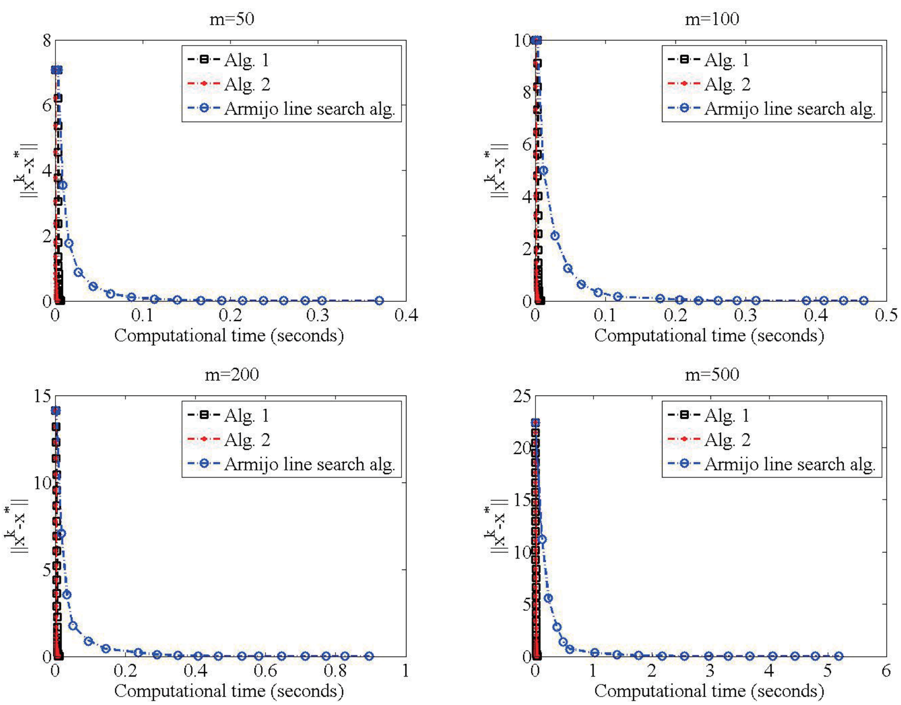

Figure 1: Performance of the algorithms in Problem 1, Example 5.1 with different m

To test the effectiveness of the new algorithms, we implement them in MATLAB to solve variational inequality problems. Also, we compare them with the Armijo-line search algorithm (ALA) [28], which has the same conditions. We do not compare with the extragradient-type ones [24, 32, 10] because these algorithms need the Lipschitz continuity, which is omitted in our algorithms. We use MATLAB version R2010b on a PC with Intel®Core2$^{\text{TM}}$ Quad Processor Q9400 2.66Ghz 4GB Ram.

Example 5.1. The algorithms are tested in the following problems:

- Problem 1: $C = [-5, 5]^m \subset \mathbb{R}^m$, $F : \mathbb{R}^m \to \mathbb{R}^m$, $F(x) = x$ for all $x \in \mathbb{R}^m$.
- Problem 2: $C = [-5, 5]^m \subset \mathbb{R}^m$, $F : \mathbb{R}^m \to \mathbb{R}^m$, $F(x) = Ax$ for all $x \in \mathbb{R}^m$, $A = (a_{ij})$,

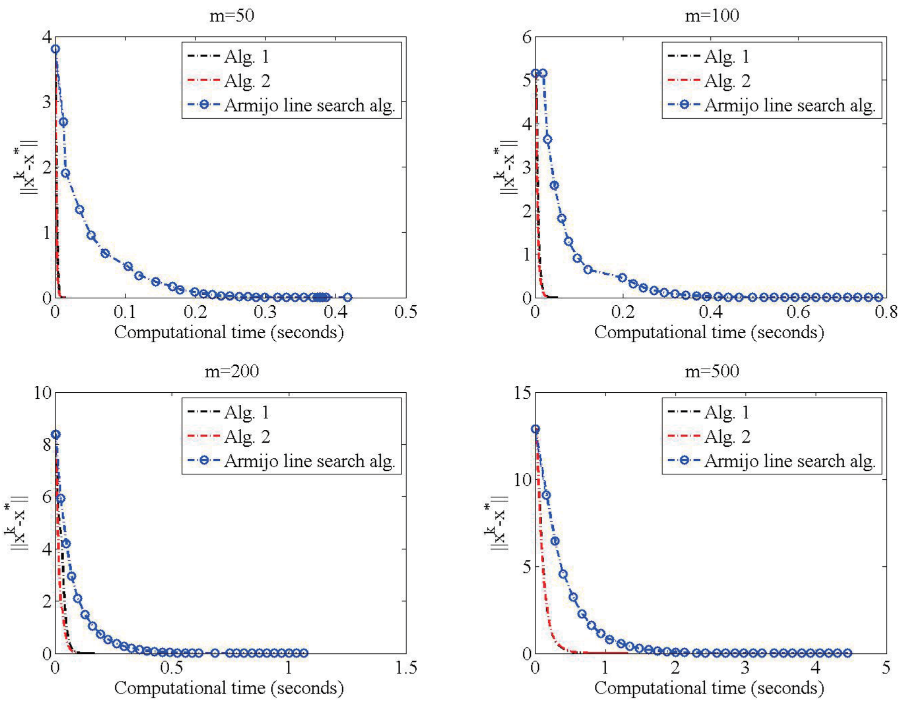

Figure 2: Performance of the algorithms in Problem 2, Example 5.1 with different m

where

$$a_{ij} = \begin{cases} -1 & \text{if } j = m+1-i,\ j > i \\ 1 & \text{if } j = m+1-i,\ j < i \\ 0 & \text{otherwise.} \end{cases}$$

- Problem 3: $C = \mathbb{R}^m$, $F : \mathbb{R}^m \to \mathbb{R}^m$, $F(x) = Bx + q$ for all $x \in \mathbb{R}^m$, where $q = (1, \dots, 1)^T$ and $B = (b_{ij})$,

$$b_{ij} = \begin{cases} 2 & \text{if } j = i, \\ 1 & \text{otherwise.} \end{cases}$$

- Problem 4. This is an extreme problem. Let $C = [-5, 5]^m \subset \mathbb{R}^m$, $F : \mathbb{R}^m \to \mathbb{R}^m$, $F(x) = Ax$ for all $x \in \mathbb{R}^m$, A is the diagonal matrix whose diagonal entries are $10^{-5}, 10^5, 1, , 1, \dots, 1$. The mapping F in this problem is strongly monotone and Lipschitz continuous, hence the Projection Algorithm (PA) [17] is the most suitable one for solving this problem. It should give better results than our algorithms and (ALA). But we found that in this problem, even (PA) does not work effectively. We will explain why. It is well known that the Projection Algorithm

$$\begin{cases} x^0 \in C, \\ x^{k+1} = P_C(x^k - \lambda F(x^k)) \end{cases} \tag{PA}$$

converges linearly under the assumption that $\lambda \in (0, 2\frac{\gamma}{L^2})$, where γ is the strong monotonicity constant and L is the Lipschitz constant of F [17]. We have $\gamma = 10^{-5}$

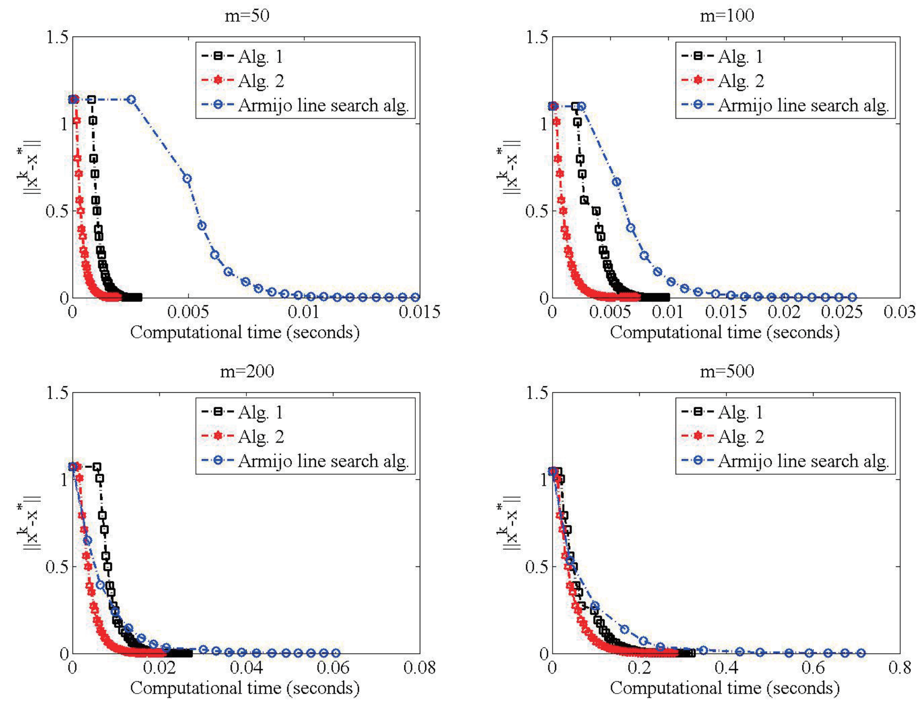

Figure 3: Performance of the algorithms in Problem 3, Example 5.1 with different m

and $L = 10^5$, hence $\lambda < 2.10^{-15}$. Since the step-size λ is too small, the algorithm converges very slowly. Our primary calculations show that: to obtain the accuracy $\|x^k - x^*\| \leq 10^{-3}$, (PA) needs to perform at least 3.10^{20} iterations. The similar problem occurs with our algorithms and (ALA). Since A has extreme eigenvalues, the stepsizes tend zero very fast, and hence, slow down the algorithms. We have tried to test our algorithms and (ALA) with this problem. The result is that these algorithms have not obtained the accuracy $\|x^k - x^*\| \leq 10^{-3}$ after 6 hours of CPU time.

Since we can not compare the algorithms in such an extreme problem, we test them in a less extreme one. That is Problem 4 with A is the diagonal matrix whose diagonal entries are $10^{-2}, 10^2, 1, , 1, \ldots, 1$.

The unique solution of Problem 1, 2 and 4 is $x^* = (0, \ldots, 0)^T$. Meanwhile, in Problem 3, it is $x^* = B^{-1}.q$. It is easy seen that in these problems, the convergence conditions are satisfied. We implement the algorithms with the same starting point x^0, which is randomly generated and the same stopping rule $\|x^k - x^*\| \leq 10^{-4}$, where x^* is the unique solution of the problem. The parameters are chosen as follows:

- For (ALA): since $\gamma, \sigma \in (0, 1)$, we have tested the algorithm with $\gamma, \sigma = 0.1, 0.2, \ldots, 0.9$ and found that among these options, (ALA) performs the best with $\gamma = 0.5$, $\sigma = 0.3$. These values of γ, σ also have been used in [28]. Hence, in our tests, we will use $\gamma = 0.5$, $\sigma = 0.3$.
- In Algorithm 3.1 and 3.3, $\rho = 0.7$, $\xi = 0.7$ and $\alpha_0 = \|x^0\|$.

We test the algorithms with different m. The results are presented in Figures 1, 2, 3 and Table 1. We can see that the computational time of the new algorithms is smaller than that of (ALA). This happened because in our algorithms, we do not have to perform the line-search procedure, which is very computationally expensive. Moreover, in (ALA), one has to compute two projections onto the feasible set and its intersection with a half-space. Meanwhile, in the new algorithms, we only have to compute one projection onto C.

	Algorithm 3.1		Algorithm 3.3		(ALA)	
	`Times(s)`	`Iter.`	`Times(s)`	`Iter.`	`Times(s)`	`Iter.`
m=50	22.3890	111809	20.2758	111818	73.0360	12832
m=100	25.8910	112931	23.7800	112945	-	-
m=500	28.6792	147044	27.7840	147080	-	-

Table 1: Comparison of Algorithm 3.1 and 3.3 with the Armijo line search algorithm in Problem 4, Example 5.1. Dash - indicates that the algorithm did not stop after 1000 seconds of CPU time.

Example 5.2. *(Application to traffic networks)*
Consider a transportation network with nodes $\{a, b, \ldots\}$, connected by oriented links $q \in \mathcal{L}$ [20]. Denote by W the set of pairs of origins and destinations. For each element $w = (a \to b) \in W$, P_w is the set of all paths with the origin a and the destination b. Let Θ be the set of all paths in the transportation network:

$$\Theta = \cup_{w \in W} P_w.$$

The paths in Θ are numbered:

$$\Theta = \{p_1, p_2, \ldots, p_n\}.$$

Denote by x_i the traffic flow on the path p_i and let $x = (x_1, x_2, \ldots, x_n)$ - the vector flow of the whole transportation network. For $w = (a \to b) \in W$, let b_w be the amount of vehicles that need to go from a to b, and define

$$C := \left\{ x \in \mathbb{R}^n_+ : \sum_{p \in P_w} x_p = b_w \ \forall w \in W \right\}.$$

The flow on a link q is equal to the sum of the flows on all paths containing this link, that is:

$$u_q = \sum_{p \in \Theta} \delta_{pq} x_p,$$

where

$$\delta_{pq} := \begin{cases} 1 & \text{if the link } q \text{ in path } p, \\ 0 & \text{otherwise.} \end{cases}$$

The user cost t_q on the link q depends on its flow u_q by the formula

$$t_q := \begin{cases} \tau_q u_q + \sigma_q & \text{if } 0 \le u_q \le \nu_q; \\ \rho_q u_q + \tau_q \nu_q + \sigma_q - \rho_q \nu_q & \text{if } u_q > \nu_q. \end{cases} \tag{5.1}$$

In (5.1), ν_q is the maximum capacity of the road q. When the traffic flow u_q exceeds this capacity, the user cost increases with the huge rate ρ_q (traffic jam). The cost of the path p is equal to the total cost of all its link:

$$G_p(x) := \sum_{q\in\mathcal{L}} \delta_{pq} t_q.$$

A flow vector x^* with the components x_q^* $(q \in P_w,\ w \in W)$ is an equilibrium pattern if, once established, no user want to change his/her path. In other words, when the traffic network attains an equilibrium state, among all path of P_w, the path with traffic has the lowest cost. This means, for all $w \in W$ and for all $r \in P_w$, it holds that

$$x_r^* > 0 \Rightarrow G_r(x^*) = \min_{p\in P_w} G_p(x^*). \tag{5.2}$$

The problem of finding x^* satisfying (5.2) is equivalent to the variational inequality VI(G, C) [20], where $G(x) = (G_1(x), G_2(x), \ldots, G_n(x))$. The cost functions defined by (5.1) are increasing, hence the mapping G is monotone [20]. Moreover, it is continuous on C. All the conditions of Assumption 3.1 are satisfied, we will apply Algorithm 3.1 to find the equilibrium state of the transportation network. Assume that the traffic network has 5 nodes, joined by links $q_1, q_2, \ldots, q_8$ as shown in Figure 4. Let $W = \{(1 \to 5)\}$ and $b_{(1\to5)} = 1000$. From node 1 to node 5, there are 5 paths: $p_1 = q_1 + q_6$, $p_2 = q_3 + q_8$, $p_3 = q_2 + q_7$, $p_4 = q_2 + q_5 + q_8$, $p_5 = q_2 + q_4 + q_6$. The constants of the cost functions are provided in Table 2.

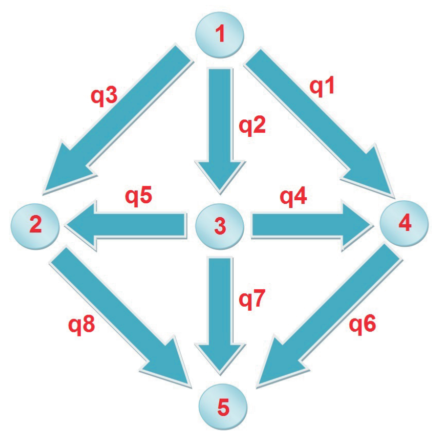

Figure 4: Traffic network with 5 nodes

In Algorithm 3.1, we choose $x^0 = (200, 200, 200, 200, 200)$, $\rho = \xi = 0.7$, $\alpha_0 = \|x^0\|$. Since the exact solution of the problem is unknown, we use the stopping criterion: $\|y^k - P_C\left[y^k - G(y^k)\right]\| < 10^{-4}$. Note that x^* is a solution of the problem if and only if $x^* - P_C\left[x^* - G(x^*)\right] = 0$. After 138 iterations, the algorithm find the approximate solution:

$$x^* = (338.9726,\ 342.2060,\ 283.7184,\ 28.1883,\ 6.9147)^{\mathrm{T}}.$$

In Table 3 and Figure 5, we present the comparison results of the proposed algorithms with the Armijo line search algorithm (ALA). The parameters of these algorithms are chosen as in Example 5.1. As we can see, in this example, the new algorithms have a better behavior in terms of the computational time.

q	τ_q	σ_q	ν_q	ρ_q
q_1	1	100	100	10
q_2	1.1	120	120	11
q_3	0.9	80	80	9
q_4	0.1	150	150	8
q_5	0.1	70	70	11
q_6	0.7	140	210	12
q_7	1.2	150	150	13
q_8	0.6	160	250	14

Table 2: Constants of the cost functions.

	Alg. 3.1		Alg. 3.3		(ALA)	
	`Times(s)`	`Iter.`	`Times(s)`	`Iter.`	`Times(s)`	`Iter.`
$x^0 = (200, 200, 200, 200, 200)$	1.2203	138	1.8120	219	6.0115	140
$x^0 = (1000, 0, 0, 0, 0)$	1.2567	175	1.6086	238	4.3766	117
$x^0 = (0, 0, 1000, 0, 0)$	0.8163	112	1.6248	236	4.2734	127
$x^0 = (100, 150, 200, 250, 300)$	0.9663	135	1.5722	235	4.5473	133

Table 3: Performance of the algorithms in Example 5.2 with different starting points.

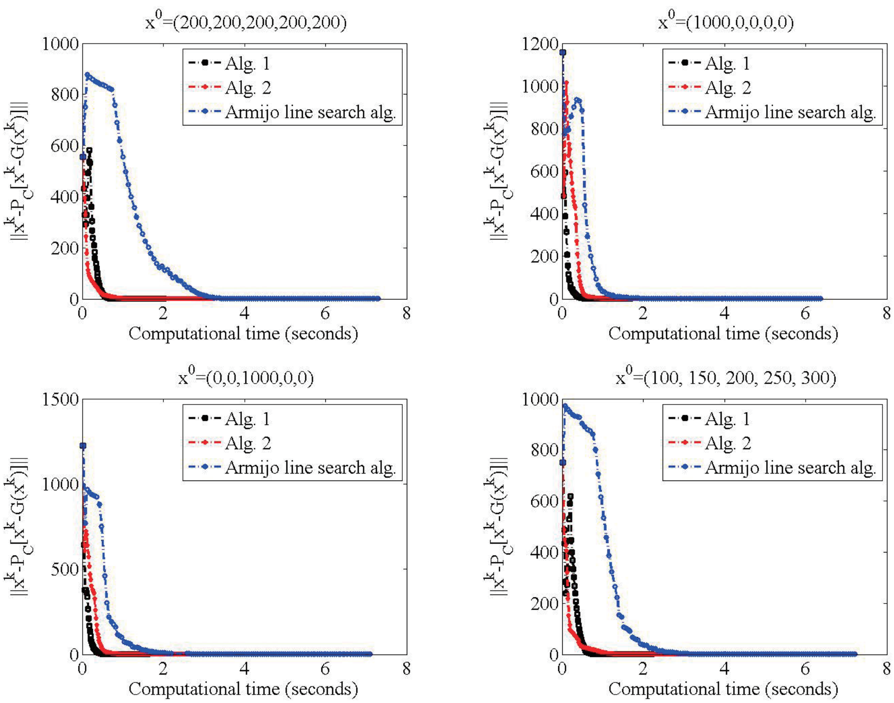

Figure 5: Comparision of the algorithms in Example 5.2

$\boxed{6}$ Conclusion

In this article, two novel projection algorithms have been proposed for solving pseudomonotone variational inequalities. In contrast to the existing algorithms mentioned in the introduction, the new ones have the followings advantages:

1. The involving mapping A need not to be Lipschitz continuous. We prove the convergence of our algorithms under the assumption that A is pseudomonotone and some suitable conditions.
2. At each iteration, we only have to compute one projection onto the feasible set. Also, the new algorithms do not use the Armijo line search as the existing ones. These features help to speed-up our algorithm.
3. To implement the new algorithms, we do not have to verify the nonemptiness of the solution set. We prove that the solvability of the problem is equivalent to the convergent of the generated sequence.

The effectiveness of the new algorithms have been confirmed by numerical experiments. Also, an application to the traffic network equilibrium problem has been presented.

Acknowledgements

The author would like to thank the anonymous referees for their remarks which helped to significantly improve the paper.

References

[1] P.K. Anh and T.N. Hai, Splitting extragradient-like algorithms for strongly pseudomonotone equilibrium problems, *Numer. Algor.* 76 (2017) 67–91.

[2] P.K. Anh and T.N. Hai, Novel self-adaptive algorithms for non-Lipschitz equilibrium problems with applications, *J. Global Optim.* 73 (2019) 637–657.

[3] P.K. Anh, T.N. Hai and P.M. Tuan, On ergodic algorithms for equilibrium problems, *J. Global Optim.* 64 (2016) 179–195.

[4] P.K. Anh and N.T. Vinh, Self-adaptive gradient projection algorithms for variational inequalities involving non-Lipschitz continuous operators, Numer. Algorithms 81 (2019) 983–1001.

[5] T.Q. Bao and P.Q. Khanh, A projection-type algorithm for pseudomonotone nonlipschitzian multi-valued variational inequalities, *Nonconvex Optim. Appl.* 77 (2005) 113–129.

[6] H.H. Bauschke and J. M. Borwein, On projection algorithms for solving convex feasibility problems, *SIAM Review* 38 (1996) 367–426.

[7] H.H. Bauschke and P.L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, New York, 2011.

[8] J.Y. Bello Cruz and A.N. Iusem, An explicit algorithm for monotone variational inequalities, Optimization 61 (2012) 855–871.

[9] J.Y. Bello Cruz and A.N. usem, Convergence of direct methods for paramonotone variational inequalities, Comput. Optim. Appl. 46 (2010) 247–263.

[10] Y. Censor, A. Gibali andS. Reich, The subgradient extragradient method for solving variational inequalities in Hilbert space, *J. Optim. Theory Appl.* 148 (2011) 318–335.

[11] J.Y. Bello Cruz and R.D. Millán, A direct splitting method for nonsmooth variational inequalities, *J. Optim. Theory Appl.* 161 (2014) 729–737.

[12] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer, New York, 2003.

[13] T.N. Hai, Two modified extragradient algorithms for solving variational inequalities, *J. Global Optim.* DOI 10.1007/s10898-020-00895-y

[14] T.N. Hai and N.T. Vinh, Two new splitting algorithms for equilibrium problems, *Rev. R. Acad. Cienc. Exactas Fís. Nat. Ser. A Math. RACSAM* 111 (2017) 1051–1069.

[15] H. Iiduka, Fixed point optimization algorithm and its application to power control in CDMA data networks, *Math. Program.* 133 (2012) 227–242.

[16] H. Iiduka and I. Yamada, An ergodic algorithm for the power-control games for CDMA data networks, *J. Math. Model. Algorithms* 8 (2009) 1–18.

[17] P.D. Khanh and P.T. Vuong, Modified projection method for strongly pseudomonotone variational inequalities, *J. Global Optim.* 58 (2014) 341–350.

[18] J.K. Kim, P.N. Anh and T.N. Hai, The Bruck's ergodic iteration method for the Ky Fan inequality over the fixed point set, *Int. J. Comput. Math.* 94 (2017) 2466–2480.

[19] D. Kinderlehrer and G. Stampacchia, *An Introduction to Variational Inequalities and their Applications*, Academic Press, New York, 1980.

[20] I.V. Konnov, *Combined Relaxation Methods for Variational Inequalities*, Springer, Berlin, 2000.

[21] I.V. Konnov, A combined relaxation method for variational inequalities with nonlinear constraints, *Math. Program.* 80 (1998) 239–252.

[22] I.V. Konnov, A class of combined iterative methods for solving variational inequalities, *J. Optim. Theory Appl.* 94 (1997) 677–693.

[23] I.V. Konnov, *Equilibrium Models and Variational Inequalities*, Elsevier, Amsterdam, 2007.

[24] G.M. Korpelevich, The extragradient method for finding saddle points and other problems, *Ekon. Mat. Metody* 12 (1976) 747–756.

[25] J. Mashreghi and M. Nasri, Forcing strong convergence of Korpelevich ' s method in Banach spaces with its applications in game theory, *Nonlinear Anal.* 72 (2010) 2086–2099.

[26] M.A. Noor, Extragradient methods for pseudomonotone variational inequalities, *J. Optim. Theory Appl.* 117 (2003) 475–488.

[27] R.T. Rockafellar, *Convex Analysis* Princeton University Press, Princeton, New Jersey, 1970.

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

[29] L.Q. Thuy and T.N. Hai, A projected subgradient algorithm for bilevel equilibrium problems and applications, *J. Optim. Theory Appl.* 175 (2017) 411–431.

[30] P. Tseng, A modified forward-backward splitting method for maximal monotone mapping, *SIAM J. Control. Optim.* 38 (2000) 431–446.

[31] P.T. Vuong, J.J. Strodiot and V.H. Nguyen, Extragradient methods and linesearch algorithms for solving Ky Fan inequalities and fixed point problems, *J. Optim. Theory Appl.* 155 (2013) 605–627.

[32] J. Yang and H. Liu, Strong convergence result for solving monotone variational inequalities in Hilbert space, *Numer. Algor.* 80 (2019) 741–752.

Manuscript received 9 August 2020
revised 2 March 2021
accepted for publication 2 June 2021

Trinh Ngoc Hai
School of Applied Mathematics and Informatics
Hanoi University of Science and Technology
1 Dai Co Viet, Hai Ba Trung, Hanoi, Vietnam
E-mail hai.trinhngoc@hust.edu.vn