



---

*Research article*

**A modified inertial viscosity extragradient type method for equilibrium problems application to classification of diabetes mellitus: Machine learning methods**

**Suthep Suantai<sup>1</sup>, Watcharaporn Yajai<sup>2</sup>, Pronpat Peeyada<sup>2</sup>, Watcharaporn Cholanjiak<sup>2,\*</sup> and Petcharaporn Chachvarat<sup>3,\*</sup>**

<sup>1</sup> Department of Mathematics, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand

<sup>2</sup> School of Science, University of Phayao, Phayao 56000, Thailand

<sup>3</sup> Faculty of Medicine at University of Phayao, Phayao 56000, Thailand

\* **Correspondence:** Email: watcharaporn.ch@up.ac.th, faifai2529@hotmail.com.

**Abstract:** Diabetes is one of the four major types of noncommunicable diseases (cardiovascular disease, diabetes, cancer and chronic respiratory diseases). It is chronic condition that occurs when the body does not produce enough insulin therefore results in raised blood sugar levels. Insulin is a hormone that regulates the blood sugar when food consumption. If the proper treatment is not received organs of the body like kidneys, nervous system and eyes can deteriorate. Therefore, it is better to predict diabetes as early as possible because lead to serious damage to many of the body's systems. In this paper, we modify extragradient method with an inertial extrapolation step and viscosity-type method to solve equilibrium problems of pseudomonotone bifunction operator in real Hilbert spaces. Strong convergence result is obtained under the assumption that the bifunction satisfies the Lipchitz-type condition. Moreover, we show choosing stepsize parameter in many ways, this shows that our algorithm is flexible using. Finally, we apply our algorithm to solve the diabetes mellitus classification in machine learning and show the algorithm's efficiency by comparing with existing algorithms.

**Keywords:** equilibrium problems; extragradient algorithm; Hilbert spaces; viscosity method; classification

**Mathematics Subject Classification:** 65K15, 47H05, 49M37

---

## 1. Introduction

Let  $\mathcal{H}$  be a real Hilbert space and  $\mathcal{K}$  is a nonempty closed and convex subset of a real Hilbert space  $\mathcal{H}$ . The equilibrium problem (EP) is to find an element  $u^* \in \mathcal{K}$  such that

$$f(u^*, v) \geq 0, \forall v \in \mathcal{K} \quad (1.1)$$

where  $f : \mathcal{K} \times \mathcal{K} \rightarrow \mathbb{R}$  is a bifunction and satisfying  $f(z, z) = 0$  for all  $z \in \mathcal{H}$ , and  $EP(f, \mathcal{K})$  is denoted for a solution set of EP (1.1). EP (1.1) generalizes many various problems in optimization problems such as variational inequalities problems, Nash equilibrium problems, linear programming problems, among others.

To solve the problem EP (1.1), the two-step extragradient method (TSEM) was proposed by Tran et al. [25] in 2008 which is inspired by the concept ideas to solve variational inequalities of Korpelevich [10]. Under the Lipschitz condition of the bifunction  $f$  the convergence theorem was proved. However, only weak convergence was obtained in Hilbert spaces. For obtaining strong convergence, this goal was completed by Hieu [8] with Halpern subgradient extragradient method (HSEM) which was modified from the HSEM of Kraikaew and Saejung [11] for variational inequalities. This method is defined by  $u \in \mathcal{H}$  and

$$\begin{cases} y_i &= \arg \min_{y \in \mathcal{K}} \left\{ \lambda f(x_i, y) + \frac{1}{2} \|x_i - y\|^2 \right\}, \\ T_i &= \{v \in \mathcal{H} : \langle (x_i - \lambda t_i) - y_i, v - y_i \rangle \leq 0\}, \quad t_i \in \partial_2 f(x_i, y_i), \\ z_i &= \arg \min_{y \in T_i} \left\{ \lambda f(y_i, y) + \frac{1}{2} \|x_i - y\|^2 \right\}, \\ x_{i+1} &= \alpha_i u + (1 - \alpha_i) z_i, \end{cases} \quad (1.2)$$

where  $\lambda$  is still some constant depending on the interval that makes the bifunction  $f$  satisfies the Lipschitz condition and  $\{\alpha_i\} \subset (0, 1)$  which satisfies the principle conditions

$$\lim_{i \rightarrow \infty} \alpha_i = 0, \quad \sum_{i=1}^{\infty} \alpha_i = \infty. \quad (1.3)$$

Very recently, Muangchoo [14] combined a viscosity-type method with the extragradient algorithm for obtaining strong convergence theorem of the EP (1.1). This method is defined by

$$\begin{cases} v_i &= \arg \min_{v \in \mathcal{K}} \left\{ \lambda_i f(z_i, v) + \frac{1}{2} \|z_i - v\|^2 \right\}, \\ \mathcal{E}_i &= \{t \in \mathcal{H} : \langle z_i - \lambda_i \omega_i - v_i, t - v_i \rangle \leq 0\}, \quad \omega_i \in \partial_2 f(z_i, v_i), \\ t_i &= \arg \min_{v \in \mathcal{E}_i} \left\{ \mu \lambda_i f(v_i, v) + \frac{1}{2} \|z_i - v\|^2 \right\}, \\ z_{i+1} &= \alpha_i g(z_i) + (1 - \alpha_i) t_i, \end{cases} \quad (1.4)$$

where  $\mu \in (0, \sigma) \subset \left(0, \min \left\{1, \frac{1}{2c_1}, \frac{1}{2c_2}\right\}\right)$ ,  $g$  is a contraction function on  $\mathcal{H}$  with contraction constant  $\xi \in [0, 1)$  ( $\|g(x) - g(y)\| \leq \xi \|x - y\|$ ,  $\forall x, y \in \mathcal{H}$ ),  $\{\alpha_i\}$  satisfies the principle conditions  $\lim_{i \rightarrow \infty} \alpha_i = 0$  and  $\sum_{i=1}^{+\infty} \alpha_i = +\infty$ , and the step-sizes  $\{\lambda_i\}$  is developed by updating the step-sizes method without knowing the Lipschitz-type constants of the bifunction  $f$  which satisfies the following:

$$\lambda_{i+1} = \begin{cases} \min \left\{ \sigma, \frac{\mu f(v_i, t_i)}{K_i} \right\}, & \text{if } \frac{\mu f(v_i, t_i)}{K_i} > 0, \\ \lambda_0, & \text{otherwise,} \end{cases} \quad (1.5)$$

where  $K_i = f(z_i, t_i) - f(z_i, v_i) - c_1 \|z_i - v_i\|^2 - c_2 \|t_i - v_i\|^2 + 1$ .

Finding technique to speed up the convergence of the algorithm is the way that many mathematicians are interested. One of that is an inertial which was first introduced by Polyak [16]. Very recently, Shehu et al. [22] modified the inertial technique with the Halpern-type algorithm and subgradient extragradient method for obtaining strong convergence to a solution of  $EP(f, \mathcal{K})$  such that  $f$  is pseudomonotone. This method is defined by  $\gamma \in \mathcal{H}$  and

$$\begin{cases} w_i &= \alpha_i \gamma + (1 - \alpha_i)z_i + \delta_i(z_i - z_{i-1}), \\ v_i &= \arg \min_{v \in \mathcal{K}} \left\{ \lambda_i f(w_i, v) + \frac{1}{2} \|w_i - v\|^2 \right\}, \\ \mathcal{E}_i &= \{t \in \mathcal{H} : \langle w_i - \lambda_i \omega_i - v_i, t - v_i \rangle\}, \quad \omega_i \in \partial_2 f(w_i, v_i), \\ t_i &= \arg \min_{v \in \mathcal{E}_i} \left\{ \lambda f(v_i, v) + \frac{1}{2} \|w_i - v\|^2 \right\}, \\ z_{i+1} &= \tau w_i + (1 - \tau)t_i, \end{cases} \quad (1.6)$$

where the inertial parameter  $\delta_i \in [0, \frac{1}{3}]$ ,  $\tau \in (0, \frac{1}{2}]$ , the update step-size  $\{\lambda_i\}$  satisfies the following:

$$\lambda_{i+1} = \begin{cases} \min \left\{ \frac{\mu}{2} \frac{\|w_i - v_i\|^2 + \|t_i - v_i\|^2}{f(w_i, t_i) - f(w_i, v_i) - f(v_i, t_i), \lambda_i} \right\}, & \text{if } f(w_i, t_i) - f(w_i, v_i) - f(v_i, t_i) > 0, \\ \lambda_i, & \text{otherwise,} \end{cases} \quad (1.7)$$

and  $\{\alpha_i\}$  still satisfies the principle conditions  $\lim_{i \rightarrow \infty} \alpha_i = 0$  and  $\sum_{i=1}^{\infty} \alpha_i = \infty$ . This update step-size  $\{\lambda_i\}$  is limited in the computation and can not be modified in another way.

In this paper, motivated and inspired by the above works in literature, we introduce a modified inertial viscosity extragradient type method for solving equilibrium problems. Moreover, we apply our main result to solve a data classification problem in machine learning and show the performance of our algorithm by comparing with many existing methods.

## 2. Preliminaries

Let us begin with some concepts of monotonicity of a bifunction [2, 15]. Let  $\mathcal{K}$  be a nonempty, closed and convex subset of  $\mathcal{H}$ . A bifunction  $f : \mathcal{H} \times \mathcal{H} \rightarrow \mathbb{R}$  is said to be:

(i) strongly monotone on  $\mathcal{K}$  if there exists a constant  $\gamma > 0$  such that

$$f(x, y) + f(y, x) \leq -\gamma \|x - y\|^2, \quad \forall x, y \in \mathcal{K};$$

(ii) mototone on  $\mathcal{K}$  if  $f(x, y) \leq -f(y, x), \forall x, y \in \mathcal{K}$ ;

(iii) pseudomonotone on  $\mathcal{K}$  if  $f(x, y) \geq 0 \implies f(y, x) \leq 0, \forall x, y \in \mathcal{K}$ ;

(iv) satisfying Lipschitz-type condition on  $\mathcal{K}$  if there exist two positive constants  $c_1, c_2$  such that

$$c_1 \|x - y\|^2 + c_2 \|y - z\|^2 \geq f(x, z) - f(x, y) - f(y, z), \quad \forall x, y, z \in \mathcal{K}.$$

The normal cone  $N_{\mathcal{K}}$  to  $\mathcal{K}$  at a point  $x \in \mathcal{K}$  is defined by  $N_{\mathcal{K}}(x) = \{w \in \mathcal{H} : \langle w, x - y \rangle \geq 0, \forall y \in \mathcal{K}\}$ . For every  $x \in \mathcal{H}$ , the metric projection  $P_{\mathcal{K}}x$  of  $x$  onto  $\mathcal{K}$  is the nearest point of  $x$  in  $\mathcal{K}$ , that is,  $P_{\mathcal{K}}x = \arg \min\{\|y - x\| : y \in \mathcal{K}\}$ . Since  $\mathcal{K}$  is nonempty, closed and convex,  $P_{\mathcal{K}}x$  exists and is unique. For each  $x, z \in \mathcal{H}$ , by  $\partial_2 f(z, x)$ , we denote the subdifferential of convex function  $f(z, \cdot)$  at  $x$ , i.e.,

$$\partial_2 f(z, x) = \{u \in \mathcal{H} : f(z, y) \geq \langle u, y - x \rangle + f(z, x), \quad \forall y \in \mathcal{H}\}.$$

In particular, for  $z \in \mathcal{K}$ ,

$$\partial_2 f(z, z) = \{u \in \mathcal{H} : f(z, y) \geq \langle u, y - z \rangle, \forall y \in \mathcal{H}\}.$$

For proving the convergence of the proposed algorithm, we need the following lemmas.

**Lemma 2.1.** [1] For each  $x \in \mathcal{H}$  and  $\lambda > 0$ ,

$$\frac{1}{\lambda} \langle x - \text{prox}_{\lambda g}(x), y - \text{prox}_{\lambda g}(x) \rangle \leq g(y) - g(\text{prox}_{\lambda g}(x)), \forall y \in \mathcal{K},$$

where  $\text{prox}_{\lambda g}(x) = \arg \min \{\lambda g(y) + \frac{1}{2} \|x - y\|^2 : y \in \mathcal{K}\}$ .

For any point  $u \in \mathcal{H}$ , there exists point  $P_{\mathcal{K}}u \in \mathcal{K}$  such that

$$\|u - P_{\mathcal{K}}u\| \leq \|u - y\|, \forall y \in \mathcal{K}.$$

$P_{\mathcal{K}}$  is called the metric projection of  $\mathcal{H}$  onto  $\mathcal{K}$ . We know that  $P_{\mathcal{K}}$  is a nonexpansive mapping of  $\mathcal{H}$  onto  $\mathcal{K}$ , that is,  $\|P_{\mathcal{K}}x - P_{\mathcal{K}}y\| \leq \|x - y\|$ ,  $\forall x, y \in \mathcal{K}$ .

**Lemma 2.2.** [18] Let  $g : \mathcal{K} \rightarrow \mathbb{R}$  be a convex, subdifferentiable and lower semicontinuous function on  $\mathcal{K}$ . Suppose the convex set  $\mathcal{K}$  has nonempty interior, or  $g$  is continuous at a point  $x^* \in \mathcal{K}$ . Then,  $x^*$  is a solution to the following convex problem  $\min\{g(x) : x \in \mathcal{K}\}$  if and only if  $0 \in \partial g(x^*) + N_{\mathcal{K}}(x^*)$ , where  $\partial g(\cdot)$  denotes the subdifferential of  $g$  and  $N_{\mathcal{K}}(x^*)$  is the normal cone of  $\mathcal{K}$  at  $x^*$ .

**Lemma 2.3.** [19, 22] Let  $S \subseteq \mathbb{R}$  be a nonempty, closed, and convex subset of a real Hilbert space  $\mathcal{H}$ . Let  $u \in \mathcal{H}$  be arbitrarily given,  $z = P_S u$ , and  $\Omega = \{x \in \mathcal{H} : \langle x - u, x - z \rangle \leq 0\}$ . Then  $\Omega \cap S = \{z\}$ .

**Lemma 2.4.** [28] Let  $\{a_i\}$  and  $\{c_i\}$  be nonnegative sequences of real numbers such that  $\sum_{i=1}^{\infty} c_i < \infty$ , and let  $\{b_i\}$  be a sequence of real numbers such that  $\limsup_{i \rightarrow \infty} b_i \leq 0$ . If for any  $i \in \mathbb{N}$  such that

$$a_{i+1} \leq (1 - \gamma_i)a_i + \gamma_i b_i + c_i,$$

where  $\{\gamma_i\}$  is a sequence in  $(0, 1)$  such that  $\sum_{i=1}^{\infty} \gamma_i = \infty$ , then  $\lim_{i \rightarrow \infty} a_i = 0$ .

**Lemma 2.5.** [29] Let  $\{a_i\}$ ,  $\{b_i\}$  and  $\{c_i\}$  be positive sequences, such that

$$a_{i+1} \leq (1 + c_i)a_i + b_i, \quad i \geq 1.$$

If  $\sum_{i=1}^{\infty} c_i < +\infty$  and  $\sum_{i=1}^{\infty} b_i < +\infty$ ; then,  $\lim_{i \rightarrow +\infty} a_i$  exists.

The convergence of Algorithm 3.1 will be given under the conditions that

**Condition 2.6.** (A1)  $f$  is pseudomonotone on  $\mathcal{K}$  with  $\text{int}(\mathcal{K}) \neq \emptyset$  or  $f(x, \cdot)$  is continuous at some  $z \in \mathcal{K}$  for every  $x \in \mathcal{K}$ ;

(A2)  $f$  satisfies Lipschitz-type condition on  $\mathcal{H}$  with two constants  $c_1$  and  $c_2$ ;

(A3)  $f(\cdot, y)$  is sequentially weakly upper semicontinuous on  $\mathcal{K}$  for each fixed point  $y \in \mathcal{K}$ , i.e., if  $\{x_i\} \subset \mathcal{K}$  is a sequence converging weakly to  $x \in \mathcal{K}$ , then  $f(x, y) \geq \limsup_{i \rightarrow \infty} f(x_i, y)$ ;

(A4) for  $x \in \mathcal{H}$ ,  $f(x, \cdot)$  is convex and lower semicontinuous, subdifferentiable on  $\mathcal{H}$ ;

(A5)  $V : \mathcal{H} \rightarrow \mathcal{H}$  is contraction with contraction constant  $\alpha$ .

### 3. Main results

Now, we are in a position to present a modification of algorithm (EGM) in [25] for equilibrium problems.

**Algorithm 3.1.** (Modified viscosity type inertial extragradient algorithm for EPs)

**Initialization.** Let  $x_0, x_1 \in H$ ,  $0 < \lambda_i \leq \lambda < \frac{1}{2 \max\{c_1, c_2\}}$ ,  $\tau \in (0, \frac{1}{2}]$ .

**Step 1.** Given the current iterates  $x_{i-1}$  and  $x_i (i \geq 1)$  and  $\alpha_i \in (0, 1)$ ,  $\theta_i \in [0, \frac{1}{3})$ , compute

$$w_i = \alpha_i V(x_i) + (1 - \alpha_i)x_i + \theta_i(x_i - x_{i-1}),$$

$$y_i = \arg \min_{y \in K} \{ \lambda_i f(w_i, y) + \frac{1}{2} \|y - w_i\|^2 \}.$$

If  $y_i = w_i$  then stop and  $y_i$  is a solution. Otherwise, go to **Step 2**.

**Step 2.** Compute

$$z_i = \arg \min_{y \in K} \{ \lambda_i f(y_i, y) + \frac{1}{2} \|y - w_i\|^2 \}.$$

**Step 3.** Compute

$$x_{i+1} = (1 - \tau)w_i + \tau z_i.$$

Set  $i = i + 1$  and return to **Step 1**.

In this section, we will analyse the convergence of Algorithm 3.1.

For the rest of this paper, we assume the following condition.

**Condition 3.2.** (i)  $\{\alpha_i\} \subset (0, 1]$  is non-increasing with  $\lim_{i \rightarrow \infty} \alpha_i = 0$  and  $\sum_{i=1}^{\infty} \alpha_i = \infty$ ;

(ii)  $0 \leq \theta_i \leq \theta_{i+1} \leq \theta < \frac{1}{3}$  and  $\lim_{i \rightarrow \infty} \frac{\theta_i}{\alpha_i} \|x_i - x_{i-1}\| = 0$ ;

(iii)  $EP(f, K) \neq \emptyset$ .

Before we prove the strong convergence result, we need some lemmas below.

**Lemma 3.3.** Assume that Conditions 2.6 and 3.2 hold. Let  $\{x_i\}$  be generated by Algorithm 3.1. Then there exists  $N > 0$  such that

$$\|x_{i+1} - u\|^2 \leq \|w_i - u\|^2 - \|x_{i+1} - w_i\|^2, \quad \forall u \in EP(f, K), \quad i \geq N. \quad (3.1)$$

*Proof.* By the definition of  $y_i$ , and Lemma 2.1, we have

$$\frac{1}{\lambda_i} \langle w_i - y_i, y - y_i \rangle \leq f(w_i, y) - f(w_i, y_i), \quad \forall y \in K. \quad (3.2)$$

Putting  $y = z_i$  into (3.2), we obtain

$$\frac{1}{\lambda_i} \langle y_i - w_i, y_i - z_i \rangle \leq f(w_i, z_i) - f(w_i, y_i). \quad (3.3)$$

By the definition of  $z_i$ , we have

$$\frac{1}{\lambda_i} \langle w_i - z_i, y - z_i \rangle \leq f(y_i, y) - f(y_i, z_i), \quad \forall y \in \mathcal{K}. \quad (3.4)$$

(3.3) and (3.4) imply that

$$\frac{2}{\lambda_i} \langle w_i - z_i, y - z_i \rangle + \frac{2}{\lambda_i} \langle y_i - w_i, y_i - z_i \rangle \leq 2f(y_i, y) + 2(f(w_i, z_i) - f(w_i, y_i) - f(y_i, z_i)). \quad (3.5)$$

If  $f(w_i, z_i) - f(w_i, y_i) - f(y_i, z_i) > 0$ , then

$$f(w_i, z_i) - f(w_i, y_i) - f(y_i, z_i) \leq c_1 \|w_i - y_i\|^2 + c_2 \|z_i - y_i\|^2 \quad (3.6)$$

Observe that (3.6) is also satisfied if  $f(w_i, z_i) - f(w_i, y_i) - f(y_i, z_i) \leq 0$ . By (3.5) and (3.6), we have

$$\langle w_i - z_i, y - z_i \rangle + \langle y_i - w_i, y_i - z_i \rangle \leq \lambda_i f(y_i, y) + \lambda_i c_1 \|w_i - y_i\|^2 + \lambda_i c_2 \|z_i - y_i\|^2. \quad (3.7)$$

Note that

$$\langle w_i - z_i, z_i - y \rangle = \frac{1}{2} (\|w_i - y\|^2 - \|w_i - z_i\|^2 - \|z_i - y\|^2). \quad (3.8)$$

and

$$\langle w_i - y_i, z_i - y_i \rangle = \frac{1}{2} (\|w_i - y_i\|^2 + \|z_i - y_i\|^2 - \|w_i - z_i\|^2). \quad (3.9)$$

Using (3.8) and (3.9) in (3.7), we obtain, for all  $y \in \mathcal{K}$ ,

$$\|z_i - y\|^2 \leq \|w_i - y\|^2 - (1 - 2\lambda_i c_1) \|w_i - y_i\|^2 - (1 - 2\lambda_i c_2) \|z_i - y_i\|^2 + 2\lambda_i f(y_i, y). \quad (3.10)$$

Taking  $y = u \in EP(f, \mathcal{K}) \subset \mathcal{K}$ , one has  $f(u, y_i) \geq 0, \forall i$ . By (A1), we obtain  $f(y_i, u) \leq 0, \forall i$ . Hence, we obtain from (3.10) that

$$\|z_i - u\|^2 \leq \|w_i - u\|^2 - (1 - 2\lambda_i c_1) \|w_i - y_i\|^2 - (1 - 2\lambda_i c_2) \|z_i - y_i\|^2. \quad (3.11)$$

It follows from  $\lambda_i \in (0, \frac{1}{2\max\{c_1, c_2\}})$  and (3.11), we have

$$\|z_i - u\| \leq \|w_i - u\|. \quad (3.12)$$

On the other hand, we have

$$\|x_{i+1} - u\|^2 = (1 - \tau) \|w_i - u\|^2 + \tau \|z_i - u\|^2 - (1 - \tau) \tau \|z_i - w_i\|^2. \quad (3.13)$$

Substituting (3.11) into (3.13), we obtain

$$\begin{aligned} \|x_{i+1} - u\|^2 &\leq \|w_i - u\|^2 - \tau \|w_i - u\|^2 + \tau \|w_i - u\|^2 - \tau (1 - 2\lambda_i c_1) \|w_i - y_i\|^2 \\ &\quad - \tau (1 - 2\lambda_i c_2) \|z_i - y_i\|^2 - (1 - \tau) \tau \|z_i - w_i\|^2. \end{aligned} \quad (3.14)$$

Moreover, we have  $z_i - w_i = \frac{1}{\tau}(x_{i+1} - w_i)$ , which together with (3.14) gives

$$\begin{aligned} \|x_{i+1} - u\|^2 &\leq \|w_i - u\|^2 - \tau(1 - 2\lambda_i c_1)\|w_i - y_i\|^2 - \tau(1 - 2\lambda_i c_2)\|z_i - y_i\|^2 \\ &\quad - (1 - \tau)\tau \frac{1}{\tau^2} \|x_{i+1} - w_i\|^2 \\ &\leq \|w_i - u\|^2 - \frac{1 - \tau}{\tau} \|x_{i+1} - w_i\|^2 \\ &\leq \|w_i - u\|^2 - \epsilon \|x_{i+1} - w_i\|^2, \quad \forall i \geq N, \end{aligned} \quad (3.15)$$

where  $\epsilon = \frac{1-\tau}{\tau}$ .  $\square$

**Lemma 3.4.** Assume that Conditions 2.6 and 3.2 hold. Let  $\{x_i\}$  be generated by Algorithm 3.1. Then, for all  $u \in EP(f, \mathcal{K})$ ,

$$\begin{aligned} -2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle &\geq \|x_{i+1} - u\|^2 - \|x_i - u\|^2 + 2\theta_{i+1}\|x_{i+1} - x_i\|^2 - 2\theta_i\|x_i - x_{i-1}\|^2 \\ &\quad + \alpha_{i+1}\|V(x_i) - x_{i+1}\|^2 - \alpha_i\|x_i - V(x_i)\|^2 - \theta_i\|x_i - u\|^2 \\ &\quad + \theta_{i-1}\|x_{i-1} - u\|^2 + (1 - 3\theta_{i+1} - \alpha_i)\|x_i - x_{i+1}\|^2. \end{aligned} \quad (3.16)$$

*Proof.* By Lemma 2.5, we have

$$\|x_{i+1} - u\|^2 \leq \|w_i - u\|^2 - \|x_{i+1} - w_i\|^2. \quad (3.17)$$

Moreover, from the definition of  $w_i$ , we obtain that

$$\begin{aligned} \|w_i - u\|^2 &= \|x_i - u\|^2 + \|\theta_i(x_i - x_{i-1}) - \alpha_i(x_i - V(x_i))\|^2 \\ &\quad + 2\langle x_i - u, \theta_i(x_i - x_{i-1}) - \alpha_i(x_i - V(x_i)) \rangle \\ &= \|x_i - u\|^2 + \|\theta_i(x_i - x_{i-1}) - \alpha_i(x_i - V(x_i))\|^2 \\ &\quad + 2\theta_i \langle x_i - u, x_i - x_{i-1} \rangle - 2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle. \end{aligned} \quad (3.18)$$

Replacing  $u$  by  $x_{i+1}$  in (3.18), we have

$$\begin{aligned} \|w_i - x_{i+1}\|^2 &= \|x_i - x_{i+1}\|^2 + \|\alpha_i(x_i - V(x_i)) - \theta_i(x_i - x_{i-1})\|^2 \\ &\quad + 2\theta_i \langle x_i - x_{i+1}, x_i - x_{i-1} \rangle - 2\alpha_i \langle x_i - x_{i+1}, x_i - V(x_i) \rangle. \end{aligned} \quad (3.19)$$

Substituting (3.18) and (3.19) into (3.17), we have

$$\begin{aligned} \|x_{i+1} - u\|^2 &\leq \|x_i - u\|^2 + \|\theta_i(x_i - x_{i-1}) - \alpha_i(x_i - V(x_i))\|^2 + 2\theta_i \langle x_i - u, x_i - x_{i-1} \rangle \\ &\quad - 2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle - \|x_i - x_{i+1}\|^2 - 2\theta_i \langle x_i - x_{i+1}, x_i - x_{i-1} \rangle \\ &\quad + 2\alpha_i \langle x_i - x_{i+1}, x_i - V(x_i) \rangle - \|\alpha_i(x_i - V(x_i)) - \theta_i(x_i - x_{i-1})\|^2 \\ &= \|x_i - u\|^2 - \|x_i - x_{i+1}\|^2 + 2\theta_i \langle x_i - u, x_i - x_{i-1} \rangle - 2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle \\ &\quad + 2\alpha_i \langle x_i - x_{i+1}, x_i - V(x_i) \rangle + \theta_i \|x_i - x_{i+1}\|^2 + \theta_i \|x_i - x_{i-1}\|^2 \\ &\quad - \theta_i \|x_i - x_{i+1} + (x_i - x_{i-1})\|^2. \end{aligned} \quad (3.20)$$

Therefore, we obtain

$$\|x_{i+1} - u\|^2 - \|x_i - u\|^2 - \theta_i \|x_i - x_{i-1}\|^2 + \|x_i - x_{i+1}\|^2 - \theta_i \|x_i - x_{i+1}\|^2$$

$$\begin{aligned}
&\leq 2\theta_i \langle x_i - u, x_i - x_{i-1} \rangle - 2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle + 2\alpha_i \langle x_i - x_{i+1}, x_i - V(x_i) \rangle \\
&= -2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle - \theta_i \|x_{i-1} - u\|^2 + \theta_i \|x_i - u\|^2 + \theta_i \|x_i - x_{i-1}\|^2 \\
&\quad - \alpha_i \|V(x_i) - x_{i+1}\|^2 + \alpha_i \|x_{i+1} - x_i\|^2 + \alpha_i \|x_i - V(x_i)\|^2.
\end{aligned} \tag{3.21}$$

It follows that

$$\begin{aligned}
-2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle &\geq \|x_{i+1} - u\|^2 - \|x_i - u\|^2 - \theta_i \|x_i - x_{i-1}\|^2 + \|x_i - x_{i+1}\|^2 \\
&\quad - \theta_i \|x_i - x_{i+1}\|^2 + \theta_i \|x_{i-1} - u\|^2 - \theta_i \|x_i - u\|^2 - \theta_i \|x_i - x_{i-1}\|^2 \\
&\quad + \alpha_i \|V(x_i) - x_{i+1}\|^2 - \alpha_i \|x_{i+1} - x_i\|^2 - \alpha_i \|x_i - V(x_i)\|^2 \\
&\geq \|x_{i+1} - u\|^2 - \|x_i - u\|^2 + 2\theta_{i+1} \|x_{i+1} - x_i\|^2 - 2\theta_i \|x_i - x_{i-1}\|^2 \\
&\quad + \theta_i (\|x_{i-1} - u\|^2 - \|x_i - u\|^2) + \alpha_i (\|V(x_i) - x_{i+1}\|^2 - \|x_i - V(x_i)\|^2) \\
&\quad + (1 - \theta_i - 2\theta_{i+1} - \alpha_i) \|x_{i+1} - x_i\|^2.
\end{aligned} \tag{3.22}$$

Since  $\theta_i$  is non-decreasing and  $\alpha_i$  is non-increasing, we then obtain

$$\begin{aligned}
-2\alpha_i \langle x_i - u, x_i - V(x_i) \rangle &\geq \|x_{i+1} - u\|^2 - \|x_i - u\|^2 + 2\theta_{i+1} \|x_{i+1} - x_i\|^2 - 2\theta_i \|x_i - x_{i-1}\|^2 \\
&\quad + \alpha_{i+1} \|V(x_i) - x_{i+1}\|^2 - \alpha_i \|x_i - V(x_i)\|^2 - \theta_i \|x_i - u\|^2 \\
&\quad + \theta_{i-1} \|x_{i-1} - u\|^2 + (1 - 3\theta_{i+1} - \alpha_i) \|x_i - x_{i+1}\|^2.
\end{aligned}$$

□

**Lemma 3.5.** Assume that Conditions 2.6 and 3.2 hold. Then  $\{x_i\}$  generated by Algorithm 3.1 is bounded.

*Proof.* From (3.15) and Condition 3.2 (ii), there exists  $K > 0$  such that

$$\begin{aligned}
\|x_{i+1} - u\| &\leq \|w_i - u\| \\
&= \|\alpha_i V(x_i) + (1 - \alpha_i)x_i + \theta_i(x_i - x_{i-1}) - u\| \\
&\leq \alpha_i \|V(x_i) - u\| + (1 - \alpha_i) \|x_i - u\| + \theta_i \|x_i - x_{i-1}\| \\
&= \alpha_i \|V(x_i) - u\| + (1 - \alpha_i) \|x_i - u\| + \alpha_i \frac{\theta_i}{\alpha_i} \|x_i - x_{i-1}\|
\end{aligned} \tag{3.23}$$

$$\begin{aligned}
&\leq \alpha_i \|V(x_i) - u\| + (1 - \alpha_i) \|x_i - u\| + \alpha_i K \\
&\leq \alpha_i (\|V(x_i) - V(u)\| + \|V(u) - u\|) + (1 - \alpha_i) \|x_i - u\| + \alpha_i K
\end{aligned} \tag{3.24}$$

$$\leq (1 - \alpha_i(1 - \alpha)) \|x_i - u\| + \alpha_i(1 - \alpha) \left( \frac{\|V(u) - u\| + K}{1 - \alpha} \right) \tag{3.25}$$

$$\leq \max\{\|x_i - u\|, \frac{\|V(u) - u\| + K}{1 - \alpha}\}. \tag{3.26}$$

This implies that  $\|x_{i+1} - u\| \leq \max\{\|x_1 - u\|, \frac{\|V(u) - u\| + K}{1 - \alpha}\}$ . This shows that  $\{x_i\}$  is bounded. □

**Lemma 3.6.** Assume that Conditions 2.6 and 3.2 hold. Let  $\{x_i\}$  be generated by Algorithm 3.1. For each  $i \geq 1$ , define

$$u_i = \|x_i - u\|^2 - \theta_{i-1} \|x_{i-1} - u\|^2 + 2\theta_i \|x_i - x_{i-1}\|^2 + \alpha_i \|x_i - V(x_i)\|^2.$$

Then  $u_i \geq 0$ .

*Proof.* Since  $\{\theta_i\}$  is non-decreasing with  $0 \leq \theta_i < \frac{1}{3}$ , and  $2\langle x, y \rangle = \|x\|^2 + \|y\|^2 - \|x - y\|^2$  for all  $x, y \in H$ , we have

$$\begin{aligned}
 u_i &= \|x_i - u\|^2 - \theta_{i-1}[\|x_{i-1} - x_i\|^2 + \|x_i - u\|^2 + 2\langle x_{i-1} - x_i, x_i - u \rangle] \\
 &\quad + 2\theta_i\|x_i - x_{i-1}\|^2 + \alpha_i\|x_i - V(x_i)\|^2 \\
 &= \|x_i - u\|^2 - \theta_{i-1}[2\|x_{i-1} - x_i\|^2 + 2\|x_i - u\|^2 - \|x_{i-1} - 2x_i + u\|^2] \\
 &\quad + 2\theta_i\|x_i - x_{i-1}\|^2 + \alpha_i\|x_i - V(x_i)\|^2 \\
 &\geq \|x_i - u\|^2 - 2\theta_i\|x_{i-1} - x_i\|^2 - \frac{2}{3}\|x_i - u\|^2 + \theta_{i-1}\|x_{i-1} - 2x_i + u\|^2 \\
 &\quad + 2\theta_i\|x_i - x_{i-1}\|^2 + \alpha_i\|x_i - V(x_i)\|^2 \\
 &\geq \frac{1}{3}\|x_i - u\|^2 + \alpha_i\|x_i - V(x_i)\|^2 \\
 &\geq 0.
 \end{aligned}$$

This completes the proof.  $\square$

**Lemma 3.7.** Assume that Conditions 2.6 and 3.2 hold. Let  $\{x_i\}$  be generated by Algorithm 3.1. Suppose

$$\lim_{i \rightarrow \infty} \|x_{i+1} - x_i\| = 0,$$

and

$$\lim_{i \rightarrow \infty} (\|x_{i+1} - u\|^2 - \theta_i\|x_i - u\|^2) = 0.$$

Then  $\{x_i\}$  converges strongly to  $u \in EP(f, \mathcal{K})$ .

*Proof.* By our assumptions, we have

$$\begin{aligned}
 0 &= \lim_{i \rightarrow \infty} (\|x_{i+1} - u\|^2 - \theta_i\|x_i - u\|^2) \\
 &= \lim_{i \rightarrow \infty} [(\|x_{i+1} - u\| + \sqrt{\theta_i}\|x_i - u\|)(\|x_{i+1} - u\| - \sqrt{\theta_i}\|x_i - u\|)].
 \end{aligned} \tag{3.27}$$

In the case

$$\lim_{i \rightarrow \infty} (\|x_{i+1} - u\| + \sqrt{\theta_i}\|x_i - u\|) = 0,$$

this implies that  $\{x_i\}$  converges strongly to  $u$  immediately. Assume this limit does not hold. Then there is a subset  $N^* \subseteq N$  and a constant  $\rho > 0$  such that

$$\|x_{i+1} - u\| + \sqrt{\theta_i}\|x_i - u\| \geq \rho, \quad \forall i \in N^*. \tag{3.28}$$

Using (3.27) and  $\theta_i \leq \theta < 1$ . For  $i \in N^*$ , it then follows that

$$\begin{aligned}
 0 &= \lim_{i \rightarrow \infty} (\|x_{i+1} - u\| - \sqrt{\theta_i}\|x_i - u\|) \\
 &\geq \limsup_{i \rightarrow \infty} (\|x_i - u\| - \|x_{i+1} - x_i\| - \sqrt{\theta_i}\|x_i - u\|)
 \end{aligned}$$

$$\begin{aligned}
&\geq \limsup_{i \rightarrow \infty} ((1 - \sqrt{\theta})\|x_i - u\| - \|x_{i+1} - x_i\|) \\
&= (1 - \sqrt{\theta}) \limsup_{i \rightarrow \infty} \|x_i - u\| - \lim_{i \rightarrow \infty} \|x_{i+1} - x_i\| \\
&= (1 - \sqrt{\theta}) \limsup_{i \rightarrow \infty} \|x_i - u\|.
\end{aligned}$$

Consequently, we have  $\limsup_{i \rightarrow \infty} \|x_i - u\| \leq 0$ . Since  $\liminf_{i \rightarrow \infty} \|x_i - u\| \geq 0$  obviously holds, it follows that  $\lim_{i \rightarrow \infty} \|x_i - u\| = 0$ . This implies (by (3.28))

$$\begin{aligned}
\|x_{i+1} - x_i\| &\geq \|x_{i+1} - u\| - \|x_i - u\| \\
&= \|x_{i+1} - u\| + \sqrt{\theta_i}\|x_i - u\| - (1 + \sqrt{\theta_i})\|x_i - u\| \\
&\geq \frac{\rho}{2},
\end{aligned}$$

for all  $i \in N^*$  sufficiently large, a contradiction to the assumption that  $\lim_{i \rightarrow \infty} \|x_{i+1} - x_i\| = 0$ . This completes the proof.  $\square$

We now give the following strong convergence result of Algorithm 3.1.

**Theorem 3.8.** Assume that Conditions 2.6 and 3.2 hold. Then  $\{x_i\}$  generated by Algorithm 3.1 strongly converges to the solution  $u = P_{EP(f, \mathcal{K})}V(u)$ .

*Proof.* From Lemma 3.6 and (3.16), we have

$$\begin{aligned}
&u_{i+1} - u_i - \alpha_{i+1}\|x_{i+1} - V(x_{i+1})\|^2 + \alpha_{i+1}\|V(x_i) - x_{i+1}\|^2 + (1 - 3\theta_{i+1} - \alpha_i)\|x_i - x_{i+1}\|^2 \\
&\leq -2\alpha_i\langle x_i - u, x_i - V(x_i) \rangle.
\end{aligned} \tag{3.29}$$

Since  $P_{EP(f, \mathcal{K})}V$  is contraction, by the Banach fixed point theorem, there exist unique  $u = P_{EP(f, \mathcal{K})}V(u)$ . It follow from Lemma 3.3 that

$$\begin{aligned}
\|x_{i+1} - u\|^2 &\leq \|w_i - u\|^2 \\
&= \|\alpha_i(V(x_i) - u) + (1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1})\|^2 \\
&\leq \|(1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1})\|^2 + 2\langle \alpha_i(V(x_i) - u), w_i - u \rangle \\
&= \|(1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1})\|^2 + 2\alpha_i\langle V(x_i) - V(u), w_i - u \rangle + 2\alpha_i\langle V(u) - u, w_i - u \rangle \\
&\leq \|(1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1})\|^2 + 2\alpha_i\langle V(u) - u, w_i - u \rangle + 2\alpha_i\alpha\|x_i - u\|\|w_i - u\| \\
&\leq \|(1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1})\|^2 + 2\alpha_i\langle V(u) - u, w_i - u \rangle + \alpha_i\alpha(\|x_i - u\|^2 + \|w_i - u\|^2) \\
&\leq \frac{1}{1 - \alpha_i\alpha} \left( \|(1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1})\|^2 + \alpha_i\alpha\|x_i - u\|^2 + 2\alpha_i\langle V(u) - u, w_i - u \rangle \right) \\
&\leq \frac{1}{1 - \alpha_i\alpha} \left( \|(1 - \alpha_i)(x_i - u)\|^2 + 2\langle \theta_i(x_i - x_{i-1}), (1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1}) \rangle \right. \\
&\quad \left. + \alpha_i\alpha\|x_i - u\|^2 + 2\alpha_i\langle V(u) - u, w_i - u \rangle \right) \\
&= \frac{(1 - \alpha_i)^2 + \alpha_i\alpha}{1 - \alpha_i\alpha} \|x_i - u\|^2 + \frac{1}{1 - \alpha_i\alpha} \left( 2\langle \theta_i(x_i - x_{i-1}), (1 - \alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1}) \rangle \right.
\end{aligned}$$

$$\begin{aligned}
& + 2\alpha_i \langle V(u) - u, w_i - u \rangle \Big) \\
& = \left(1 - \left(\frac{2\alpha_i(1-\alpha)}{1-\alpha_i\alpha} - \frac{(\alpha_i)^2}{1-\alpha_i\alpha}\right)\right) \|x_i - u\|^2 + \frac{1}{1-\alpha_i\alpha} \Big( 2\langle \theta_i(x_i - x_{i-1}), (1-\alpha_i)(x_i - u) \\
& \quad + \theta_i(x_i - x_{i-1}) \rangle + 2\alpha_i \langle V(u) - u, w_i - u \rangle \Big) \\
& \leq \left(1 - \frac{2\alpha_i(1-\alpha)}{1-\alpha_i\alpha}\right) \|x_i - u\|^2 + \frac{2\alpha_i(1-\alpha)}{1-\alpha_i\alpha} \Big( \frac{\alpha_i}{2(1-\alpha)} \|x_i - u\|^2 \\
& \quad + \frac{1}{\alpha_i(1-\alpha)} \langle \theta_i(x_i - x_{i-1}), (1-\alpha_i)(x_i - u) + \theta_i(x_i - x_{i-1}) \rangle \\
& \quad + \frac{1}{1-\alpha} \langle V(u) - u, w_i - u \rangle \Big). \tag{3.30}
\end{aligned}$$

We will consider into 2 cases.

**Case 1.** In the case of  $u_{i+1} \leq u_i + t_i$  for all  $i \geq i_0$  for some  $i_0 \in \mathbb{N}$ ,  $t_i \geq 0$  and  $\sum_{i=1}^{\infty} t_i < +\infty$ . Then  $u_i \geq 0$ ,  $\forall i \geq 1$  by Lemma 2.5, we have  $\lim_{i \rightarrow \infty} u_i = \lim_{i \rightarrow \infty} u_{i+1}$  exists. Since  $\{x_i\}$  is bounded by Lemma 3.5, there exists  $M_1 > 0$  such that  $2|\langle x_i - u, x_i - V(x_i) \rangle| \leq M_1$  and  $M_2 > 0$  such that  $\|x_{i+1} - V(x_{i+1})\|^2 + \|V(x_i) - x_{i+1}\|^2 \leq M_2$ . Since  $0 \leq \theta_i \leq \theta_{i+1} \leq \theta < \frac{1}{3}$  and  $\lim_{i \rightarrow \infty} \alpha_i = 0$ , there exist  $N \in \mathbb{N}$  and  $\gamma_1 > 0$  such that  $1 - 3\theta_{i+1} - \alpha_i \geq \gamma_1$  for all  $i \geq N$ . Therefore, for  $i \geq N$ , we obtain from (3.29) that

$$\gamma_1 \|x_{i+1} - x_i\|^2 \leq \alpha_i M_1 + \alpha_{i+1} M_2 + u_i - u_{i+1} \rightarrow 0, \tag{3.31}$$

as  $i \rightarrow \infty$ . Hence  $\lim_{i \rightarrow \infty} \|x_{i+1} - x_i\| = 0$ . For  $u \in EP(f, \mathcal{K})$ , we have

$$\begin{aligned}
\|w_i - u\|^2 &= \|\alpha_i V(x_i) + (1 - \alpha_i)x_i + \theta_i(x_i - x_{i-1}) - u\|^2 \\
&\leq \|\alpha_i V(x_i) + (1 - \alpha_i)x_i - u\|^2 + 2\langle \theta_i(x_i - x_{i-1}), w_i - u \rangle \\
&\leq \alpha_i \|V(x_i) - u\|^2 + (1 - \alpha_i) \|x_i - u\|^2 + 2\theta_i \|x_i - x_{i-1}\| \|w_i - u\| \\
&\leq \alpha_i \|V(x_i) - u\|^2 + (1 - \alpha_i) \|x_i - u\|^2 + 2\frac{\theta_i}{\alpha_i} \|x_i - x_{i-1}\| \|w_i - u\| \\
&\leq \alpha_i \|V(x_i) - u\|^2 + \|x_i - u\|^2 + 2\frac{\theta_i}{\alpha_i} \|x_i - x_{i-1}\| \|w_i - u\|, \tag{3.32}
\end{aligned}$$

and from (3.14), we have

$$\begin{aligned}
\|x_{i+1} - u\|^2 &= \|w_i - u\|^2 - \tau(1 - 2\lambda_i c_1) \|w_i - y_i\|^2 - \tau(1 - 2\lambda_i c_2) \|z_i - y_i\|^2 \\
&\quad - (1 - \tau)\tau \frac{1}{\tau^2} \|x_{i+1} - w_i\|^2 \\
&\leq \alpha_i \|V(x_i) - u\|^2 + \|x_i - u\|^2 + 2\frac{\theta_i}{\alpha_i} \|x_i - x_{i-1}\| \|w_i - u\| - \tau(1 - 2\lambda_i c_1) \|w_i - y_i\|^2 \\
&\quad - \tau(1 - 2\lambda_i c_2) \|z_i - y_i\|^2 - \frac{1 - \tau}{\tau} \|x_{i+1} - w_i\|^2. \tag{3.33}
\end{aligned}$$

This implies that

$$\begin{aligned}
&\tau(1 - 2\lambda_i c_1) \|w_i - y_i\|^2 + \tau(1 - 2\lambda_i c_2) \|z_i - y_i\|^2 + \frac{1 - \tau}{\tau} \|x_{i+1} - w_i\|^2 \\
&\leq \alpha_i \|V(x_i) - u\|^2 + \|x_i - u\|^2 + 2\frac{\theta_i}{\alpha_i} \|x_i - x_{i-1}\| \|w_i - u\| - \|x_{i+1} - u\|^2. \tag{3.34}
\end{aligned}$$

By our condition and (3.31), we obtain

$$\lim_{i \rightarrow \infty} \|w_i - y_i\| = \lim_{i \rightarrow \infty} \|z_i - y_i\| = \lim_{i \rightarrow \infty} \|x_{i+1} - w_i\| = 0. \quad (3.35)$$

Since  $\{x_i\}$  is bounded, that is, there exists a subsequence  $\{x_{i_k}\}$  of  $\{x_i\}$  such that  $x_{i_k} \rightharpoonup x^*$  for some  $x^* \in H$ . From (3.31) and (3.35), we get  $w_{i_k} \rightarrow x^*$  and  $y_{i_k} \rightarrow x^*$  as  $k \rightarrow \infty$ .

By the definition of  $z_i$  and (3.6), we have

$$\begin{aligned} \lambda_{i_k} f(y_{i_k}, y) &\geq \lambda_{i_k} f(y_{i_k}, z_{i_k}) + \langle w_{i_k} - z_{i_k}, y - z_{i_k} \rangle \\ &\geq \lambda_{i_k} f(w_{i_k}, z_{i_k}) - \lambda_{i_k} f(w_{i_k}, y_{i_k}) - c_1 \|w_{i_k} - y_{i_k}\|^2 - c_2 \|z_{i_k} - y_{i_k}\|^2 \\ &\quad + \langle w_{i_k} - z_{i_k}, y - z_{i_k} \rangle \\ &\geq \langle y_{i_k} - w_{i_k}, y_{i_k} - z_{i_k} \rangle + \langle w_{i_k} - z_{i_k}, y - z_{i_k} \rangle - c_1 \|w_{i_k} - y_{i_k}\|^2 - c_2 \|z_{i_k} - y_{i_k}\|^2. \end{aligned}$$

It follows from  $\{z_{i_k}\}$  is bounded,  $0 < \lambda_{i_k} \leq \lambda < \frac{1}{2 \max\{c_1, c_2\}}$  and Condition 2.6 (A3) that  $0 \leq \limsup_{k \rightarrow \infty} f(y_{i_k}, y) \leq f(x^*, y)$  for all  $y \in H$ . This implies that  $f(x^*, y) \geq 0$  for all  $y \in \mathcal{K}$ , this shows that  $x^* \in EP(f, \mathcal{K})$ . Then, we have

$$\begin{aligned} \limsup_{i \rightarrow \infty} \langle V(u) - u, w_i - u \rangle &= \lim_{k \rightarrow \infty} \langle V(u) - u, w_{i_k} - u \rangle \\ &= \langle V(u) - u, x^* - u \rangle \leq 0, \end{aligned} \quad (3.36)$$

by  $u = P_{EP(f, \mathcal{K})} V(u)$ . Applying (3.36) to the inequality (3.30) with Lemma 2.4, we can conclude that  $x_i \rightarrow u = P_{EP(f, \mathcal{K})} V(u)$  as  $i \rightarrow \infty$ .

**Case 2.** In another case of  $\{u_i\}$ , we let  $\phi : \mathbb{N} \rightarrow \mathbb{N}$  be the map defined for all  $i \geq i_0$  (for some  $i_0 \in \mathbb{N}$  large enough) by

$$\phi(i) = \max\{k \in \mathbb{N} : k \leq i, u_k + t_k \leq u_{k+1}\}. \quad (3.37)$$

Clearly,  $\phi(i)$  is a non-decreasing sequence such that  $\phi(i) \rightarrow \infty$  for  $i \rightarrow \infty$  and  $u_{\phi(i)} + t_{\phi(i)} \leq u_{\phi(i)+1}$  for all  $i \geq i_0$ . Hence, similar to the proof of Case 1, we therefore obtain from (3.31) that

$$\gamma_1 \|x_{\phi(i)+1} - x_{\phi(i)}\|^2 \leq \alpha_{\phi(i)} M_1 + \alpha_{\phi(i)+1} M_2 + u_{\phi(i)} - u_{\phi(i)+1} \rightarrow 0 \quad (3.38)$$

for some constant  $M_1, M_2 > 0$ . Thus

$$\lim_{i \rightarrow \infty} \|x_{\phi(i)+1} - x_{\phi(i)}\| = 0. \quad (3.39)$$

By the same proof of Case 1, one also derive

$$\lim_{i \rightarrow \infty} \|x_{\phi(i)+1} - w_{\phi(i)}\| = \lim_{i \rightarrow \infty} \|w_{\phi(i)} - x_{\phi(i)}\| = \lim_{i \rightarrow \infty} \|x_{\phi(i)} - z_{\phi(i)}\| = 0. \quad (3.40)$$

Again observe that for  $j \geq 0$  by (3.29), we have  $u_{j+1} < u_j + t_j$  when  $x_j \notin \Omega = \{x \in H : \langle x - x_0, x - u \rangle \leq 0\}$ . Hence  $x_{\phi(i)} \in \Omega$  for all  $i \geq i_0$  since  $u_{\phi(i)} + t_{\phi(i)} \leq u_{\phi(i)+1}$ . Since  $\{x_{\phi(i)}\}$  is bounded, there exist subsequence  $\{x_{\phi(i_k)}\}$  of  $\{x_{\phi(i)}\}$  which converges weakly to some  $x^* \in \mathcal{H}$ . As  $\Omega$  is a closed and convex set, it is then weakly closed and so  $x^* \in \Omega$ . Using (3.40), one can see as in Case 1 that  $z_{\phi(i_k)} \rightharpoonup x^*$  and  $x^* \in EP(f, \mathcal{K})$  contains  $u$  as its only element. We therefore have  $x^* = u$ . Furthermore,

$$\begin{aligned} \|x_{\phi(i)} - u\|^2 &= \langle x_{\phi(i)} - V(x_i), x_{\phi(i)} - u \rangle - \langle u - V(x_i), x_{\phi(i)} - u \rangle \\ &\leq -\langle u - V(x_i), x_{\phi(i)} - u \rangle, \end{aligned}$$

due to  $x_{\phi(i)} \in \Omega$ . This gives

$$\limsup_{i \rightarrow \infty} \|x_{\phi(i)} - u\| \leq 0.$$

Hence

$$\lim_{i \rightarrow \infty} \|x_{\phi(i)} - u\| = 0. \quad (3.41)$$

By definition,  $u_{\phi(i)+1}$ , we have

$$\begin{aligned} u_{\phi(i)+1} &= \|x_{\phi(i)+1} - u\|^2 - \theta_{\phi(i)} \|x_{\phi(i)} - u\|^2 + 2\theta_{\phi(i)+1} \|x_{\phi(i)+1} - x_{\phi(i)}\|^2 + \alpha_{\phi(i)+1} \|x_{\phi(i)+1} - V_{\phi(i)+1}\|^2 \\ &\leq (\|x_{\phi(i)+1} - x_{\phi(i)}\| + \|x_{\phi(i)} - u\|)^2 - \theta_{\phi(i)} \|x_{\phi(i)} - u\|^2 + 2\theta_{\phi(i)+1} \|x_{\phi(i)+1} - x_{\phi(i)}\|^2 \\ &\quad + \alpha_{\phi(i)+1} \|x_{\phi(i)+1} - V_{\phi(i)+1}\|^2. \end{aligned} \quad (3.42)$$

By our Condition 3.2 (i), (3.39) and (3.41), we obtain  $\lim_{i \rightarrow \infty} u_{\phi(i)+1} = 0$ . We next show that we actually have  $\lim_{i \rightarrow \infty} u_i = 0$ . To this end, first observe that, for  $i \geq i_0$ , one has  $u_i + t_i \leq u_{\phi(i)+1}$  if  $i \neq \phi(i)$ . It follows that for all  $i \geq i_0$ , we have  $u_i \leq \max\{u_{\phi(i)}, u_{\phi(i)+1}\} = u_{\phi(i)+1} \rightarrow 0$ , since  $\lim_{i \rightarrow \infty} t_i = 0$ , hence  $\limsup_{i \rightarrow \infty} u_i \leq 0$ . On the other hand, Lemma 3.6 implies that  $\liminf_{i \rightarrow \infty} u_i \geq 0$ . Hence, we obtain  $\lim_{i \rightarrow \infty} u_i = 0$ . Consequently, the boundedness of  $\{x_i\}$ ,  $\lim_{i \rightarrow \infty} \alpha_i = 0$ , and (3.29) show that  $\|x_i - x_{i+1}\| \rightarrow 0$ , as  $i \rightarrow \infty$ . Hence the definition of  $u_i$  yields  $(\|x_{i+1} - u\|^2 - \theta_i \|x_i - u\|^2) \rightarrow 0$ , as  $i \rightarrow \infty$ . By using Lemma 3.7, we obtain the desired conclusion immediately.  $\square$

Setting  $V(x) = x_0$ ,  $\forall x \in \mathcal{H}$ , then we obtain the following modified Halpern inertial extragradient algorithm for EPs:

---

**Algorithm 3.9.** (Modified Halpern inertial extragradient algorithm for EPs)

---

**Initialization.** Let  $x_0, x_1 \in H$ ,  $0 < \lambda_i \leq \lambda < \frac{1}{2 \max\{c_1, c_2\}}$ ,  $\tau \in (0, \frac{1}{2}]$ .

**Step 1.** Given the current iterates  $x_{i-1}$  and  $x_i$  ( $i \geq 1$ ) and  $\alpha_i \in (0, 1)$ ,  $\theta_i \in [0, \frac{1}{3})$ , compute

$$\begin{aligned} w_i &= \alpha_i x_0 + (1 - \alpha_i) x_i + \theta_i (x_i - x_{i-1}), \\ y_i &= \arg \min_{y \in \mathcal{K}} \{ \lambda_i f(w_i, y) + \frac{1}{2} \|y - w_i\|^2 \}. \end{aligned}$$

If  $y_i = w_i$  then stop and  $y_i$  is a solution. Otherwise, go to **Step 2**.

**Step 2.** Compute

$$z_i = \arg \min_{y \in \mathcal{K}} \{ \lambda_i f(y_i, y) + \frac{1}{2} \|y - w_i\|^2 \}.$$

**Step 3.** Compute

$$x_{i+1} = (1 - \tau) w_i + \tau z_i.$$

Set  $i = i + 1$  and return to **Step 1**.

---

From Algorithm 3.1, the convergence depends on the parameter  $\{\lambda_i\}$  with the condition  $0 < \lambda_i \leq \lambda < \frac{1}{2 \max\{c_1, c_2\}}$ . So, the step size  $\{\lambda_i\}$  can be considered in many ways. Applying step size concept of Shehu et al. [22], we then obtain the following modified viscosity type inertial extragradient stepsize algorithm for EPs:

---

**Algorithm 3.10.** (Modified viscosity type inertial extragradient stepsize algorithm for EPs)

---

**Initialization.** Let  $x_0, x_1 \in H$ ,  $\lambda_1 \in (0, \frac{1}{2 \max\{c_1, c_2\}})$ ,  $\mu \in (0, 1)$ ,  $\tau \in (0, \frac{1}{2}]$ .

**Step 1.** Given the current iterates  $x_{i-1}$  and  $x_i (i \geq 1)$  and  $\alpha_i \in (0, 1)$ ,  $\theta_i \in [0, \frac{1}{3})$ , compute

$$w_i = \alpha_i V(x_i) + (1 - \alpha_i)x_i + \theta_i(x_i - x_{i-1}),$$

$$y_i = \arg \min_{y \in \mathcal{K}} \{ \lambda_i f(w_i, y) + \frac{1}{2} \|y - w_i\|^2 \}.$$

If  $y_i = w_i$  then stop and  $y_i$  is a solution. Otherwise, go to **Step 2**.

**Step 2.** Compute

$$z_i = \arg \min_{y \in \mathcal{K}} \{ \lambda_i f(y_i, y) + \frac{1}{2} \|y - w_i\|^2 \}.$$

**Step 3.** Compute

$$x_{i+1} = (1 - \tau)w_i + \tau z_i,$$

and

$$\lambda_{i+1} = \begin{cases} \min\{ \frac{\mu}{2} \frac{\|w_i - y_i\|^2 + \|z_i - y_i\|^2}{f(w_i, z_i) - f(w_i, y_i) - f(y_i, z_i)}, \lambda_i \}, & \text{if } f(w_i, z_i) - f(w_i, y_i) - f(y_i, z_i) > 0, \\ \lambda_i, & \text{Otherwise.} \end{cases}$$

Set  $i = i + 1$  and return to **Step 1**.

---

**Remark 3.11.** (i) Since  $V(x) = x_0$ ,  $\forall x \in \mathcal{H}$  is contraction, thus the modified Halpern inertial extragradient Algorithm 3.9 converges strongly to  $x^* = P_{EP(f, \mathcal{K})} x_0$  with Conditions 2.6 and 3.2;

(ii) Since the step size  $\{\lambda_i\}$  in Algorithm 3.10 is a monotonically decreasing sequence with lower bound  $\min\{\lambda_1, \frac{1}{2 \max\{c_1, c_2\}}\}$  [22], thus Algorithm 3.10 converges strongly to the solution  $u = P_{EP(f, \mathcal{K})} V(u)$  by Theorem 3.8.

We now give an example in infinitely dimensional spaces  $L_2[0, 1]$  to support the main theorem.

**Example 3.12.** Let  $V : L_2[0, 1] \rightarrow L_2[0, 1]$  be defined by  $V(x(t)) = \frac{x(t)}{2}$  where  $x(t) \in L_2[0, 1]$ . We can choose  $x_0(t) = \frac{\sin(t)}{2}$  and  $x_1(t) = \sin(t)$ . The stopping criterion is defined by  $\|x_i - x_{i-1}\| < 10^{-2}$ .

We set the following parameters for each algorithm, as seen in Table 1.

**Table 1.** Chosen parameters of each algorithm.

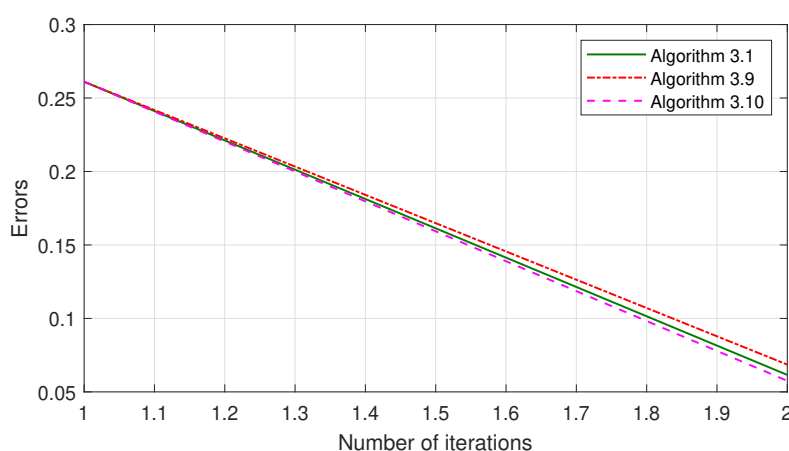
|             | Algorithm 3.1      | Algorithm 3.9      | Algorithm 3.10     |
|-------------|--------------------|--------------------|--------------------|
| $\lambda_i$ | 0.1                | 0.1                | -                  |
| $\lambda_1$ | -                  | -                  | 0.12               |
| $\theta_i$  | 0.29               | 0.29               | 0.29               |
| $\alpha_i$  | $\frac{1}{100i+1}$ | $\frac{1}{100i+1}$ | $\frac{1}{100i+1}$ |
| $\tau_i$    | 0.15               | 0.1                | 0.15               |
| $\mu$       | -                  | -                  | 0.2                |

Next, we compare the performance of Algorithms 3.1, 3.9 and 3.10. We obtain the results as seen in Table 2.

**Table 2.** The performance of each algorithm.

|           | Algorithm 3.1 | Algorithm 3.9 | Algorithm 3.10 |
|-----------|---------------|---------------|----------------|
| CPU Time  | 1.2626        | 1.2010        | 177.9459       |
| Iter. No. | 2             | 2             | 2              |

From Figure 1, we see that the performance of Algorithm 3.10 is better than Algorithms 3.1 and 3.9.

**Figure 1.** The Cauchy error plotting number of iterations.

#### 4. Application to data classification problem

According to the International Diabetes Federation (IDF), there are approximately 463 million people with diabetes worldwide, and it is estimated that by 2045 there will be 629 million people with diabetes. In Thailand, the incidence of diabetes is continuously increasing. There are about 300,000 new cases per year, and 3.2 million people with diabetes are registered in the Ministry of Public Health's registration system. They are causing huge losses in health care costs. Only one disease of diabetes causes the average cost of treatment costs up to 47,596 million baht per year. This

has led to an ongoing campaign about the dangers of the disease. Furthermore, diabetes mellitus makes additional noncommunicable diseases that present a high risk for the patient, as they easily contact and are susceptible to infectious diseases such as COVID-19 [23]. Because it is a chronic disease that cannot be cured. There is a chance that the risk of complications spreading to the extent of the loss of vital organs of the body. By the International Diabetes Federation and the World Health Organization (WHO) has designated November 14 of each year as World Diabetes Day to recognize the importance of this disease.

In this research, we used the PIMA Indians diabetes dataset which was downloaded from Kaggle (<https://www.kaggle.com/uciml/pima-indians-diabetesdatabase>) and is available publicly on UCI repository for training processing by our proposed algorithm. The dataset contains 768 pregnant female patients which 500 were non-diabetics and 268 were diabetics. There were 9 variables present inside the dataset; eight variables contain information about patients, and the 9th variable is the class predicting the patients as diabetic and nondiabetic. This dataset contains the various attributes that are Number of times pregnant; Plasma glucose concentration at 2 Hours in an oral glucose tolerance test (GTIT); Diastolic Blood Pressure (mm Hg); Triceps skin fold thickness (mm); 2-Hour Serum insulin (lh/ml); Body mass index [weight in kg/(Height in m)]; Diabetes pedigree function; Age (years); Binary value indicating non-diabetic /diabetic. For the implementation of machine learning algorithms, 614 were used as a training dataset and 154 were used as a testing training dataset by using 5-fold cross-validation [12]. For benchmarking classifier, we consider the following various methods which have been proposed to classify diabetes (see Table 3):

**Table 3.** Classification accuracy of different methods with literature.

| Authors                         | Methods                               | Accuracy (%) |
|---------------------------------|---------------------------------------|--------------|
| Li [13]                         | Ensemble of SVM, ANN, and NB          | 58.3         |
| Brahim-Belhouari and Bermak [4] | NB, SVM, DT                           | 76.30        |
| Smith et al. [24]               | Neural ADAP algorithm                 | 76           |
| Quinlan [17]                    | C4.5 Decision trees                   | 71.10        |
| Bozkurt et al. [3]              | Artificial neural network             | 76.0         |
| Sahan et al. [20]               | Artificial immune System              | 75.87        |
| Smith et al. [24]               | Ensemble of MLP and NB                | 64.1         |
| Chatrati et al. [5]             | Linear discriminant analysis          | 72           |
| Deng and Kasabov [7]            | Self-organizing maps                  | 78.40        |
| Deng and Kasabov [7]            | Self-organizing maps                  | 78.40        |
| Choubey et al. [6]              | Ensemble of RF and XB                 | 78.9         |
| Saxena et al. [21]              | Feature selection of KNN, RF, DT, MLP | 79.8         |
| Our Algorithm 3.1               | Extreme learning machine              | <b>80.03</b> |

We focus on extreme learning machine (ELM) proposed by Huang et al. [9] for applying our algorithms to solve data classification problems. It is defined as follows:

Let  $E := \{(\mathbf{x}_n, \mathbf{t}_n) : \mathbf{x}_n \in \mathbb{R}^n, \mathbf{t}_n \in \mathbb{R}^m, n = 1, 2, \dots, P\}$  be a training set of  $P$  distinct samples where  $\mathbf{x}_n$  is an input training data and  $\mathbf{t}_n$  is a training target. The output function of ELM for single-hidden layer feed forward neural networks (SLFNs) with  $M$  hidden nodes and activation function  $U$  is

$$\mathbf{O}_n = \sum_{j=1}^M \Theta_j U(w_j, b_j, \mathbf{x}_n),$$

where  $w_j$  and  $b_j$  are parameters of weight and finally the bias, respectively and  $\Theta_j$  is the optimal output weight at the  $j$ -th hidden node. The hidden layer output matrix  $H$  is defined as follows:

$$H = \begin{bmatrix} U(w_1, b_1, \mathbf{x}_1) & \dots & U(w_M, b_M, \mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ U(w_1, b_1, \mathbf{x}_P) & \dots & U(w_M, b_M, \mathbf{x}_P) \end{bmatrix}$$

To solve ELM is to find optimal output weight  $\Theta = [\Theta_1^T, \dots, \Theta_M^T]^T$  such that  $H\Theta = T$ , where  $T = [\mathbf{t}_1^T, \dots, \mathbf{t}_P^T]^T$  is the training data. In some cases, finding  $\Theta = H^\dagger T$ , where  $H^\dagger$  is the *Moore-Penrose generalized inverse* of  $H$ . However, if  $H$  does not exist, then, finding such a solution  $\Theta$  through convex minimization can overcome such difficulty.

In this section, we process some experiments on the classification problem. This problem can be seen as the following convex minimization problem:

$$\min_{\Theta \in \mathbb{R}^M} \{ \|H\Theta - T\|_2^2 + \lambda \|\Theta\|_1 \}, \quad (4.1)$$

where  $\lambda$  is a regularization parameter. This problem is called the least absolute shrinkage and selection operator (LASSO) [26]. Setting  $f(\Theta, \zeta) = \langle H^T(H\Theta - T), \zeta - \Theta \rangle$  and  $V(x) = Cx$  where  $C$  is constant in  $(0, 1)$ .

The binary cross-entropy loss function along with sigmoid activation function for binary classification calculates the loss of an example by computing the following average:

$$Loss = -\frac{1}{K} \sum_{j=1}^K y_j \log \hat{y}_j + (1 - y_j) \log(1 - \hat{y}_j)$$

where  $\hat{y}_j$  is the  $j$ -th scalar value in the model output,  $y_j$  is the corresponding target value, and  $K$  is the number of scalar values in the model output.

In this work, the performance of machine learning techniques for all classes is accurately measured. The accuracy is calculated by adding the total number of correct predictions to the total number of predictions. The performance parameter calculation of precision and recall are measured. The formulation of three measures [27] are defined as follow:

$$\text{Precision(Pre)} = \frac{TP}{TP + FP}. \quad (4.2)$$

$$\text{Recall(Rec)} = \frac{TP}{TP + FN}. \quad (4.3)$$

$$\text{Accuracy(Acc)} = \frac{TP + TN}{TP + FP + TN + FN} \times 100\%, \quad (4.4)$$

where a confusion matrix for original and predicted classes are shown in terms of TP, TN, FP, and FN are the True Positive, True Negative, False Positives, and False Negatives, respectively. Similarly, P and N are the Positive and Negative population of Malignant and Benign cases, respectively.

For starting our computation, we set the activation function as sigmoid, hidden nodes  $M = 160$ , regularization parameter  $\lambda = 1 \times 10^{-5}$ ,  $\theta_i = 0.3$ ,  $\alpha_i = \frac{1}{i+1}$ ,  $\tau = 0.5$ ,  $\mu = 0.2$  for Algorithms 3.1, 3.9 and 3.10 and  $C = 0.9999$  for Algorithms 3.1 and 3.10. The stopping criteria is the number of iteration 250. We obtain the results of the different parameters  $S$  when  $\lambda_i = \frac{S}{\max(\text{eigenvalue}(A^T A))}$  for Algorithms 3.1, 3.9 and the different parameters  $\lambda_1$  for Algorithm 3.10 as seen in Table 4.

**Table 4.** Training and validation loss and training time of the different parameter  $\lambda_i$  and  $\lambda_1$ .

|                | $S, \lambda_1$ | Training Time | Loss     |            |
|----------------|----------------|---------------|----------|------------|
|                |                |               | Training | Validation |
| Algorithm 3.1  | 0.2            | 0.4164        | 0.286963 | 0.275532   |
|                | 0.4            | 0.4337        | 0.283279 | 0.273650   |
|                | 0.6            | 0.4164        | 0.286963 | 0.275532   |
|                | 0.9            | 0.4459        | 0.278714 | 0.272924   |
|                | 0.99           | 0.4642        | 0.278144 | 0.272921   |
| Algorithm 3.9  | 0.2            | 0.4283        | 0.291883 | 0.279878   |
|                | 0.4            | 0.5293        | 0.288831 | 0.277365   |
|                | 0.6            | 0.4246        | 0.286890 | 0.276099   |
|                | 0.9            | 0.4247        | 0.284851 | 0.275079   |
|                | 0.99           | 0.5096        | 0.284356 | 0.274879   |
| Algorithm 3.10 | 0.2            | 1.3823        | 0.286963 | 0.275532   |
|                | 0.4            | 1.5652        | 0.283279 | 0.273650   |
|                | 0.6            | 1.4022        | 0.281060 | 0.273120   |
|                | 0.9            | 1.9170        | 0.278714 | 0.272924   |
|                | 0.99           | 1.3627        | 0.278144 | 0.272921   |

We can see that  $\lambda_i = \lambda_1 = 0.99$  greatly improves the performance of Algorithms 3.1, 3.9 and 3.10. Therefore, we choose it as the default inertial parameter for next computation.

We next choose  $\lambda_i = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\alpha_i = \frac{1}{i+1}$ ,  $\tau = 0.5$  for Algorithms 3.1 and 3.9 and  $C = 0.9999$  for Algorithm 3.1 with  $\lambda_1 = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\alpha_i = \frac{1}{i+1}$ ,  $\tau = 0.5$ ,  $C = 0.9999$ , and  $\mu = 0.2$  for Algorithm 3.10. The stopping criteria is the number of iteration 250. We consider the different initialization parameter  $\theta$  where

$$\theta_i = \begin{cases} \theta & \text{if } x_i = x_{i-1} \text{ and } i \leq N, \\ \frac{\theta}{i^2 \|x_i - x_{i-1}\|} & \text{otherwise,} \end{cases}$$

where  $N$  is a number of iterations that we want to stop. We obtain the numerical results as seen in Table 5.

**Table 5.** Training and validation loss and training time of the different parameter  $\theta$ .

|                | $\theta$                            | Training Time | Loss     |            |
|----------------|-------------------------------------|---------------|----------|------------|
|                |                                     |               | Training | Validation |
| Algorithm 3.1  | 0.1                                 | 0.4608        | 0.279629 | 0.272965   |
|                | 0.2                                 | 0.4515        | 0.278938 | 0.272931   |
|                | 0.3                                 | 0.4523        | 0.278144 | 0.272921   |
|                | $\frac{1}{i}$                       | 0.4591        | 0.280107 | 0.273004   |
|                | $\frac{1}{\ x_i - x_{i-1}\  + i^2}$ | 0.5003        | 0.280221 | 0.273015   |
| Algorithm 3.9  | 0.1                                 | 0.4723        | 0.284808 | 0.274993   |
|                | 0.2                                 | 0.4641        | 0.284587 | 0.274935   |
|                | 0.3                                 | 0.4634        | 0.284356 | 0.274879   |
|                | $\frac{1}{i}$                       | 0.5297        | 0.285004 | 0.275049   |
|                | $\frac{1}{\ x_i - x_{i-1}\  + i^2}$ | 0.4825        | 0.285019 | 0.275053   |
| Algorithm 3.10 | 0.1                                 | 1.4071        | 0.279629 | 0.272965   |
|                | 0.2                                 | 1.3505        | 0.278938 | 0.272931   |
|                | 0.3                                 | 1.4819        | 0.278144 | 0.272921   |
|                | $\frac{1}{i}$                       | 1.3276        | 0.280107 | 0.273004   |
|                | $\frac{1}{\ x_i - x_{i-1}\  + i^2}$ | 1.4228        | 0.280221 | 0.273015   |

We can see that  $\theta = 0.3$  greatly improves the performance of Algorithms 3.1, 3.9 and 3.10. Therefore, we choose it as the default inertial parameter for next computation.

We next set  $\lambda_i = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\theta_i = 0.3$ ,  $\tau = 0.5$  for Algorithms 3.1 and 3.9 and  $C = 0.9999$  for Algorithm 3.1 with  $\lambda_1 = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\theta_i = 0.3$ ,  $\tau = 0.5$ ,  $C = 0.9999$ , and  $\mu = 0.2$  for Algorithm 3.10. The stopping criteria is the number of iteration 250. We consider the different initialization parameter  $\alpha_i$ . The numerical results are shown in Table 6.

**Table 6.** Training and validation loss and training time of the different parameter  $\alpha_i$ .

|                | $\alpha_i$          | Training Time | Loss     |            |
|----------------|---------------------|---------------|----------|------------|
|                |                     |               | Training | Validation |
| Algorithm 3.1  | $\frac{1}{i+1}$     | 0.4407        | 0.278144 | 0.272921   |
|                | $\frac{1}{10i+1}$   | 0.4054        | 0.278143 | 0.272921   |
|                | $\frac{1}{i^2+1}$   | 0.4938        | 0.278143 | 0.272921   |
|                | $\frac{1}{10i^2+1}$ | 0.4876        | 0.278143 | 0.272921   |
|                | $\frac{1}{i+1}$     | 0.4163        | 0.284356 | 0.274879   |
| Algorithm 3.9  | $\frac{1}{10i+1}$   | 0.4274        | 0.279201 | 0.273129   |
|                | $\frac{1}{i^2+1}$   | 0.5150        | 0.278294 | 0.272931   |
|                | $\frac{1}{10i^2+1}$ | 0.5960        | 0.278160 | 0.272922   |
|                | $\frac{1}{i+1}$     | 1.4292        | 0.278144 | 0.272921   |
|                | $\frac{1}{10i+1}$   | 1.3803        | 0.278143 | 0.272921   |
| Algorithm 3.10 | $\frac{1}{i^2+1}$   | 1.2452        | 0.278143 | 0.272921   |
|                | $\frac{1}{10i^2+1}$ | 1.4100        | 0.278143 | 0.272921   |
|                | $\frac{1}{10i^2+1}$ |               |          |            |

We can see that  $\alpha_i = \frac{1}{10i+1}$  greatly improves the performance of Algorithm 3.1,  $\alpha_i = \frac{1}{10i^2+1}$  greatly improves the performance of Algorithm 3.9, and  $\alpha_i = \frac{1}{i^2+1}$  greatly improves the performance of Algorithm 3.10. Therefore, we choose it as the default inertial parameter for next computation.

We next calculate the numerical results by setting  $\lambda_i = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\theta_i = 0.3$ ,  $\alpha_i = \frac{1}{10i+1}$  and  $C = 0.9999$  for Algorithm 3.1,  $\lambda_i = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\theta_i = 0.3$ ,  $\alpha_i = \frac{1}{10i^2+1}$  for Algorithm 3.9 and  $\lambda_1 = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\theta_i = 0.3$ ,  $C = 0.9999$ ,  $\alpha_i = \frac{1}{i^2+1}$ , and  $\mu = 0.2$  for Algorithm 3.10. The stopping criteria is the number of iteration 250. We consider the different initialization parameter  $\tau$ . The numerical results are shown in Table 7.

We can see that  $\tau = 0.5$  greatly improves the performance of Algorithms 3.1, 3.9 and 3.10. Therefore, we choose it as the default inertial parameter for next computation.

We next calculate the numerical results by setting  $\lambda_i = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\theta_i = 0.3$ ,  $\tau = 0.5$  for Algorithms 3.1 and 3.9 and  $\alpha_i = \frac{1}{10i+1}$  for Algorithm 3.1 with  $\alpha_i = \frac{1}{10i^2+1}$  for Algorithm 3.9 and  $\lambda_1 = \frac{0.99}{\max(\text{eigenvalue}(A^T A))}$ ,  $\theta_i = 0.3$ ,  $\alpha_i = \frac{1}{i^2+1}$ ,  $\tau = 0.5$ , and  $\mu = 0.2$  for Algorithm 3.10. The stopping criteria is the number of iteration 250. We obtain the results of the different parameters  $C$  when  $V(x) = Cx$  for Algorithms 3.1 and 3.10 as seen in Table 8.

From Tables 3–8, we choose the parameters for Algorithm 3.1 to compare the exist algorithms from the literature. The following Table 9 shows choosing the necessary parameters of each algorithm.

For comparison, We set sigmoid as an activation function, hidden nodes  $M = 160$  and regularization parameter  $\lambda = 1 \times 10^{-5}$ .

Table 10 shows that Algorithm 3.1 has the highest efficiency in precision, recall, and accuracy. It also has the lowest number of iterations. It has the highest probability of correctly classifying tumors compared to algorithms examinations. We present the training and validation loss with the accuracy of training to show that Algorithm 3.1 has no overfitting in the training dataset.

**Table 7.** Training and validation loss and training time of the different parameter  $\tau$ .

|                | $\tau$           | Training Time | Loss     |            |
|----------------|------------------|---------------|----------|------------|
|                |                  |               | Training | Validation |
| Algorithm 3.1  | 0.1              | 0.4278        | 0.300531 | 0.299144   |
|                | 0.3              | 0.4509        | 0.299074 | 0.293717   |
|                | 0.5              | 0.5239        | 0.278143 | 0.272921   |
|                | $\frac{i}{2i+1}$ | 0.4708        | 0.282187 | 0.274017   |
| Algorithm 3.9  | 0.1              | 0.4592        | 0.300531 | 0.299144   |
|                | 0.3              | 0.4900        | 0.299074 | 0.293717   |
|                | 0.5              | 0.4261        | 0.278160 | 0.272922   |
|                | $\frac{i}{2i+1}$ | 0.5224        | 0.282191 | 0.274018   |
| Algorithm 3.10 | 0.1              | 1.3401        | 0.300531 | 0.299144   |
|                | 0.3              | 1.3771        | 0.299074 | 0.293717   |
|                | 0.5              | 1.8681        | 0.278143 | 0.272921   |
|                | $\frac{i}{2i+1}$ | 1.4671        | 0.282187 | 0.274017   |

**Table 8.** Training and validation loss and training time of the different parameter  $c$ .

|                | $C$    | Training Time | Loss     |            |
|----------------|--------|---------------|----------|------------|
|                |        |               | Training | Validation |
| Algorithm 3.1  | 0.3    | 0.4796        | 0.278902 | 0.273066   |
|                | 0.5    | 0.4270        | 0.278695 | 0.273024   |
|                | 0.7    | 0.4190        | 0.278480 | 0.272982   |
|                | 0.9    | 0.4209        | 0.278257 | 0.272941   |
|                | 0.9999 | 0.4844        | 0.278143 | 0.272921   |
| Algorithm 3.10 | 0.3    | 1.5886        | 0.278251 | 0.272928   |
|                | 0.5    | 1.6358        | 0.278222 | 0.272926   |
|                | 0.7    | 1.3808        | 0.278191 | 0.272924   |
|                | 0.9    | 1.5176        | 0.278159 | 0.272922   |
|                | 0.9999 | 1.4598        | 0.278143 | 0.272921   |

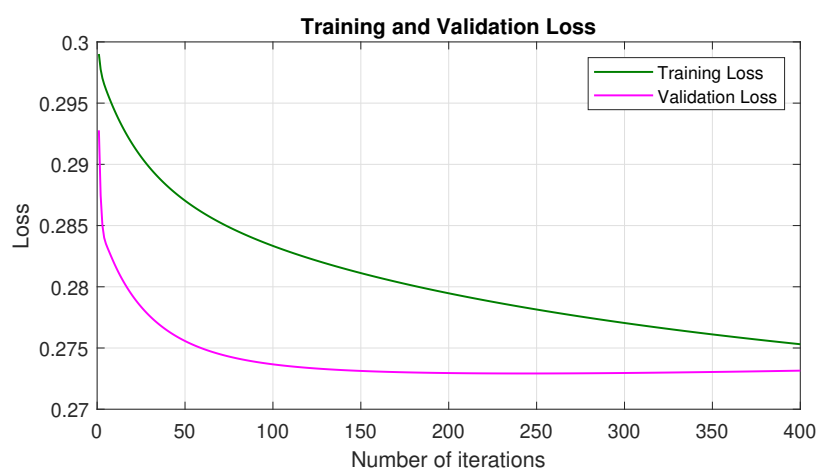
**Table 9.** Chosen parameters of each algorithm.

|             | Algorithm in (1.2)                    | Algorithm in (1.4)                    | Algorithm in (1.6)                       | Algorithm 3.1                          | Algorithm 3.9                          | Algorithm 3.10                         |
|-------------|---------------------------------------|---------------------------------------|------------------------------------------|----------------------------------------|----------------------------------------|----------------------------------------|
| $\mu$       | -                                     | 0.3                                   | 0.3                                      | -                                      | -                                      | 0.2                                    |
| $\lambda_1$ | -                                     | $\frac{0.5}{\max(\text{eig}(A^T A))}$ | $\frac{0.9999}{\max(\text{eig}(A^T A))}$ | -                                      | -                                      | $\frac{0.99}{\max(\text{eig}(A^T A))}$ |
| $\lambda_i$ | $\frac{0.5}{\max(\text{eig}(A^T A))}$ | -                                     | -                                        | $\frac{0.99}{\max(\text{eig}(A^T A))}$ | $\frac{0.99}{\max(\text{eig}(A^T A))}$ | -                                      |
| $\theta_i$  | -                                     | -                                     | 0.3                                      | 0.3                                    | 0.3                                    | 0.3                                    |
| $\alpha_i$  | $\frac{1}{100i+1}$                    | $\frac{1}{100i+1}$                    | $\frac{1}{2i+1}$                         | $\frac{1}{10i+1}$                      | $\frac{1}{10i^2+1}$                    | $\frac{1}{i^2+1}$                      |
| $\tau$      | -                                     | -                                     | 0.5                                      | 0.5                                    | 0.5                                    | 0.5                                    |
| $c$         | -                                     | -                                     | -                                        | 0.9999                                 | -                                      | 0.9999                                 |

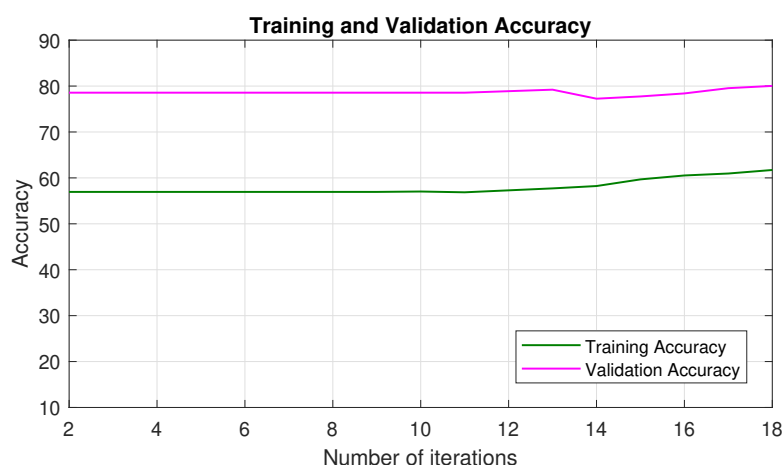
**Table 10.** The performance of each algorithm.

| Algorithm          | Iteration No. | Training Time | Pre   | Rec   | Acc (%) |
|--------------------|---------------|---------------|-------|-------|---------|
| Algorithm in (1.2) | 25            | 0.0537        | 80.97 | 97.50 | 80.03   |
| Algorithm in (1.4) | 25            | 0.3132        | 80.97 | 97.50 | 80.03   |
| Algorithm in (1.6) | 30            | 0.1182        | 80.97 | 97.50 | 80.03   |
| Algorithm 3.1      | 18            | 0.0375        | 80.97 | 97.50 | 80.03   |
| Algorithm 3.9      | 18            | 0.0401        | 80.97 | 97.50 | 80.03   |
| Algorithm 3.10     | 18            | 0.1045        | 80.97 | 97.50 | 80.03   |

From Figures 2 and 3, we see that our Algorithm 3.1 has good fit model this means that our Algorithm 3.1 suitably learns the training dataset and generalizes well to classification the PIMA Indians diabetes dataset.



**Figure 2.** Training and validation loss plots of Algorithm 3.1.



**Figure 3.** Training and validation accuracy plots of Algorithm 3.1.

## 5. Conclusions

In general, screening for diabetes in pregnancy we use The American College of Obstetricians and Gynecologists (ACOG) recommendations. The accuracy of our method is 80.03% and high accuracy may use for predict correctly diabetes in pregnancy in the future.

In this paper, we introduce a modified extragradient method with inertial extrapolation step and viscosity-type method to solve equilibrium problems of pseudomonotone bifunction operator in real Hilbert spaces. We then prove the strong convergence theorem of the proposed algorithm under the assumption that the bifunction satisfies the Lipchitz-type condition. Moreover, we show choosing stepsize parameter  $\{\lambda_i\}$  in many ways, this shows that our algorithm is flexible using, see in Algorithms 3.1 and 3.10. Finally, we show our algorithms are better performance than existing algorithms to solve the diabetes mellitus classification in machine learning.

## Acknowledgments

This research was also supported by Fundamental Fund 2022, Chiang Mai University and the NSRF via the Program Management Unit for Human Resources and Institutional Development, Research and Innovation (grant number B05F640183). W. Chalamjiak would like to thank National Research Council of Thailand (N42A650334) and Thailand Science Research and Innovation, the University of Phayao (FF65-UOE).

## Conflict of interest

The authors declare no conflict of interest.

## References

1. H. H. Bauschke, P. L. Combettes, *Convex analysis and monotone operator theory in Hilbert spaces*, Springer, New York, 2011. <https://doi.org/10.1007/978-3-319-48311-5>
2. E. Blum, W. Oettli, From optimization and variational inequalities to equilibrium problems, *Math. Student.*, **63** (1994), 123–145.
3. M. R. Bozkurt, N. Yurtay, Z. Yilmaz, C. Setkaya, Comparison of different methodologies for determining diabetes, *Turk. J. Electr. Eng. Co.*, **22** (2014), 1044–1055. <https://doi.org/10.3906/elk-1209-82>
4. S. Brahim-Belhouari, A. Bermak, Gaussian process for nonstationary time series prediction, *Comput. Stat. Data Anal.*, **47** (2014), 705–712. <https://doi.org/10.1016/j.csda.2004.02.006>
5. S. P. Chatrati, G. Hossain, A. Goyal, A. Bhan, S. Bhattacharya, D. Gaurav, et al., Smart home health monitoring system for predicting type 2 diabetes and hypertension, *J. King Saud Univ.-Com.*, **34** (2020), 862–870. <https://doi.org/10.1016/j.jksuci.2020.01.010>
6. D. K. Choubey, M. Kumar, V. Shukla, S. Tripathi, V. K. Dhandhanian, Comparative analysis of classification methods with PCA and LDA for diabetes, *Curr. Diabetes Rev.*, **16** (2020), 833–850. <https://doi.org/10.2174/1573399816666200123124008>
7. D. Deng, N. Kasabov, On-line pattern analysis by evolving self-organizing maps, *Neurocomputing*, **51** (2003), 87–103. [https://doi.org/10.1016/S0925-2312\(02\)00599-4](https://doi.org/10.1016/S0925-2312(02)00599-4)
8. D. V. Hieu, Halpern subgradient extragradient method extended to equilibrium problems, *RACSAM Rev. R. Acad. A*, **111** (2017), 823–840. <https://doi.org/10.1007/s13398-016-0328-9>
9. G. B. Huang, Q. Y. Zhu, C. K. Siew, Extreme learning machine: Theory and applications, *Neurocomputing*, **70** (2006), 489–501. <https://doi.org/10.1016/j.neucom.2005.12.126>
10. G. M. Korpelevich, The extragradient method for finding saddle points and other problems, *Matecon*, **12** (1976), 747–756. Available from: <https://cs.uwaterloo.ca/~y328yu/classics/extragrad.pdf>.
11. R. Kraikaew, S. Saejung, Strong convergence of the Halpern subgradient extragradient method for solving variational inequalities in Hilbert spaces, *J. Optim. Theory Appl.*, **163** (2014), 399–412. <https://doi.org/10.1007/s10957-013-0494-2>

12. V. A. Kumari, R. Chitra, Classification of diabetes disease using support vector machine, *Int. J. Eng. Res. Appl.*, **3** (2013), 1797–1801.
13. L. Li, Diagnosis of diabetes using a weight-adjusted voting approach, *IEEE Int. Conf. Bioinform. Bioeng.*, 2014, 320–324. <https://doi.org/10.1109/BIBE.2014.27>
14. K. Muangchoo, A new strongly convergent algorithm to solve pseudomonotone equilibrium problems in a real Hilbert space, *J. Math. Comput. Sci.*, **24** (2022), 308–322. <http://dx.doi.org/10.22436/jmcs.024.04.03>
15. L. D. Muu, W. Oettli, Convergence of an adaptive penalty scheme for finding constrained equilibria, *Nonlinear Anal.-Theor.*, **18** (1992), 1159–1166. [http://dx.doi.org/10.1016/0041-5553\(86\)90159-X](http://dx.doi.org/10.1016/0041-5553(86)90159-X)
16. B. T. Polyak, Some methods of speeding up the convergence of iteration methods, *USSR Comput. Math. Math. Phys.*, **4** (1964), 1–17. [https://doi.org/10.1016/0041-5553\(64\)90137-5](https://doi.org/10.1016/0041-5553(64)90137-5)
17. J. R. Quinlan, *C4.5: Programs for machine learning*, Elsevier, 2014.
18. R. T. Rockafellar, *Convex analysis*, Princeton University Press, 1970.
19. Y. Shehu, O. S. Iyiola, Weak convergence for variational inequalities with inertial-type method, *Appl. Anal.*, **101** (2022), 192–216. <https://doi.org/10.1080/00036811.2020.1736287>
20. S. Sahan, K. Polat, H. Kodaz, S. Gunes, *The medical applications of attribute weighted artificial immune system (AWAIS): Diagnosis of heart and diabetes diseases*, International Conference on Artificial Immune Systems, Springer, **3627** (2005), 456–468. [https://doi.org/10.1007/11536444\\_35](https://doi.org/10.1007/11536444_35)
21. R. Saxena, S. K. Sharma, M. Gupta, G. C. Sampada, A novel approach for feature selection and classification of diabetes mellitus: Machine learning methods, *Comput. Intell. Neurosci.*, **2022** (2022). <https://doi.org/10.1155/2022/3820360>
22. Y. Shehu, C. Izuchukwu, J. C. Yao, X. Qin, Strongly convergent inertial extragradient type methods for equilibrium problems, *Appl. Anal.*, 2021, 1–29. <https://doi.org/10.1080/00036811.2021.2021187>
23. World Health Organization, *Global action plan for the prevention and control of NCDs 2013–2020*, World Health Organization, 2013. Available from: [https://apps.who.int/iris/bitstream/handle/10665/94384/9789241506236%20\\_eng.pdf?sequence=1](https://apps.who.int/iris/bitstream/handle/10665/94384/9789241506236%20_eng.pdf?sequence=1).
24. J. W. Smith, J. E. Everhart, W. C. Dickson, W. C. Knowler, R. S. Johannes, Using the Adap learning algorithm to forecast the onset of diabetes mellitus, *Proc. Annu. Symp. Comput. Appl. Med. Care*, **9** (1988), 261–265. Available from: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2245318/>.
25. D. Q. Tran, M. L. Dung, V. H. Hguyen, Extragradient algorithms extended to equilibrium problems, *Optimization*, **57** (2008), 749–776. <https://doi.org/10.1080/02331930601122876>
26. R. Tibshirani, Regression shrinkage and selection via the lasso, *J. Roy. Stat. Soc. B*, **58** (1996), 267–288. <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>
27. T. Thomas, N. Pradhan, V. S. Dhaka, Comparative analysis to predict breast cancer using machine learning algorithms: A survey, *IEEE Int. Conf. Invent. Comput. Technol.*, 2020, 192–196. <https://doi.org/10.1109/ICICT48043.2020.9112464>

- 
28. H. K. Xu, Iterative algorithms for nonlinear operators, *J. London Math. Soc.*, **66** (2002), 240–256.  
<https://doi.org/10.1112/S0024610702003332>
29. M. O. Osilike, S. C. Aniagbosor, B. G. Akuchu, Fixed points of asymptotically demicontractive mappings in arbitrary Banach spaces, *Panamerican Math. J.*, **12** (2002), 77–88.



AIMS Press

© 2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)