CARPATHIAN J. MATH.
Volume **42** (2026), No. 2,
Pages 09-26

Online version at `https://www.carpathian.cunbm.utcluj.ro/`
ISSN 1584-2851 (Print edition); ISSN 1843-4401 (Electronic)
DOI: https://doi.org/10.37193/CJM.2026.02.02

A Double Inertial Fixed Point Algorithm with Linesearch and Its Application to Machine Learning for Data Classification

PUNTITA SAE-JIA[1], SUTHEP SUANTAI[2*]

ABSTRACT. In this paper, we introduce and study a new accelerated common fixed point algorithm based on the viscosity approximation, double inertial, and linesearch technique. The convergence properties and practical applications of the proposed algorithm are explored, highlighting its effectiveness in solving bilevel optimization problems and its potential in machine learning for data classification. Based on our experiment, it is found that our proposed algorithm has superior convergence behaviour than the existing algorithms in the literature.

1. INTRODUCTION

Fixed point theory is a fundamental concept in mathematics and applied sciences, offering powerful tools for analyzing and solving a wide range of problems. A point $x \in X$ is a fixed point of $T : X \to X$, where X is a nonempty set if $Tx = x$. On the other hand, optimization focuses on finding an optimal solution to a given problem under specific constraints and has applications in areas such as machine learning, supply chain management, and financial modeling. By combining fixed point theory with optimization, researchers have developed various algorithms that effectively solve complex optimization problems. In fact, many optimization problems, including convex and bilevel optimization, can be transformed into fixed point problems, making this approach particularly efficient and versatile.

The convex bilevel optimization problem consists of an outer-level minimization problem, defined as

$$\min_{x \in \Gamma} \phi(x), \tag{1.1}$$

where ϕ is a strongly convex and differentiable function from a real Hilbert space $\mathcal{H}$ to $\mathbb{R}$, and Γ represents the set of minimizers of the inner-level problem:

$$\arg\min_{x \in \mathcal{H}} \{f(x) + g(x)\}, \tag{1.2}$$

where $f : \mathcal{H} \to \mathbb{R}$ is a convex and differentiable function, and $g \in \Gamma_0(\mathcal{H})$, the set of proper, lower semicontinuous, and convex functions from $\mathcal{H}$ to $\mathbb{R}$.

It is well-known that, if $x^* \in \Gamma$ is a solution of (1.2) then

$$0 \in \nabla f(x^*) + \partial g(x^*). \tag{1.3}$$

Moreover, we know that

$$x^* \text{ is a solution of (1.3)} \iff x^* = \text{prox}_{ag}(x^* - a\nabla f(x^*)),$$

Received: 11.03.2025. In revised form: 04.08.2025. Accepted: 11.09.2025
2010 *Mathematics Subject Classification.* 47H10; 65K10; 90C25.
Key words and phrases. *accelerated algorithm; double inertial algorithm; linesearch; convex bilevel optimization problem; data classification; noncommunicable diseases.*
Corresponding author: Suthep Suantai; suthep.s@cmu.ac.th

where $a > 0$ and $\text{prox}_{ag}(y) = \underset{x\in H}{\arg\min}\{g(x) + \frac{1}{2a}\|x - y\|_2^2\}$. It is known that proximal gradient mapping $T := \text{prox}_{\alpha g}(I - \alpha\nabla f)$ is nonexpansive if $a \in (0, \frac{2}{L_f})$ where L_f is a Lipschitz constant of ∇f, and its fixed point set is $\underset{x\in H}{\arg\min}\{f(x) + g(x)\}$. From above fact, we see that the set of all common fixed points of $T_n := \text{prox}_{\alpha_n g}(I - \alpha_n\nabla f)$ is Γ, the set of minimizers of Problem (1.2).

Furthermore, $x \in \Gamma$ is a solution for problem (1.1) if the following variational inequality holds:

$$\langle\nabla\phi(x), y - x\rangle \geq 0, \ \forall y \in \Gamma.$$

This means that solving the bilevel problem (1.1) is equivalent to finding a fixed point of T_n. Hence, fixed point theory is one of the the most effective tools for this kind of problem, see [1–3].

To approximate the optimal solution for Problems (1.1), Sabach and Shtern [4] constructed an algorithm, called BiG-SAM (*Bilevel Gradient Sequential Averaging Method*) which was defined as follows:

Algorithm 1 BiG-SAM

Input: $x_1 \in \mathbb{R}^m, \gamma_n \in (0,1), \alpha_n \in (0, \frac{1}{L_f})$ and $s \in (0, \frac{2}{L_\phi+\sigma})$ where L_f and L_ϕ are the Lipschitz constants of ∇f and $\nabla\phi$, respectively.
Compute:

$$\begin{cases} y_n &= \text{prox}_{\alpha_n g}(x_n - \alpha_n\nabla f(x_n)), \\ x_{n+1} &= \gamma_n(x_n - s\nabla\phi(x_n)) + (1-\gamma_n)y_n, \ n \geq 1. \end{cases}$$

They showed that $x_n \to x \in \Omega$, where ω is the set of all solutions of Problems (1.1).

Later, Shehu et al. [5] introduced the algorithm iBiG-SAM (*inertial Bilevel Gradient Sequential Averaging Method*) for accelerating the convergence rate of Algorithm 1 by using inertial technique that was proposed by Polyak [6].

Algorithm 2 iBiG-SAM

Input: $x_0, x_1 \in \mathbb{R}^m, \alpha \geq 3, \gamma_n \in (0,1), \alpha_n \in (0, \frac{2}{L_f}), s \in (0, \frac{2}{L_\phi+\sigma}]$ where L_f and L_ϕ are the Lipschitz constants of ∇f and $\nabla\phi$, respectively.
For $n \geq 1$ **:**
Choose: $\theta_n \in [0, \bar{\theta_n}]$ where $\bar{\theta_n}$ is defined by

$$\bar{\theta_n} := \begin{cases} \min\{\frac{n-1}{n+\alpha-1}, \frac{\tau_n}{\|x_n - x_{n-1}\|}\}, & \text{if } x_n \neq x_{n-1}, \\ \frac{n-1}{n+\alpha-1}, & \text{otherwise.} \end{cases}$$

Compute:

$$\begin{cases} y_n &= x_n + \theta_n(x_n - x_{n-1}), \\ t_n &= \text{prox}_{\alpha_n g}(y_n - \alpha_n\nabla f(y_n)), \\ w_n &= y_n - s\nabla\phi(y_n), \\ x_{n+1} &= \gamma_n w_n + (1-\gamma_n)t_n. \end{cases}$$

Under conditions that

$$\lim_{n\to\infty}\gamma_n = 0 \text{ and } \sum_{n=1}^{\infty}\gamma_n = \infty, \tag{1.4}$$

They also proved that $x_n \to x \in \Omega$.

After that, in order to accelerate the convergence of Algorithm 2, Duan and Zhang [7] introduced the following three algorithms, namely Algorithm 3, Algorithm 4 and Algorithm 5.

Algorithm 3 aiBiG-SAM: The alternated inertial Bilevel Gradient Sequential Averaging Method

Input: $x_0, x_1 \in \mathbb{R}^m, \alpha \geq 3, \xi > 0, \alpha_n \in (0, \frac{2}{L_f}), s \in (0, \frac{2}{L_\phi + \sigma}]$ where L_f and L_ϕ are the Lipschitz constants of ∇f and $\nabla \phi$, respectively. Let $\{\gamma_n\}$ be a sequence in $(0,1)$ such that satisfies (1.4).

For $n \geq 1$ **:**

Step 1. Compute:

$$y_n = \begin{cases} x_n + \theta_n(x_n - x_{n-1}), & \text{if } n \text{ is odd,} \\ x_n, & \text{otherwise.} \end{cases}$$

When n is odd, choose θ_n such that $0 \leq |\theta_n| \leq \bar{\theta_n}$ with $\bar{\theta_n}$ defined by

$$\bar{\theta_n} := \begin{cases} \min\{\frac{n}{n+\alpha-1}, \frac{\tau_n}{\|x_n - x_{n-1}\|}\}, & \text{if } x_n \neq x_{n-1}, \\ \frac{n}{n+\alpha-1}, & \text{otherwise.} \end{cases}$$

When n is even, $\theta_n = 0$.

Step 2. Compute:

$$\begin{cases} t_n & = \text{prox}_{\alpha_n g}(y_n - \alpha_n \nabla f(y_n)), \\ w_n & = y_n - s\nabla \phi(y_n), \\ x_{n+1} & = \gamma_n w_n + (1-\gamma_n)t_n. \end{cases}$$

Step 3. If $\|x_n - x_{n-1}\| < \xi$, then stop. Otherwise, set $n = n+1$ and go to Step 1.

Algorithm 4 miBiG-SAM: The multi-step inertial Bilevel Gradient Sequential Averaging Method

Input: $x_0, x_1 \in \mathbb{R}^m, \alpha \geq 3, \xi > 0, \alpha_n \in (0, \frac{2}{L_f}), s \in (0, \frac{2}{L_\phi + \sigma}]$ where L_f and L_ϕ are the Lipschitz constants of ∇f and $\nabla \phi$, respectively. Let $\{\gamma_n\}$ be a sequence in $(0,1)$ such that satisfies (1.4).

For $n \geq 1$ **:**

Step 1. Given $x_n, x_{n-1}, \ldots, x_{n-q+1}$ and compute

$$y_n = x_n + \sum_{i \in Q} \theta_{i,n}(x_{n-i} - x_{n-1-i}),$$

where $Q = \{0, 1, \ldots, q-1\}$. Choose $\theta_{i,n}$ such that $0 \leq |\theta_{i,n}| \leq \bar{\theta}_n$ with $\bar{\theta}_n$ defined by

$$\bar{\theta}_n := \begin{cases} \min\{\frac{n}{n+\alpha-1}, \frac{\tau_n}{\sum_{i\in Q}\|x_{n-i} - x_{n-1-i}\|}\}, & \text{if } \sum_{i \in Q}\|x_{n-i} - x_{n-1-i}\| \neq 0, \\ \frac{n}{n+\alpha-1}, & \text{otherwise.} \end{cases}$$

Step 2. Compute:

$$\begin{cases} t_n & = \text{prox}_{\alpha_n g}(y_n - \alpha_n \nabla f(y_n)), \\ w_n & = y_n - s\nabla \phi(y_n), \\ x_{n+1} & = \gamma_n w_n + (1-\gamma_n)t_n. \end{cases}$$

Step 3. If $\|x_n - x_{n-1}\| < \xi$, then stop. Otherwise, set $n = n+1$ and go to Step 1.

Algorithm 5 amiBiG-SAM: The multi-step alternative inertial Bilevel Gradient Sequential Averaging Method

Input: $x_0, x_1 \in \mathbb{R}^m, \alpha \geq 3, \xi > 0, \alpha_n \in (0, \frac{2}{L_f}), s \in (0, \frac{2}{L_\phi + \sigma}]$ where L_f and L_ϕ are the Lipschitz constants of ∇f and $\nabla \phi$, respectively. Let $\{\gamma_n\}$ be a sequence in $(0,1)$ such that satisfies (1.4).
For $n \geq 1$ **:**
Step 1. Given $x_n, x_{n-1}, \ldots, x_{n-q+1}$ and compute

$$y_n = \begin{cases} x_n + \sum_{i \in Q} \theta_{i,n}(x_{n-i} - x_{n-1-i}), & \text{if } n \text{ is odd,} \\ x_n, & \text{otherwise,} \end{cases}$$

where $Q = \{0, 1, \ldots, q-1\}$. When n is odd, choose $\theta_{i,n}$ such that $0 \leq |\theta_{i,n}| \leq \bar{\theta}_n$ with $\bar{\theta}_n$ defined by

$$\bar{\theta}_n := \begin{cases} \min\{\frac{n}{n+\alpha-1}, \frac{\tau_n}{\sum_{i \in Q} \|x_{n-i} - x_{n-1-i}\|}\}, & \text{if } \sum_{i \in Q} \|x_{n-i} - x_{n-1-i}\| \neq 0, \\ \frac{n}{n+\alpha-1}, & \text{otherwise.} \end{cases}$$

When n is even, $\theta_n = 0$.
Step 2. Compute:

$$\begin{cases} t_n &= \text{prox}_{\alpha_n g}(y_n - \alpha_n \nabla f(y_n)), \\ w_n &= y_n - s\nabla\phi(y_n), \\ x_{n+1} &= \gamma_n w_n + (1-\gamma_n) t_n. \end{cases}$$

Step 3. If $\|x_n - x_{n-1}\| < \xi$, then stop. Otherwise, set $n = n+1$ and go to Step 1.

In comparison to Algorithm 1 and Algorithm 2, they demonstrated that the convergence behavior of Algorithm 3, Algorithm 4, and Algorithm 5 was better. We observed that Algorithm 1 - Algorithm 5 were introduced by using a fixed point technique. After that, viscosity approximation and some fixed point methods together with the inertial technique were employed to construct accelerated algorithm for solving convex bilevel optimization, see [8–11].

It is seen in Algorithm 1 - Algorithm 5 that the Lipschitz continuity condition on ∇f must be assumed and that the Lipschitz constant L_f determines the stepsize α_n. The Lipschitz constant L_f can be challenging to find, though. Therefore, we shall develop an algorithm whose stepsize is independent from the Lipschitz constants ∇f. In order to, overcome this difficulty, Cruz et al. [12] substituted weak assumptions for the Lipschitz continuity condition on ∇f, as described below:

Assumption 1.1.
(A1) $f : \mathcal{H} \to \mathbb{R}$ *is a convex and differentiable function and the gradient* ∇f *is uniformly continuous on* $\mathcal{H}$*;*
(A2) $g : \mathcal{H} \to \mathbb{R}$ *is a proper lower semicontinuous and convex function.*

We see that the Assumption 1.1 (A1) is a weaker condition than the Lipschitz continuity condition on ∇f. Moreover, they proposed an linesearch method for choosing the stepsize α_n that is not depend on any Lipschitz constants.

Linesearch 1.1. Given $x, \sigma > 0, \theta \in (0,1)$ and $\delta \in (0, \frac{1}{2})$.

Input. Set $\alpha = \sigma$ and $J(x, \alpha) := \text{prox}_{\alpha g}(x - \alpha \nabla f(x))$ with $x \in \text{dom}\, g$.
While $\alpha \|\nabla f(J(x,\alpha)) - \nabla f(x)\| > \delta \|J(x,\alpha) - x\|$ **do**
$\alpha = \theta\alpha$
End While
Output α

In 2021, Suantai et al. [13] introduced a new linesearch for choosing the stepsize α_n as follows:

Linesearch 1.2. Given $x \in \mathcal{H}, \sigma > 0, \delta > 0$, and $\theta \in (0,1)$.

Input. Set $\alpha = \sigma$.
While $\frac{\alpha}{2}\|\nabla f(J^2(x,\alpha)) - \nabla f(J(x,\alpha))\| + \|\nabla f(J(x,\alpha)) - \nabla f(x)\| > \delta(\|J^2(x,\alpha) - J(x,\alpha)\| + \|J(x,\alpha) - x\|)$ **do**
$\alpha = \theta\alpha$
End While
Output α

Using Linesearch 1.2, they introduced a new viscosity forward-backward algorithm with inertial technique as follows:

Algorithm 6 An accelerated viscosity forward-backward algorithm with Linesearch 1.2

Initialization: Choose $x_1,\ x_0 \in \mathcal{H}, \sigma > 0, \delta \in (0, \frac{1}{8})$ and $\theta \in (0,1)$. Take $\{\gamma_n\}, \{\tau_n\} \subset (0,\infty)$, and $\{\mu_n\} \subset (0,\infty)$.
Iterative steps: For $n \geq 1$, calculate x_{n+1} as follows:
Step 1. Compute the inertial step:

(1.5)
$$\theta_n = \begin{cases} \min\{\mu_n, \frac{\tau_n}{\|x_n - x_{n-1}\|}\}, & \text{if } x_n \neq x_{n-1}; \\ \mu_n, & \text{otherwise.}, \end{cases}$$

(1.6)
$$w_n = x_n + \theta_n(x_n - x_{n-1}).$$

Step 2. Compute the forward-backward step:

(1.7)
$$z_n = \text{prox}_{\alpha_n g}(w_n - \alpha_n \nabla f(w_n))$$

(1.8)
$$y_n = \text{prox}_{\alpha_n g}(z_n - \alpha_n \nabla f(z_n))$$

where α_n = Linesearch 1.2$(w_n, \sigma, \theta, \delta)$.
Step 3. Compute the viscosity step:

(1.9)
$$x_{n+1} = \gamma_n F(x_n) + (1 - \gamma_n) y_n.$$

Set n := n+1 and return to Step 1.

They showed that the sequence $\{x_n\}$ generated by Algorithm 6 converges strongly to $x^* \in \Omega$, the solution set of problems (1.1).

This paper focuses on the combination of fixed point and optimization theory by proposing a new accelerated fixed point algorithm incorporating multiple techniques, including the viscosity approximation method, inertial, and linesearch technique.

Inspired and motivated by the results of Cruz and Nghia [12], Suantai et al. [13] and the above-mentioned research, we aim to introduce a new accelerated algorithm using the basic Linesearch 1.1 together with the viscosity approximation method for solving the convex bilevel optimization problem (1.1) and apply the obtained result to solving some data classification problems. Furthermore, comparison of the performance of our proposed algorithm with the other algorithms are also given.

The structure of this paper is as follows: Section 2 provides fundamental definitions and essential lemmas, while Section 3 presents the main theoretical results. Section 4 discusses the implementation of our proposed method for data classification problems, particularly using diabetes, breast cancer and Hypertension datasets, and includes comparison results with other algorithms. Finally, the conclusion of our work is given in 5.

2. PRELIMINARIES AND LEMMAS

Throughout this work, let $\mathcal{H}$ be a real Hilbert space and $T : \mathcal{H} \to \mathcal{H}$ a mapping. We use the notation $x_n \to x$ to indicate the strong convergence of the sequence $\{x_n\}$ to $x \in \mathcal{H}$, and $x_n \rightharpoonup x$ to denote weak convergence.

Definition 2.1. *A mapping $T : \mathcal{H} \to \mathcal{H}$ is said to be*

(1) Lipschitzian *if there exists $L \geq 0$ such that*

$$\|Tx - Ty\| \leq L\|x - y\|, \ \forall x, y \in \mathcal{H},$$

(2) k-contraction *if T is Lipscitzian with constant $k \in [0, 1)$*

(3) nonexpansive *if T is Lipscitzian with constant $L = 1$*

Definition 2.2. *Let $x \in \mathcal{H}$ and C be a nonempty closed and convex subset of $\mathcal{H}$. Then there is a unique point $x^* \in C$ such that*

$$\|x^* - x\| \leq \|y - x\|, \ \ \forall y \in C.$$

Let $P_C : \mathcal{H} \to C$ be defined by $P_C x = x^*$. The mapping P_C is known as a metric projection *of $\mathcal{H}$ on C.*

We know that P_C is nonexpansive and satisfies the following inequality

$$\langle x - P_C x, y - P_C x\rangle \leq 0 \ \forall x \in \mathcal{H} \text{ and } y \in C \tag{2.10}$$

Definition 2.3. *Let $\Gamma_0(\mathcal{H})$ be the set of all proper lower semicontinuous and convex functions $f : \mathcal{H} \to (-\infty, +\infty]$. For $g \in \Gamma_0(\mathcal{H})$, the subdifferential ∂g of g is defined by*

$$\partial g(x) := \{u \in \mathcal{H} : g(x) + \langle u, y - x\rangle \leq g(y), \ \forall y \in \mathcal{H}\}, \ \forall x \in \mathcal{H}.$$

At this point, we provide a few of the interconnections between the subdifferential operator and the proximity operator. For $x \in \mathcal{H}$ and $\alpha > 0$, we have

$$\frac{x - \operatorname{prox}_{\alpha g}(x)}{\alpha} \in \partial g(\operatorname{prox}_{\alpha g}(x)) \tag{2.11}$$

where $\operatorname{prox}_{\alpha g} = (I + \alpha\partial g)^{-1}$.

We complete this section by providing helpful lemmas and propositions that support our main results.

Lemma 2.1 ([14]). *The following holds with $x, y \in \mathcal{H}$ and $\lambda \in [0, 1]$:*

(1) $\|x \pm y\|^2 = \|x\|^2 \pm 2\langle x, y\rangle + \|y\|^2$;

(2) $\|x + y\|^2 \leq \|x\|^2 + 2\langle y, x + y\rangle$;

(3) $\|\lambda x + (1 - \lambda)y\|^2 = \lambda\|x\|^2 + (1 - \lambda)\|y\|^2 - \lambda(1 - \lambda)\|x - y\|^2$.

Lemma 2.2 ([15]). *Let $h \in \Gamma_0(\mathcal{H})$. Let $\{x_n\}$ and $\{y_n\}$ be two sequences in $\mathcal{H}$ such that $y_n \in \partial h(x_n)$ for all $n \in \mathbb{N}$. If $x_n \rightharpoonup x$ and $y_n \to y$, then $y \in \partial h(x)$.*

Lemma 2.3 ([16]). *Let $\{a_n\} \subset \mathbb{R}_+, \{b_n\} \subset \mathbb{R}$, and $\{\xi_n\} \subset (0, 1)$ be such that $\sum_{n=1}^{\infty} \xi_n = \infty$ and*

$$a_{n+1} \leq (1 - \xi_n)a_n + \xi_n b_n, \ \forall n \in \mathbb{N}.$$

If $\limsup_{i\to\infty} b_{n_i} \leq 0$ for every subsequence $\{a_{n_i}\}$ of $\{a_n\}$ satisfying $\liminf_{i\to\infty}(a_{n_i+1} - a_{n_i}) \geq 0$, then $\lim_{n\to\infty} a_n = 0$.

3. Main results

In this section, we introduce a new accelerated common fixed point algorithm that solves convex bilevel optimization problems without any Lipschitz continuity condition on ∇f. It is modified by developing Algorithm 6 with double inertial technique and Linesearch 1.1. We then provide a strong convergence result of our proposed algorithm under specific conditions. Using Assumption 1.1, we now focus on Problems (1.1) and (1.2). For clarity, let $h := f + g$. The set of minimizers of the Problem (1.2) is represented by Γ, and we assume that Γ is nonemty.

Throughout this section, let $F : \mathcal{H} \to \mathcal{H}$ be a k-contraction mapping with $k \in [0, 1)$ and let $\{\gamma_n\}, \{\tau_n\} \subset (0, \infty), \{\mu_n\} \subset (0, \infty)$ and $\{\rho_n\} \subset (-\infty, 0)$. We start by developing the algorithm as follows:

Algorithm 7 An double inertial viscosity forward-backward algorithm with linesearch (DIVFBAL)

Initialization: Choose $x_1,\ x_0,\ x_{-1} \in \mathcal{H}, \sigma > 0, \delta \in (0, \frac{1}{8})$ and $\theta \in (0, 1)$. Take $\{\gamma_n\}, \{\tau_n\} \subset (0, \infty), \{\mu_n\} \subset (0, \infty)$ and $\{\rho_n\} \subset (-\infty, 0)$.

Iterative steps: For $n \geq 1$, calculate x_{n+1} as follows:

Step 1. Compute the inertial parameters θ_n and δ_n by

$$\theta_n = \begin{cases} \min\{\mu_n, \frac{\tau_n}{\|x_n - x_{n-1}\|}\}, & \text{if } x_n \neq x_{n-1}; \\ \mu_n, & \text{otherwise}, \end{cases} \tag{3.12}$$

and

$$\delta_n = \begin{cases} \max\{\rho_n, \frac{-\tau_n}{\|x_{n-1} - x_{n-2}\|}\}, & \text{if } x_n \neq x_{n-1}; \\ \rho_n, & \text{otherwise}, \end{cases} \tag{3.13}$$

and let

$$w_n = x_n + \theta_n(x_n - x_{n-1}) + \delta_n(x_{n-1} - x_{n-2}). \tag{3.14}$$

Step 2. Compute the forward-backward step:

$$z_n = \text{prox}_{\alpha_n g}(w_n - \alpha_n \nabla f(w_n)) \tag{3.15}$$

$$y_n = \text{prox}_{\beta_n g}(z_n - \beta_n \nabla f(z_n)) \tag{3.16}$$

where $\alpha_n = Linesearch1.1(w_n, \sigma, \theta, \delta)$ and $\beta_n := Linesearch1.1(z_n, \sigma, \theta, \delta)$.

Step 3. Compute the viscosity step:

$$x_{n+1} = \gamma_n F(x_n) + (1 - \gamma_n) y_n. \tag{3.17}$$

Set n := n+1 and return to Step 1.

The following tool is required in order to demonstrate a strong convergence result of Algorithm 7.

Lemma 3.4. *Let $\{x_n\}$ be a sequence generated by Algorithm 7 and $x^* \in \mathcal{H}$. Then the following inequality holds:*

$$\|w_n - x^*\|^2 - \|z_n - x^*\|^2 \geq 2\alpha_n[h(z_n) - h(x^*)] + (1 - 2\delta)\|z_n - w_n\|^2 \tag{3.18}$$

$$\|z_n - x^*\|^2 - \|y_n - x^*\|^2 \geq 2\beta_n[h(y_n) - h(x^*)] + (1 - 2\delta)\|y_n - z_n\|^2 \tag{3.19}$$

for all $n \in \mathbb{N}$.

Proof. Let $x^* \in \mathcal{H}$. To prove (3.18), we obtain from (2.11) that

$$\frac{w_n - z_n}{\alpha_n} - \nabla f(w_n) \in \partial g(z_n) \text{ for all } n \in \mathbb{N}.$$

Based on the definition of $\partial g(z_n)$, the equation given above yields

$$\begin{array}{rcl} g(x^*) - g(z_n) & \geq & \langle \frac{w_n - z_n}{\alpha_n} - \nabla f(w_n), x^* - z_n \rangle \\ & = & \frac{1}{\alpha_n} \langle w_n - z_n, x^* - z_n \rangle + \langle \nabla f(w_n), z_n - x^* \rangle. \end{array} \tag{3.20}$$

By Assumption 1.1 (AI), we obtain the fact that

$$f(x) - f(y) \geq \langle \nabla f(y), x - y \rangle \ \forall x, y \in \mathcal{H}. \tag{3.21}$$

From (3.21), we get

$$f(x^*) - f(w_n) \geq \langle \nabla f(w_n), x^* - w_n \rangle, \tag{3.22}$$

and

$$f(w_n) - f(z_n) \geq \langle \nabla f(z_n), w_n - z_n \rangle. \tag{3.23}$$

From (3.20) and (3.22), we obtain that

$$\begin{array}{rl} h(x^*) - f(w_n) - g(z_n) & \\ \geq & \frac{1}{\alpha_n} \langle w_n - z_n, x^* - z_n \rangle + \langle \nabla f(w_n), z_n - w_n \rangle \\ = & \frac{1}{\alpha_n} \langle w_n - z_n, x^* - z_n \rangle + \langle \nabla f(w_n) - \nabla f(z_n), z_n - w_n \rangle \\ & + \langle \nabla f(z_n), z_n - w_n \rangle \\ \geq & \frac{1}{\alpha_n} \langle w_n - z_n, x^* - z_n \rangle - \|\nabla f(w_n) - \nabla f(z_n)\| \cdot \|z_n - w_n\| \\ & + \langle \nabla f(z_n), z_n - w_n \rangle. \end{array}$$

From $\alpha := Linesearch 1.1(w_n, \sigma, \theta, \delta)$ and (3.23), the above inequality become

$$\frac{1}{\alpha_n} \langle w_n - z_n, z_n - x^* \rangle \geq h(z_n) - h(x^*) - \frac{\delta}{\alpha_n} \|z_n - w_n\|^2 \ \forall n \in \mathbb{N}. \tag{3.24}$$

By Lemma 2.1 (1), we get

$$\|w_n - x^*\|^2 - \|z_n - x^*\|^2 \geq 2\alpha_n [h(z_n) - h(x^*)] + (1 - \delta) \|z_n - w_n\|^2 \ \forall n \in \mathbb{N}.$$

Similarly, we have

$$\frac{z_n - y_n}{\beta_n} - \nabla f(z_n) \in \partial g(y_n),$$

$$g(x^*) - g(y_n) \geq \langle \frac{z_n - y_n}{\beta_n} - \nabla f(z_n), x^* - y_n \rangle,$$

$$f(x^*) - f(z_n) \geq \langle \nabla f(z_n), x^* - z_n \rangle,$$

and

$$f(z_n) - f(y_n) \geq \langle \nabla f(y_n), z_n - y_n \rangle.$$

From the above inequalities, we obtain that

$$\begin{array}{rcl} h(x^*) - f(z_n) - g(y_n) & \geq & \frac{1}{\beta_n} \langle z_n - y_n, x^* - y_n \rangle + \langle \nabla f(z_n), y_n - z - n \rangle \\ & = & \frac{1}{\beta_n} \langle z_n - y_n, x^* - y_n \rangle \\ & & + \langle \nabla f(z_n) - \nabla f(y_n), y_n - z_n \rangle \\ & & + \langle \nabla f(y_n), y_n - z - n \rangle \\ & \geq & \frac{1}{\beta_n} \langle z_n - y_n, x^* - y_n \rangle \\ & & - \|\nabla f(z_n) - \nabla f(y_n)\| \cdot \|y_n - z_n\| \\ & & + \langle \nabla f(y_n), y_n - z - n \rangle \\ & \geq & \frac{1}{\beta_n} \langle z_n - y_n, x^* - y_n \rangle \\ & & - \frac{\delta}{\beta_n} \|y_n - z_n\|^2 + f(y_n) - f(z_n). \end{array}$$

It implies that

$$\frac{1}{\beta_n}\langle z_n - y_n, y_n - x^*\rangle \geq h(y_n) - h(x^*) - \frac{\delta}{\beta_n}\|y_n - z_n\|^2 \ \forall n \in \mathbb{N}.$$

Again, by Lemma 2.1 (1), we obtain that

$$\|z_n - x^*\|^2 - \|y_n - x^*\|^2 \geq 2\beta_n(h(y_n) - h(x^*)) + (1-2\delta)\|y_n - z_n\|^2 \ \forall n \in \mathbb{N}.$$

□

Theorem 3.2. *Let $\{x_n\} \subset \mathcal{H}$ be a sequence generated by Algorithm 7. Then:*

1). For $x^* \in \Gamma$, we have

$$\|x_{n+1} - x^*\| \leq \max\{\|x_n - x^*\|, \frac{\frac{\theta_n}{\gamma_n}\|x_n - x_{n-1}\| + \frac{|\delta_n|}{\gamma_n}\|x_{n-1} - x_{n-2}\| + \|F(x^*) - x^*\|}{1-k}\}$$

2). If the sequence $\{\alpha_n\}, \{\gamma_n\}$, and $\{\tau_n\}$ satisfy the following conditions:

2.1). $\alpha_n \geq a$ for some $a \in \mathbb{R}_{++}$;

2.2). $\gamma_n \in (0,1)$ such that $\lim\limits_{n\to\infty} \gamma_n = 0$ and $\sum\limits_{n=1}^{\infty} \gamma_n = \infty$;

2.3). $\lim\limits_{n\to\infty} \frac{\tau_n}{\gamma_n} = 0$,

then $x_n \to x^* \in \Gamma$, where $x^* = P_\Gamma f(x^*)$.

Proof. Let $x^* \in \Gamma$ such that $x^* = P_\Gamma f(x^*)$. By Lemma 3.4, we have

$$\|w_n - x^*\|^2 - \|z_n - x^*\|^2 \geq (1-2\delta)\|z_n - w_n\|^2, \tag{3.25}$$

and

$$\|z_n - x^*\|^2 - \|y_n - x^*\|^2 \geq (1-2\delta)\|y_n - z_n\|^2. \tag{3.26}$$

From (3.25) and (3.26), we get

$$\begin{aligned} \|y_n - x^*\| &\leq \|z_n - x^*\| \\ &\leq \|w_n - x^*\| \\ &\leq \|x_n - x^*\| + \theta_n\|x_n - x_{n-1}\| + |\delta_n| \cdot \|x_{n-1} - x_{n-2}\|, \end{aligned} \tag{3.27}$$

and

$$\begin{aligned} \|y_n - x^*\|^2 &\leq \|z_n - x^*\|^2 - (1-2\delta)\|y_n - z_n\|^2 \\ &\leq \|w_n - x^*\|^2 - (1-2\delta)[\|z_n - w_n\|^2 + \|y_n - z_n\|^2] \\ &= \|x_n - x^*\|^2 + 2\theta_n\langle x_n - x^*, x_n - x_{n-1}\rangle \\ &\quad +2\delta_n\langle x_n - x^*, x_{n-1} - x_{n-2}\rangle \\ &\quad +\|\theta_n(x_n - x_{n-1}) + \delta_n(x_{n-1} - x_{n-2})^2 \\ &\quad -(1-2\delta)[\|z_n - w_n\|^2 + \|y_n - z_n\|^2] \\ &= \|x_n - x^*\|^2 + 2\theta_n\langle x_n - x^*, x_n - x_{n-1}\rangle \\ &\quad +2\delta_n\langle x_n - x^*, x_{n-1} - x_{n-2}\rangle \\ &\quad +2\theta_n\delta_n\langle x_n - x_{n-1}, x_{n-1} - x_{n-2}\rangle \\ &\quad +\theta_n^2\|x_n - x_{n-1}\|^2 + \delta_n^2\|x_{n-1} - x_{n-2}\|^2 \\ &\quad -(1-2\delta)[\|z_n - w_n\|^2 + \|y_n - z_n\|^2] \\ &\leq \|x_n - x^*\|^2 + 2\theta_n\|x_n - x^*\| \cdot \|x_n - x_{n-1}\| \\ &\quad +2\,|\delta_n| \cdot \|x_n - x^*\| \cdot \|x_{n-1} - x_{n-2}\| \\ &\quad +2\theta_n\,|\delta_n| \cdot \|x_n - x_{n-1}\| \cdot \|x_{n-1} - x_{n-2}\| \\ &\quad +\theta_n^2\|x_n - x_{n-1}\|^2 + \delta_n^2\|x_{n-1} - x_{n-2}\|^2 \\ &\quad -(1-2\delta)[\|z_n - w_n\|^2 + \|y_n - z_n\|^2]. \end{aligned} \tag{3.28}$$

By (3.17) and (3.27), we have

$$\begin{aligned}
\|x_{n+1} - x*\| &\leq \gamma_n\|F(x_n) - F(x^*)\| + \gamma_n\|F(x*) - x^*\| \\
&\quad +(1-\gamma_n)\|y_n - x^*\| \\
&\leq \gamma_n\|x_n - x^*\| + \gamma_n\|F(x*) - x^*\| \\
&\quad +(1-\gamma_n)\|y_n - x^*\| \\
&= (1-\gamma_n(1-k))\|x_n - x^*\| + \gamma_n\|F(x^*) - x^*\| \\
&\quad +(1-\gamma_n)[\theta_n\|x_n - x_{n-1}\| + |\delta_n|\cdot\|x_{n-1} - x_{n-2}\|] \\
&\leq (1-\gamma_n(1-k))\|x_n - x^*\| + \gamma_n\|F(x^*) - x^*\| \\
&\quad +\theta_n\|x_n - x_{n-1}\| + |\delta_n|\cdot\|x_{n-1} - x_{n-2}\| \\
&= (1-\gamma_n(1-k))\|x_n - x^*\| \\
&\quad +\gamma_n(1-k)[\tfrac{\frac{\theta_n}{\gamma_n}\|x_n - x_{n-1}\| + \frac{|\delta_n|}{\gamma_n}\cdot\|x_{n-1} - x_{n-2}\| + \|F(x^*) - x^*\|}{1-k}] \\
&\leq \max\{\|x_n - x^*\|, \tfrac{\frac{\theta_n}{\gamma_n}\|x_n - x_{n-1}\| + \frac{|\delta_n|}{\gamma_n}\cdot\|x_{n-1} - x_{n-2}\| + \|F(x^*) - x^*\|}{1-k}\}.
\end{aligned}$$

Hence, we obtain 1). From (3.12), (3.13) and conditions 2.3), we have

$$\frac{\theta_n}{\gamma_n}\|x_n - x_{n-1}\| \to 0 \text{ as } n \to \infty,$$

and

$$\frac{|\delta_n|}{\gamma_n}\|x_{n-1} - x_{n-2}\| \to 0 \text{ as } n \to \infty.$$

So there exists positive constants M_1 and M_2 such that

$$\frac{\theta_n}{\gamma_n}\|x_n - x_{n-1}\| \leq M_1,$$

and

$$\frac{|\delta_n|}{\gamma_n}\|x_{n-1} - x_{n-2}\| \leq M_2 \ \forall n \in \mathbb{N}.$$

Thus, for all $n \in \mathbb{N}$

$$\begin{aligned}
\|x_{n+1} - x*\| &\leq \max\{\|x_n - x^*\|, \tfrac{M_1 + M_2 + \|F(x^*) - x^*\|}{1-k}\} \\
&\ \ \vdots \\
&\leq \max\{\|x_1 - x^*\|, \tfrac{M_1 + M_2 + \|F(x^*) - x^*\|}{1-k}\}.
\end{aligned}$$

Therefore, $\{x_n\}$ is bounded.

By Lemma 2.1 (2), (3) and (3.28), we obtain that

$$
\begin{aligned}
\|x_{n+1} - x*\| &\leq \|(1-\gamma_n)(y_n - x^*) + \gamma_n(F(x_n) - F(x^*))\|^2 \\
&\quad +2\gamma_n\langle F(x^*) - x^*, x_{n+1} - x^*\rangle \\
&\leq (1-\gamma_n)\|y_n - x^*\|^2 + \gamma_n\|F(x_n) - F(x^*)\|^2 \\
&\quad +2\gamma_n\langle F(x^*) - x^*, x_{n+1} - x^*\rangle \\
&\leq (1-\gamma_n)\|y_n - x^*\|^2 + \gamma_n k^2\|x_n - x^*\|^2 \\
&\quad +2\gamma_n\langle F(x^*) - x^*, x_{n+1} - x^*\rangle \\
&\leq (1-\gamma_n)\Big[\|x_n - x^*\|^2 + 2\theta_n\|x_n - x^*\| \cdot \|x_n - x_{n-1}\| \\
&\quad +2\,|\delta_n| \cdot \|x_n - x^*\| \cdot \|x_{n-1} - x_{n-2}\| \\
&\quad +2\theta_n\,|\delta_n| \cdot \|x_n - x_{n-1}\| \cdot \|x_{n-1} - x_{n-2}\| \\
&\quad +\theta_n^2\|x_n - x_{n-1}\|^2 + \delta_n^2\|x_{n-1} - x_{n-2}\|^2 \\
&\quad -(1-2\delta)[\|z_n - w_n\|^2 + \|y_n - z_n\|^2]\Big] \\
&\quad +\gamma_n k\|x_n - x^*\|^2 + 2\gamma_n\langle F(x^*) - x^*, x_{n+1} - x^*\rangle \\
&\leq (1-(1-k)\gamma_n)\|x_n - x^*\|^2 \\
&\quad +2\theta_n\|x_n - x^*\| \cdot \|x_n - x_{n-1}\| \\
&\quad +2\,|\delta_n| \cdot \|x_n - x^*\| \cdot \|x_{n-1} - x_{n-2}\| \\
&\quad +2\theta_n\,|\delta_n| \cdot \|x_n - x_{n-1}\| \cdot \|x_{n-1} - x_{n-2}\| \\
&\quad +\theta_n^2\|x_n - x_{n-1}\|^2 + \delta_n^2\|x_{n-1} - x_{n-2}\|^2 \\
&\quad -(1-2\delta)[\|z_n - w_n\|^2 + \|y_n - z_n\|^2] \\
&\quad +2\gamma_n\langle F(x^*) - x^*, x_{n+1} - x^*\rangle.
\end{aligned}
$$

Since

$$\theta_n\|x_n - x_{n-1}\| = \gamma_n \cdot \frac{\theta_n}{\gamma_n}\|x_n - x_{n-1}\| \to 0,$$

and

$$|\delta_n| \cdot \|x_{n-1} - x_{n-2}\| = \gamma_n \cdot \frac{|\delta_n|}{\gamma_n}\|x_{n-1} - x_{n-2}\| \to 0$$

as $n \to \infty$, there exists $M_3, M_4 > 0$ such that

$$\theta_n\|x_n - x_{n-1}\| \leq M_3,$$

and

$$|\delta_n| \cdot \|x_{n-1} - x_{n-2}\| \leq M_4.$$

It implies that

$$
\begin{aligned}
\|x_{n+1}-x^*\| &\leq (1-(1-k)\gamma_n)\|x_n-x^*\|^2 \\
&\quad +\theta_n\|x_n-x_{n-1}\|\Big[2\|x_n-x^*\| \\
&\quad +2\,|\delta_n|\cdot\|x_{n-1}-x_{n-2}\|+\theta_n\|x_n-x_{n-1}\|\Big] \\
&\quad +|\delta_n|\cdot\|x_{n-1}-x_{n-2}\| \\
&\quad \times[2\|x_n-x^*\|+|\delta_n|\cdot\|x_{n-1}-x_{n-2}\|] \\
&\quad -(1-2\delta)[\|z_n-w_n\|^2+\|y_n-z_n\|^2] \\
&\quad +2\gamma_n\langle F(x^*)-x^*,x_{n+1}-x^*\rangle \\
&= (1-(1-k)\gamma_n)\|x_n-x^*\|^2 \\
&\quad +(1-k)\gamma_n\cdot\tfrac{\theta_n}{\gamma_n}\|x_n-x_{n-1}\| \\
&\quad \times\tfrac{2\|x_n-x^*\|+2|\delta_n|\cdot\|x_{n-1}-x_{n-2}\|+\theta_n\|x_n-x_{n-1}\|}{1-k} \\
&\quad +(1-k)\gamma_n\cdot\tfrac{|\delta_n|}{\gamma_n}\|x_{n-1}-x_{n-2}\| \\
&\quad \times\tfrac{2\|x_n-x^*\|+|\delta_n|\cdot\|x_{n-1}-x_{n-2}\|}{1-k} \\
&\quad -(1-2\delta)[\|z_n-w_n\|^2+\|y_n-z_n\|^2] \\
&\quad +2\gamma_n\langle F(x^*)-x^*,x_{n+1}-x^*\rangle \\
&\leq (1-(1-k)\gamma_n)\|x_n-x^*\|^2+(1-k)\gamma_n b_n \\
&\quad -(1-2\delta)[\|z_n-w_n\|^2+\|y_n-z_n\|^2]
\end{aligned}
\tag{3.29}
$$

where

$$b_n := \frac{1}{1-k}[5M_5\frac{\theta_n}{\gamma_n}\|x_n-x_{n-1}\|+3M_5\frac{|\delta_n|}{\gamma_n}\|x_{n-1}-x_{n-2}\|+2\langle f(x^*)-x^*,x_{n+1}-x^*\rangle]$$

and $M_5=\max\{\sup_n\|x_n-x^*\|, M_3, M_4\}$.

It follows that

$$
\begin{aligned}
(1-2\delta))[\|z_n-w_n\|^2+\|y_n-z_n\|^2] &\leq \|x_n-x^*\|^2-\|x_{n+1}-x^*\|^2 \\
&\quad +(1-k)\gamma_n M
\end{aligned}
\tag{3.30}
$$

where $M=\sup_n b_n$.

From (3.29), we set

$$a_n := \|x_n-x^*\|^2 \text{ and } \xi_n := (1-k)\gamma_n.$$

Hence, we obtain

$$a_{n+1}\leq(1-\xi_n)a_n+\xi_n b_n.$$

Suppose $\{a_{n_i}\}$ is a subsequence of $\{a_n\}$ satisfying

$$\liminf_{i\to\infty}(a_{n_i+1}-a_{n_i})\geq 0.$$

By (3.30) and conditions 2.2), we have

$$
\begin{aligned}
\limsup_{i\to\infty}(1-2\delta))[\|z_{n_i}-w_{n_i}\|^2+\|y_{n_i}-z_{n_i}\|^2] &\leq \limsup_{i\to\infty}(a_{n_i}-a_{n_i+1}) \\
&\quad +(1-k)M\lim_{i\to\infty}\gamma_{n_i} \\
&= -\liminf_{i\to\infty}(a_{n_i+1}-a_{n_i}) \\
&\leq 0,
\end{aligned}
$$

which implies

$$\lim_{i\to\infty}\|z_{n_i}-w_{n_i}\|=\lim_{i\to\infty}\|y_{n_i}-z_{n_i}\|=0. \tag{3.31}$$

Using conditions 2.2) - 2.3), and (3.31), we have

$$\begin{aligned}\|x_{n_i+1}-x_{n_i}\| &\leq \gamma_{n_i}\|F(x_{n_i})-x_{n_i}\|+\|y_{n_i}-x_{n_i}\|\\ &\leq \gamma_{n_i}\|F(x_{n_i})-x_{n_i}\|+\|y_{n_i}-w_{n_i}\|+\|w_{n_i}-x_{n_i}\|\\ &\leq \gamma_{n_i}\|F(x_{n_i})-x_{n_i}\|+\|y_{n_i}-z_{n_i}\|+\|z_{n_i}-w_{n_i}\|\\ &\quad+\gamma_{n_i}\frac{\theta_{n_i}}{\gamma_{n_i}}\|x_{n_i}-x_{n_i-1}\|+\gamma_{n_i}\frac{|\delta_{n_i}|}{\gamma_{n_i}}\cdot\|x_{n_i-1}-x_{n_i-2}\|\\ &\to 0\end{aligned} \tag{3.32}$$

as $i\to\infty$. We next show that $\limsup_{i\to\infty}\langle F(x^*)-x^*,x_{n_i+1}-x^*\rangle\leq 0$.

Let $\{x_{n_{i_j}}\}$ be a subsequence of $\{x_{n_i}\}$ such that

$$\lim_{j\to\infty}\langle F(x^*)-x^*,x_{n_{i_j}}-x^*\rangle=\limsup_{i\to\infty}\langle F(x^*)-x^*,x_{n_i}-x^*\rangle.$$

Since $\{x_{n_{i_j}}\}$ is bounded, there exists a subsequence $\{x_{n_{i_{j_k}}}\}$ of $\{x_{n_{i_j}}\}$ such that $x_{n_{i_{j_k}}}\rightharpoonup\bar{x}\in\mathcal{H}$. Without loss of generality, we may assume that $x_{n_{i_j}}\rightharpoonup\bar{x}$. Thus, we also have $z_{n_{i_j}}\rightharpoonup\bar{x}$.

From Assumption 1.1 (A1), we have $\|\nabla f(w_{n_{i_j}})-\nabla f(z_{n_{i_j}})\|\to 0$ as $j\to\infty$. This together with (3.31) and condition 2.1) yields

$$\left\|\frac{w_{n_{i_j}}-z_{n_{i_j}}}{\alpha_{n_{i_j}}}+\nabla f(z_{n_{i_j}})-\nabla f(w_{n_{i_j}})\right\|\to 0 \tag{3.33}$$

as $j\to\infty$. By (2.11), we obtain that

$$\frac{w_{n_{i_j}}-z_{n_{i_j}}}{\alpha_{n_{i_j}}}+\nabla f(z_{n_{i_j}})-\nabla f(w_{n_{i_j}})\in\partial g(z_{n_{i_j}})+\nabla f(z_{n_{i_j}})=\partial h(z_{n_{i_j}}) \tag{3.34}$$

Using (3.33), (3.34) and $z_{n_{i_j}}\rightharpoonup\bar{x}$, it follows from Lemma 2.2 that $0\in\partial h(\bar{x})$. Hence, $\bar{x}\in\Gamma$.

From (3.32) and (2.10), we obtain that

$$\begin{aligned}\limsup_{i\to\infty}\langle F(x^*)-x^*,x_{n_i+1}-x^*\rangle &\leq \limsup_{i\to\infty}\langle F(x^*)-x^*,x_{n_i+1}-x_{n_i}\rangle\\ &\quad+\limsup_{i\to\infty}\langle F(x^*)-x^*,x_{n_i}-x^*\rangle\\ &= \lim_{j\to\infty}\langle F(x^*)-x^*,x_{n_{i_j}}-x^*\rangle\\ &= \langle F(x^*)-x^*,\bar{x}-x^*\rangle\\ &\leq 0.\end{aligned}$$

By Lemma 2.3, we can conclude that $x_n\to x^*$. □

Next, we will employ Algorithm 7 for solving a convex bilevel optimization problem (1.1) and (1.2).

Algorithm 8 An double inertial forward-backward algorithm with linesearch (DIFBAL)

Initialization: Choose $x_1,\ x_0,\ x_{-1} \in \mathcal{H}, \sigma > 0, \delta \in (0, \frac{1}{8})$ and $\theta \in (0,1)$. Take $\{\gamma_n\}, \{\tau_n\} \subset (0,\infty), \{\mu_n\} \subset (0,\infty)$ and $\{\rho_n\} \subset (-\infty, 0)$.
Iterative steps: For $n \geq 1$, calculate x_{n+1} as follows:
Step 1. Compute the inertial parameters θ_n and δ_n by

$$\theta_n = \begin{cases} \min\{\mu_n, \frac{\tau_n}{\|x_n - x_{n-1}\|}\}, & \text{if } x_n \neq x_{n-1}; \\ \mu_n, & \text{otherwise,} \end{cases} \tag{3.35}$$

and

$$\delta_n = \begin{cases} \max\{\rho_n, \frac{-\tau_n}{\|x_{n-1} - x_{n-2}\|}\}, & \text{if } x_n \neq x_{n-1}; \\ \rho_n, & \text{otherwise,} \end{cases} \tag{3.36}$$

and let

$$w_n = x_n + \theta_n(x_n - x_{n-1}) + \delta_n(x_{n-1} - x_{n-2}). \tag{3.37}$$

Step 2. Compute the forward-backward step:

$$z_n = \text{prox}_{\alpha_n g}(w_n - \alpha_n \nabla f(w_n)) \tag{3.38}$$

$$y_n = \text{prox}_{\beta_n g}(z_n - \beta_n \nabla f(z_n)) \tag{3.39}$$

where $\alpha_n = Linesearch1.1(w_n, \sigma, \theta, \delta)$ and $\beta_n := Linesearch1.1(z_n, \sigma, \theta, \delta)$.
Step 3. Compute the viscosity step:

$$x_{n+1} = \gamma_n(x_n - s\nabla\phi(x_n)) + (1 - \gamma_n)y_n. \tag{3.40}$$

Set n := n+1 and return to Step 1.

The following result is obtained directly by Theorem 3.2.

Theorem 3.3. *Let $\{x_n\}$ be a sequence generated by Algorithm 8 with the same condition as in Theorem 3.2. Then, $x_n \to x^* \in \Omega$ where $x^* = P_\Omega f(x^*)$.*

Proof. Set $f = I - s\nabla\phi$ in Theorem 3.2. We known that $I - s\nabla\phi$ is a contraction. By Theorem 3.2, we obtain that $x_n \to x^* \in \Gamma$, where $x^* = P_\Gamma f(x^*)$. From equation (2.10), it can be obtained that for any $x \in \Omega$,

$$\begin{aligned} 0 &\geq \langle f(x^*) - x^*, x - x^*)\rangle \\ &= \langle (x^* - s\nabla\phi(x^*)) - x^*, x - x^*\rangle \\ &= -s\langle \nabla\phi(x^*), x - x^*\rangle \end{aligned} \tag{3.41}$$

□

4. Applications

For applications, we explore the practical implementation of our proposed algorithm in the field of machine learning, specifically using the Extreme Learning Machine (ELM), which is introduced by Huang et al. [17]. ELM is a widely used neural network model known for its fast learning speed and high generalization performance. Our algorithm is integrated into the ELM framework to enhance optimization efficiency and improve classification accuracy.

To demonstrate the effectiveness of our algorithm, we apply it to real-world data classification problems, including datasets related to diabetes and breast cancer. Additionally, we apply our algorithm on a real dataset from Sriphat Medical Center at Chiang Mai University called Hypertension. We compare the performance of our method against existing algorithms in terms of accuracy, precision, recall, and the F1 score.

Let $\{(x_n, t_n) \in \mathbb{R}^n \times \mathbb{R}^m : n = 1, 2, \ldots, s\}$ be a given training dataset consisting of s samples, where x_n represents the input and t_n denotes the corresponding target output.

The Extreme Learning Machine (ELM) is a fast-learning algorithm designed for Single-Layer Feedforward Networks (SLFNs). The mathematical formulation of ELM for SLFNs is given by:

$$o_n = \sum_{j=1}^{h} \eta_j G(\langle \omega_j, x_n \rangle + b_j),\ n = 1, 2, \ldots, s,$$

where:

- o_n is the predicted output,
- h represents the number of hidden nodes,
- $G(\cdot)$ is the activation function,
- ω_j and η_j are weight vectors connecting the j-th hidden node to the input and output nodes, respectively,
- b_j is the bias term associated with the j-th hidden node.

The hidden layer output matrix, denoted by **H**, is structured as follows:

$$\mathbf{H} = \begin{bmatrix} G(\langle \omega_1, x_1 \rangle + b_1) & \cdots & G(\langle \omega_h, x_1 \rangle + b_h) \\ \vdots & \ddots & \vdots \\ G(\langle \omega_1, x_s \rangle + b_1) & \cdots & G(\langle \omega_h, x_s \rangle + b_h) \end{bmatrix}_{s \times h}.$$

The objective of SLFNs is to approximate the given training samples such that the total error is minimized:

$$t_n = \sum_{j=1}^{h} \eta_j G(\langle \omega_j, x_n \rangle + b_j),\ n = 1, 2, \ldots, s.$$

The above equation can be rewritten in a compact matrix form as:

$$\mathbf{H}u = \mathbf{T}, \tag{4.42}$$

where:

- $u = [\eta_1^T, \cdots, \eta_h^T]^T$ represents the output weight vector,
- $\mathbf{T} = [t_1^T, \cdots, t_s^T]^T$ denotes the target output matrix.

To solve u, ELM randomly assigns values to ω_j and b_j and focuses on determining u. However, when the number of hidden nodes is smaller than the number of training samples (i.e., $h < s$), **H** becomes a non-square matrix, making the equation potentially inconsistent. In such cases, the Moore-Penrose inverse provides a least-squares solution:

$$\hat{u} = \mathbf{H}^{+}\mathbf{T}, \tag{4.43}$$

where $\mathbf{H}^{+}$ is the Moore-Penrose generalized inverse of **H**.

This solution minimizes the training error in the least-squares sense:

$$\|\mathbf{H}\hat{u} - \mathbf{T}\|_2^2 = \min_u \|\mathbf{H}u - \mathbf{T}\|_2^2. \tag{4.44}$$

To improve generalization and prevent overfitting, least absolute shrinkage and selection operator (LASSO) is employed. The regularized problem is formulated as:

$$\min_u \|\mathbf{H}u - \mathbf{T}\|_2^2 + \lambda \|u\|_1, \tag{4.45}$$

where:

- $\|\cdot\|_1$ denotes the l_1-norm, defined as $\|(x_1,\ldots,x_n)\|_1 = \sum_{i=1}^{n} |x_i|$,
- $\lambda > 0$ is the regularization parameter.

This problem can be rewritten in the form of bilevel optimization:

- Define $f(u) := \|\mathbf{H}u - \mathbf{T}\|_2^2$,
- Define $g(u) := \lambda\|u\|_1$,
- For the outer level problem, set $\phi(u) := \frac{1}{2}\|u\|_2^2$ with constants $L_\phi = 1$ and $\sigma_\phi = 1$.

This formulation allows the integration of our proposed algorithm into the ELM framework, offering an efficient approach to training SLFNs with improved convergence and classification performance.

The following table provides details on the experimental setup, dataset descriptions, and comparative results.

For optimal efficiency in our experiment, we carefully selected parameters by choosing the most advantageous configuration for each algorithm, as presented in Table 1.

TABLE 1. The setting of parameters for each algorithms

Parameters	Algorithm 8	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4	Algorithm 5	Algorithm 6
σ	1	-	-	-	-	-	1
γ_n	$0.003 + \frac{1}{50n}$	-	-	-	-	-	$\frac{1}{50n}$
θ	0.9	-	-	-	-	-	0.9
δ	0.1	-	-	-	-	-	0.1
μ_n	$\frac{n-1}{n+\alpha-1}$	-	-	-	-	-	$\frac{n}{n+1}$
ρ_n	-0.00001	-	-	-	-	-	-
τ_n	$\frac{33\cdot 10^{20}}{n}$	-	$\frac{1}{(n+1)^2}$	$\frac{1}{(n+1)^2}$	$\frac{1}{(n+1)^2}$	$\frac{1}{(n+1)^2}$	$\frac{10^{50}}{n^2}$
c_n	-	$\frac{n\cdot 10^{-5}}{(n+1)\cdot L_F}$	$\frac{n\cdot 10^{-5}}{(n+1)\cdot L_F}$	$\frac{1}{L_F}$	$\frac{1}{L_F}$	$\frac{1}{L_F}$	-
s	0.001	0.001	0.001	0.001	0.001	0.001	0.001
α	3	-	3	3	3	3	-
q	-	-	-	-	4	4	-

In addition, we set

- $\lambda = 10^{-5}$
- G is sigmoid function,
- $h = 30$,
- $\nabla f(u) = 2\mathbf{H}^T(\mathbf{H}u - T)$.

In our experiments, we focus on classifying the Diabetes and Breast Cancer datasets from the UCI Machine Learning Repository into two distinct classes. Our goal is to evaluate the performance of different algorithms in solving this classification task.

To assess the effectiveness of our proposed approach, we compare it with several existing algorithms, including:

- Algorithm 8 (our proposed method),
- Algorithm 1 by Sabach and Shtern [4],
- Algorithm 2 by Shehu et al. [5],
- Algorithm 3, Algorithm 4, and Algorithm 5 by Duan and Zhang [7].

The performance of each algorithm is analyzed at different iteration numbers: 100th, and 500th iterations. The experimental results, detailing the classification accuracy(acc.), precision(pre.), recall(rec), and the F1 score, are presented in Table 2 and Table 3 for the each datasets at 100th and 500th iterations, respectively.

TABLE 2. The effectiveness of each algorithm for each dataset with 10-fold CV. at 100 iterations.

Datasets	Algorithm	acc. train	acc.test	pre. train	pre.test	rec. train	rec.test	F1 train	F1 test
Diabetes	Algorithm 7	94.4699	94.4873	0.9349	0.9323	0.9717	0.9800	0.9529	0.9543
	Algorithm 1	86.6716	85.7717	0.9384	0.9328	0.8225	0.8123	0.8766	0.8644
	Algorithm 2	91.9212	91.2949	0.9135	0.9164	0.9495	0.9362	0.9311	0.9249
	Algorithm 3	92.0742	91.2949	0.9155	0.9164	0.9500	0.9362	0.9324	0.9249
	Algorithm 4	91.9722	91.2949	0.9140	0.9164	0.9500	0.9362	0.9362	0.9249
	Algorithm 5	91.9722	91.2949	0.9140	0.9164	0.9500	0.9362	0.9362	0.9249
	Algorithm 6	94.3934	93.8055	0.9400	0.9330	0.9641	0.9642	0.9519	0.9475
Breast cancer	Algorithm 7	97.4947	97.5128	0.9549	0.9552	0.9744	0.9748	0.9646	0.9648
	Algorithm 1	97.0880	97.0737	0.9298	0.9301	0.9916	0.9915	0.9597	0.9596
	Algorithm 2	96.2420	95.8994	0.9532	0.9527	0.9386	0.9290	0.9459	0.9403
	Algorithm 3	96.2420	95.8994	0.9532	0.9527	0.9386	0.9290	0.9459	0.9403
	Algorithm 4	96.2420	95.8994	0.9532	0.9527	0.9386	0.9290	0.9459	0.9403
	Algorithm 5	96.2420	95.8994	0.9532	0.9527	0.9386	0.9290	0.9459	0.9403
	Algorithm 6	97.4622	97.3679	0.9549	0.9548	0.9735	0.9707	0.9641	0.9626
Hypertension	Algorithm 7	89.1363	89.1127	0.8478	0.8496	0.9314	0.9282	0.8876	0.8871
	Algorithm 1	87.6044	87.5085	0.8192	0.8188	0.9379	0.9367	0.8745	0.8736
	Algorithm 2	88.3882	88.2780	0.8420	0.8418	0.9206	0.9189	0.8796	0.8785
	Algorithm 3	88.4025	88.2943	0.8421	0.8420	0.9209	0.9189	0.8797	0.8787
	Algorithm 4	88.4025	88.2943	0.8421	0.8420	0.9209	0.9189	0.8797	0.8787
	Algorithm 5	88.4025	88.2943	0.8421	0.8420	0.9209	0.9189	0.8797	0.8787
	Algorithm 6	88.9053	88.8181	0.8459	0.8448	0.9282	0.9282	0.8851	0.8844

Table 2 demonstrates that at the 100th iteration our method outperforms the others in terms of accuracy, precision, recall, and F1-score across all datasets.

TABLE 3. The effectiveness of each algorithm for each dataset with 10-fold CV. at 500 iterations.

Dataset s	Algorithm	acc. train	acc.test	pre. train	pre.test	rec. train	rec.test	F1 train	F1 test
Diabetes	Algorithm 7	96.0499	94.9524	0.9439	0.9379	0.9903	0.9800	0.9665	0.9575
	Algorithm 1	90.0357	88.5095	0.9273	0.9239	0.8973	0.8763	0.9120	0.8972
	Algorithm 2	90.8257	90.3805	0.8945	0.8946	0.9531	0.9482	0.9228	0.9193
	Algorithm 3	90.8002	90.3805	0.8945	0.8946	0.9526	0.9482	0.9226	0.9193
	Algorithm 4	90.8002	90.3805	0.8945	0.8946	0.9526	0.9482	0.9226	0.9193
	Algorithm 5	90.8002	90.3805	0.8945	0.8946	0.9526	0.9482	0.9226	0.9193
	Algorithm 6	95.2090	93.5835	0.9379	0.9274	0.9819	0.9682	0.9594	0.9462
Breast cancer	Algorithm 7	97.7225	97.6620	0.9619	0.9632	0.9735	0.9707	0.9676	0.9667
	Algorithm 1	97.2181	97.2165	0.9297	0.9304	0.9958	0.9957	0.9616	0.9617
	Algorithm 2	97.2507	97.0759	0.9584	0.9583	0.9633	0.9582	0.9608	0.9581
	Algorithm 3	97.2507	97.0759	0.9584	0.9583	0.9633	0.9582	0.9608	0.9581
	Algorithm 4	97.2507	97.0759	0.9584	0.9583	0.9633	0.9582	0.9608	0.9581
	Algorithm 5	97.2507	97.0759	0.9584	0.9583	0.9633	0.9582	0.9608	0.9581
	Algorithm 6	97.6411	97.5149	0.9584	0.9590	0.9749	0.9707	0.9666	0.9646
Hypertension	Algorithm 7	89.5874	89.7017	0.8675	0.8691	0.9134	0.9143	0.8899	0.8911
	Algorithm 1	87.4463	87.3446	0.8114	0.8100	0.9476	0.9477	0.8743	0.8734
	Algorithm 2	88.0685	88.0485	0.8330	0.8333	0.9267	0.9261	0.8774	0.8771
	Algorithm 3	88.0794	88.0812	0.8331	0.8338	0.9268	0.9261	0.8775	0.8774
	Algorithm 4	88.1340	88.1958	0.8377	0.8387	0.9207	0.9214	0.8773	0.8780
	Algorithm 5	88.1340	88.1958	0.8377	0.8387	0.9207	0.9214	0.8773	0.8780
	Algorithm 6	89.2236	89.0634	0.8542	0.8522	0.9236	0.9229	0.8876	0.8860

As shown in Table 3, at the 500th iteration, our algorithm also delivers superior performance compared to the others, achieving higher accuracy, precision, recall, and F1-score.

5. CONCLUSIONS

In this work, we propose a new double inertial accelerated algorithm with linesearch technique and analyze its strong convergence theorem under some suitable conditions. Consequently, the algorithm can be effectively applied to convex bilevel optimization problems. Additionally, we employ it as a machine learning model for data classification of noncommunicable diseases and compare its performance with existing algorithms. The results indicate that our algorithm outperforms the other algorithms in the literature.

Acknowledgments

This research was partially supported by Chiang Mai University and Fundamental Fund 2025, Chiang Mai University. The first author would like to thank CMU Presidential Scholarship for the financial support.

REFERENCES

[1] Kesornprom, S.; Cholamjiak, P. Strong convergence of the modified projection and contraction methods for split feasibility problem. *Thai J. Math.* (2018), 76-94.
[2] Kesornprom, S.; Pholasa, N.; Cholamjiak, P. A modified CQ algorithm for solving the multiple-sets split feasibility problem and the fixed point problem for nonexpansive mappings. *Thai J. Math.* **17** (2019), no. 2, 475-493.
[3] Thongsri, P.; Suantai, S. New Accelerated Fixed Point Algorithms with Applications to Regression and Classification Problems. *Thai J. Math.* **18** (2020), no. (4), 2001-2011.
[4] Sabach, S.; Shtern, S. A first order method for solving convex bilevel optimization problems. *SIAM J. Optim.* **27** (2017), no. (2), 640-660.
[5] Shehu, Y.; Vuong, P. T.; Zemkoho, A. An inertial extrapolation method for convex simple bilevel optimization. *Optim. Method. Softw.* **36** (2021), no. (1), 1-19.
[6] Polyak, B. T. Some methods of speeding up the convergence of iteration methods. *USSR Comput. Math. Math. Phys.* **4** (1964), no. (5), 1-17.
[7] Duan, P.; Zhang, Y. Alternated and multi-step inertial approximation methods for solving convex bilevel optimization problems. *Optim.* **72** (2023), 2517-2545.
[8] Wattanataweekul, R.; Janngam, K.; Suantai, S. A novel two-step inertial viscosity algorithm for bilevel optimization problems applied to image recovery. *Math.* **11** (2023), no. (16), 3518.
[9] Sae-jia, P.; Suantai, S. A novel algorithm for convex Bi-level optimization problems in Hilbert spaces with applications. *Thai J. Math.* **21** (2023), no. (3), 625-645.
[10] Sae-jia, P.; Suantai, S. A new two-step inertial algorithm for solving convex bilevel optimization problems with application in data classification problems. *AIMS Math.* **9** (2024), no. (4), 8476-8496.
[11] Wattanataweekul, R.; Janngam, K.; Suantai, S. A novel double inertial viscosity algorithm for convex bilevel optimization problems applied to image restoration problems. *Optim.* (2024), 1-22.
[12] Bello Cruz, J. Y.; Nghia, T. T. On the convergence of the forward–backward splitting method with linesearches. *Optim. Methods Softw.* **31** (2016), no. (6), 1209-1238.
[13] Suantai, S.; Jailoka, P.; Hanjing, A. An accelerated viscosity forward-backward splitting algorithm with the linesearch process for convex minimization problems. *J. Inequal. Appl.* **2021**, (2021), 1-19.
[14] Takahashi, W. *Introduction to nonlinear and convex analysis*. Yokohama Publishers, 2009.
[15] Rockafellar, R. On the maximal monotonicity of subdifferential mappings. *Pac. J. Math* **33** (1970), no. (1), 209-216.
[16] Saejung, S.; Yotkaew, P. Approximation of zeros of inverse strongly monotone operators in Banach spaces. *Nonlinear Anal. Theory Methods Appl.* **75** (2012), no. (2), 742-750.
[17] Huang, G. B.; Zhu, Q. Y.; Siew, C. K. Extreme learning machine: theory and applications. *Neurocomputing* **70** (2006), no. (1-3), 489-501.

[1] PHD DEGREE PROGRAM IN MATHEMATICS, DEPARTMENT OF MATHEMATICS, FACULTY OF SCIENCE, CHIANG MAI UNIVERSITY, UNDER THE CMU PRESIDENTIAL SCHOLARSHIP.
Email address: puntita_saejia@cmu.ac.th (P. Sae-jia)

[2] RESEARCH CENTER IN OPTIMIZATION AND COMPUTATIONAL INTELLIGENCE FOR BIG DATA PREDICTION, DEPARTMENT OF MATHEMATICS, FACULTY OF SCIENCE, CHIANG MAI UNIVERSITY, CHIANG MAI, 50200, THAILAND.
Email address: suthep.s@cmu.ac.th (S. Suantai)