CARPATHIAN J. MATH.
Volume **40** (2024), No. 3,
Pages 769-788

Online version at `https://www.carpathian.cunbm.utcluj.ro/`
Print Edition: ISSN 1584 - 2851; Online Edition: ISSN 1843 - 4401
DOI: https://doi.org/10.37193/CJM.2024.03.15

A Fast Forward-Backward Algorithm Using Linesearch and Inertial Techniques for Convex Bi-level Optimization Problems with Applications

SUTHEP SUANTAI and PITI THONGSRI

ABSTRACT. In this research, we study convex bi-level optimization problems for which the inner level consists of the sum of two proper, convex, and lower semi-continuous functions. We propose and analyze a new accelerated forward-backward algorithm using linesearch and inertial techniques for solving a solution of convex bi-level optimization. We then establish a strong convergence theorem of the proposed method under some suitable conditions. As an application, we apply our algorithm to solving data classifications of some non-communicable diseases. We conduct a comparative analysis with existing algorithms to show the effectiveness of our algorithm. Our numerical experiments confirm that our proposed algorithm outperforms other methods in the literature.

1. INTRODUCTION

Bilevel optimization represents a specific category of mathematical optimization problems wherein one optimization problem is nested within another optimization problem. The solution to the outer problem is dependent on the solution to the inner problem. The difficulty of such problem is finding the optimal solution to both the leader and the follower problem simultaneously.

Let H be a real Hilbert space and let h_1, h_2 be functions that map from H to $\mathbb{R}$. The outer level problem is the minimization problem of the following form:

$$\min_{x \in X^*} g(x), \tag{1.1}$$

where $g : H \to \mathbb{R}$ is a continuously differentiable and strongly convex function with parameter $\rho > 0$ such that ∇g is Lipschitz continuous with constant L_g while X^* is the set of all minimizers of the inner level optimization problem of the following form:

$$\min_{x \in \mathbb{R}^n} \{h_1(x) + h_2(x)\}. \tag{1.2}$$

For solving Problem (1.2), we normally assume the following assumptions

(a) $h_1 : H \to \mathbb{R}$ is convex and differentiable for which ∇h_1 is L_{h_1}–Lipschitz continuous, that is,

$$\|\nabla h_1(x) - \nabla h_1(y)\| \leq L_{h_1}\|x - y\| \text{ for all } x, y \in \mathbb{R}^n;$$

(b) $h_2 : H \to \mathbb{R}$ is proper convex and lower semi-continuous.

The solution to Problem (1.2) can be described according to Theorem 16.3 of Bauschke and Combettes [1] as follows:

$$\bar{p} \in X^* \text{ if and only if } 0 \in \partial h_2(\bar{p}) + \nabla h_1(\bar{p}),$$

Received: 15.11.2023. In revised form: 28.03.2024. Accepted: 11.04.2024

2010 *Mathematics Subject Classification.* 90C30, 90C25, 65K10.

Key words and phrases. *convex bi-level optimization problems, Hilbert spaces, forward-backward algorithm, Linesearch, classification problems, non-communicable diseases.*

Corresponding author: Piti Thongsri; piti_tho@cmu.ac.th

where ∇h_1 is the gradient of h_1 and ∂h_2 is the subdifferential of h_1. On the other hand, Problem (1.2) is equivalent with the following fixed point problem:

$$\bar{p} \in X^* \text{ if and only if } \bar{p} = prox_{\alpha h_2}(I - \alpha \nabla h_1)(\bar{p}),$$

where $prox_{\alpha h_2}(x) = argmin_{y \in H}(h_2(y) + \frac{1}{2\alpha}\|x - y\|^2)$ and $\alpha > 0$. The operator $prox_{\alpha h_2}(I - \alpha \nabla h_1)$ is called the forward-backward operator of h_1 and h_2 with respect to α. We also know that $prox_{\alpha h_2}(I - \alpha \nabla h_1)$ is a nonexpansive operator when $\alpha \in (0, 2/L_{h_1})$ and L_{h_1} is a Lipschitz constant of ∇h_1. From basic principle of optimization, we know that $\bar{p} \in X^*$ is a minimizer of Problem (1.1) if and only if

$$\langle \nabla g(\bar{p}), x - \bar{p} \rangle \geq 0 \text{ for all } x \in X^*. \tag{1.3}$$

Throughout the past decade, many researchers have dedicated their efforts to seeking optimal solutions for the Problem (1.2). Lions and Mercier [10] introduced a method called Forward-Backward Splitting (FBS) as a simple algorithm to address the Problem (1.2). Their algorithm was formulated as follows:

$$x_{n+1} = prox_{\alpha_k h_2}(I - \alpha_k \nabla h_1)(x_k), \tag{1.4}$$

where the step-size $\alpha_k \in (0, 2/L_{h_1})$.

The inertial technique was initially pioneered by Polyak [12] with the aim of accelerating the convergence rate of algorithms. Subsequently, this technique has become widely utilized for this purpose.

For example, Beck and Teboulle [2] introduced a fast iterative shrinkage-thresholding algorithm (FISTA) by using this technique for solving Problem (1.2) as described by the following:

$$\begin{aligned}
&x_1 = v_0 \in C, t_1 = 1,\\
&v_k = prox_{\alpha_k h_2}(I - \alpha_k \nabla h_1)(x_k),\ \alpha > 0,\\
&t_{n+1} = (\sqrt{1 + 4t_k^2} + 1)/2,\\
&\theta_k = t_n - 1/t_{n+1},\\
&x_{n+1} = v_k + \theta_k(v_k - v_{k-1}).
\end{aligned}$$

Moreover, they illustrated that the rate of convergence of FISTA outperforms the other methods.

Recently, many researchers, including Puangpee and Suantai [13], Jailoka et al. [8] and Thongsri et al. [22], have employed the inertial technique into their proposed algorithms in order to accelerate the convergence behavior of their algorithms for solving Problem (1.2). They proposed common fixed point algorithms for a countable families of nonexpansive operators. Furthermore, they applied those algorithms to solve some convex minimization problems.

In 2017, Sabach and Shtern [15] introduced a new technique called Sequential Averaging Method (SAM) for solving convex bi-level optimization problems. They adapted an approach from [25], originally designed for a specific fixed point problem, to suit this context. Subsequently, they developed the Bilevel Gradient Sequential Averaging Method (BiG-SAM) to address the convex bi-level optimization Problems (1.1) and (1.2). The formulation of BiG-SAM was outlined in Algorithm 1.

Subsequently, they demonstrated that the sequence $\{x_k\}$ generated by BiG-SAM converges to a solution of both Problem (1.1) and (1.2), subject to specific control conditions.

Algorithm 1 Bilevel Gradient Sequential Averaging Method (BiG-SAM)

(1) **Input** : $\{\alpha_k\} \in (0, 1/L_{h_1})$, $s \in (0, 2/(\rho + L_g))$ and $\{\beta_k\} \subset (0, 1]$.
(2) **Initialization** : choose $x_1 \in \mathbb{R}^n$.
(3) **General step** : $(k = 1, 2, ...)$:

$$\begin{aligned} u_k &= prox_{\alpha_k h_2}(x_k - \alpha_k \nabla h_1(x_k)), \\ w_k &= x_k - s\nabla g(x_k), \\ x_{k+1} &= \beta_k w_k + (1 - \beta_k) u_k, \end{aligned}$$

where ∇g is the gradient of g.

In 2019, Shehu et al. [17] applied an inertial technique to improve the convergence behavior of the BiG-SAM. They presented a new algorithm named the inertial Bilevel Gradient Sequential Averaging Method (iBiG-SAM), formally defined in Algorithm 2:

Algorithm 2 Inertial Bilevel Gradient Sequential Averaging Method (iBiG-SAM)

(1) **Input** : $a \geq 3$, $\{\alpha_k\} \in (0, 1/L_{h_1})$, and $s \in (0, 2/(\rho + L_g))$ and $\{\beta_k\} \subset (0, 1]$.
(2) **Initialization** : choose $x_0, x_1 \in \mathbb{R}^n$.
(3) **Step 1** For $k = 1, 2, ...,$

$$\mu_k = \begin{cases} \min\{\frac{k}{k+a-1}, \frac{\gamma_k}{\|x_k - x_{k-1}\|}\}, & \text{if } x_k \neq x_{k-1}, \\ \frac{k}{k+a-1}, & \text{otherwise.} \end{cases}$$

(4) **Step 2** Compute :

$$\begin{aligned} w_k &= x_k + \mu_k(x_k - x_{k-1}), \\ u_k &= prox_{\alpha_k h_2}(w_k - \alpha_k \nabla h_1(w_k)), \\ v_k &= w_k - s\nabla g(w_k), \\ x_{k+1} &= \beta_k v_k + (1 - \beta_k) u_k, \end{aligned}$$

where ∇g is the gradient of g.

Very recently, Duan and Zhang [6] presented a new algorithm for solving convex bilevel optimization problems. This algorithm, called the alternated inertial Bilevel Gradient Sequential Averaging Method (aiBiG-SAM), is based on the proximal gradient algorithm. The algorithm is formally defined in Algorithm 3.

Algorithm 3 alternated inertial Bilevel Gradient Sequential Averaging Method (aiBiG-SAM)

(1) **Input** : $\alpha \geq 3$, $\{\alpha_k\} \in (0, 1/L_{h_1})$ and $s \in (0, 2/(\rho + L_g))$. Set $\lambda > 0$.
(2) **Initialization** : choose $x_0, x_1 \in \mathbb{R}^n$.
(3) **Step 1** For $(k = 1, 2, ...)$:

$$\omega_k = \begin{cases} x_k + \theta_k(x_k - x_{k-1})), & \text{if } k \text{ is odd,} \\ x_k, & \text{if } k \text{ is even.} \end{cases}$$

When k is odd, choose θ_k such that $0 \leq |\theta_k| \leq \varepsilon_k$ where ε_k is defined by

$$\varepsilon_k = \begin{cases} \min\{\frac{k}{k+\alpha-1}, \frac{\gamma_k}{\|x_k - x_{k-1}\|}\}, & \text{if } x_k \neq x_{k-1}, \\ \frac{k}{k+\alpha-1}, & \text{otherwise.} \end{cases}$$

(4) **Step 2** Compute :

$$\begin{aligned} y_k &= prox_{\alpha_k h_2}(\omega_k - \alpha_k \nabla h_1(\omega_k)), \\ z_k &= \omega_k - s\nabla g(\omega_k), \\ x_{k+1} &= \beta_k z_k + (1 - \beta_k) y_k. \end{aligned}$$

(5) **Step 3** If $\|x_k - x_{k-1}\| < \lambda$, then stop.

They also provided an analysis of the strong convergence behavior exhibited by the proposed method under some specific conditions.

Note that all the aforementioned methods require ∇h_1 to be L-Lipschitz continuous. It is noted that this condition is a condition that can be challenging to fulfill in a general context. Therefore, some improvement are still desirable.

In the sequel, we set the standing hypotheses on Problem (1.2) as follows:

(AI) $h_1 : H \to \mathbb{R}$ is a convex and differentiable function and the ∇h_1 is uniformly continuous on H;

(AII) $h_2 : H \to \mathbb{R}$ is proper convex and lower semi-continuous.

We see that the assumption (AI) is weaker condition than the Lipchitz continuity condition on ∇h_1.

To relax the continuity assumption on ∇h_1, Cruz and Nghia [3] introduced a linesearch technique for finding a suitable step-size of the forward-backward operator of h_1 and h_2. This technique does not require the Lipschitz continuous assumption on ∇h_1. More precisely, they proposed the following algorithm (Algorithm 4) for solving Problem (1.2). The algorithm was formally outlined in Algorithm 4.

Algorithm 4

(1) **Input** $x_1 \in H$, $\sigma > 0$, $\delta > 0$, and $\theta \in (0, 1)$.

$$x_{k+1} = prox_{\alpha_k h_2}(x_k - \alpha_k \nabla h_1(x_k)),$$

where $\alpha_k := \textbf{Linesearch}(\mathbf{1})(x_k, \sigma, \theta, \delta)$ is defined as follows:
(2) **Set** $\alpha = \sigma$.
(3) **While**

$$\alpha \|\nabla h_1(FB_\alpha(x)) - \nabla h_1(x)\| > \delta \|FB_\alpha(x) - x\|,$$

do $\alpha = \theta\alpha$.
(4) **End**
(5) **Output** α,
where $FB_\alpha := prox_{\alpha_k h_2}(I - \alpha_k \nabla h_1)$

They demonstrated that Linesearch(1) is well defined and stops after a finite number of steps, as shown in [[3], Lemma 3.1] and [[23], Lemma 3.4 (a)]. Moreover, they proved a weak convergence theorem of the sequence $\{x_k\}$ generated by Algorithm 4 under some suitable conditions.

Later, Kankam et al. [9] proposed an algorithm with a modification of Linesearch(1) defined as follows:

Algorithm 5

(1) **Input** $x_1 \in H$, $\sigma > 0$, $\delta > 0$, and $\theta \in (0, 1)$.

$$w_k = prox_{\alpha_k h_2}(x_k - \alpha_k \nabla h_1(x_k)),$$
$$u_{k+1} = prox_{\alpha_k h_2}(w_k - \alpha_k \nabla h_1(w_k)),$$

where $\alpha_k := \textbf{Linesearch}(\mathbf{2})(x_k, \sigma, \theta, \delta)$ is define as follows :
(2) **Set** $\alpha = \sigma$.
(3) **While**

$$\alpha \max\{\|\nabla h_1(FB_\alpha^2(x)) - \nabla h_1(FB_\alpha(x))\|, \|\nabla h_1(FB_\alpha(x)) - \nabla h_1(x)\|\}$$
$$> \delta \|(FB_\alpha^2(x)) - (FB_\alpha(x))\| + \|(FB_\alpha(x)) - (x)\|,$$

do $\alpha = \theta\alpha$
(4) **End**
(5) **Output** α,
where $FB_\alpha^2(x) := FB_\alpha(FB_\alpha(x))$.

Recently, Jailoka et al. [19] introduced an algorithm with a new linesearch (Linesearch(3)) by modification of Linesearch(1) and Linesearch(2) as follows:

Algorithm 6 Linesearch(3)$(x_k, \sigma, \theta, \delta)$

(1) **Input** $x \in H$, $\sigma > 0$, $\delta > 0$, and $\theta \in (0,1)$.
(2) **Set** $\alpha = \sigma$.
(3) **While**

$$\frac{\alpha}{2}\{\|\nabla h_1(FB_\alpha^2(x)) - \nabla h_1(FB_\alpha(x))\| + \|\nabla h_1(FB_\alpha(x)) - \nabla h_1(x)\|\}$$
$$> \delta(\|FB_\alpha^2(x) - FB_\alpha(x)\| + \|FB_\alpha(x) - x\|),$$

do $\alpha = \theta\alpha$
(4) **End**
(5) **Output** α,
where $FB_\alpha^2(x) := FB_\alpha(FB_\alpha(x))$.

They also proved a strong convergence theorem of their algorithm under some suitable conditions.

Motivated by these previous works, we aim to propose a new efficient algorithm for solving convex bi-level Problems (1.1) and (1.2). Our goal is to establish a strong convergence theorem for the proposed algorithm under some suitable conditions. Furthermore, we apply our algorithm to solve data classification of some non-communicable diseases. The paper is organized as follows. Section 2 offers an overview of the notations and important lemmas that will be used in the subsequent sections. In Section 3, we present a novel accelerated algorithm by using Linesearch(3) and inertial technique for solving Problems (1.1) and (1.2) with assumptions (AI) and (AII). Moving ahead, in Section 4, we apply our algorithm as a machine learning algorithm for solving some data classification problems. Furthermore, we illustrate and analyze the convergence behavior of our method. Finally, the concluding remarks of our paper are in Section 5.

2. Preliminaries

In this section, we provide essential tools that will be utilized in the later sections. The mathematical symbols utilized in this article are defined as follows. $\mathbb{R}, \mathbb{R}_+, \mathbb{R}_{++}$ are the set of real numbers, the set of nonnegative real numbers, and the set of positive real numbers, respectively. Throughout this paper, we represent weak and strong convergence of the sequence $\{x_k\}$ to x as $x_k \rightharpoonup x$ and $x_k \to x$, respectively.

Let C be a nonempty closed convex subset of a Hilbert space H. An operator $T : C \to H$ is said to be *L-Lipschitz* if there exist $L > 0$ such that

$$\|Tx - Ty\| \le L\,\|x - y\| \text{ for all } x, y \in C.$$

If T is Lipschitz continuous with a coefficient $L = 1$, then T is called a nonexpansive. The operator T is said to be contraction if $L \in (0,1)$.

The metric projection from H onto C, denoted by P_C, is defined as follows. For each $x \in H$, $P_C x$ is the unique element in C such that $\|x - P_C x\| = \inf_{z\in C}\|x - z\|$. It is known that

$$p^* = P_C x \Longleftrightarrow \langle x - p^*, z - p^*\rangle \le 0, \text{ for all } z \in C.$$

Note that if $g : H \to \mathbb{R} \cup \{\infty\}$ is a proper, lower semi-continuous and convex function, then the $prox_g(x)$ exists and unique for all $x \in \mathbb{R}^n$; see [4].

Let $g : H \to \mathbb{R} \cup \{\infty\}$ be a proper, lower semi-continuous and convex function. The subdifferential ∂g of g is defined by

$$\partial g(x) := \{z \in H : g(x) + \langle z, y - x\rangle \le g(y), \text{ for all } y \in H\}, \text{ for all } x \in H.$$

Here, we give certain relationships between the proximity operator and the subdifferential operator. For $\alpha > 0$ and $x \in H$, then

$$prox_{\alpha g} = (I_d + \alpha\partial g)^{-1} : H \to \text{dom } g, \tag{2.5}$$

$$\frac{x - prox_{\alpha g}(x)}{\alpha} \in \partial g(prox_{\alpha g}(x)) \tag{2.6}$$

We end this part with the following useful lemmas.

Lemma 2.1. ([20, 21]) *For any $x, y \in H$ and $\mu \in [0,1]$, the following statements hold:*

(i) $\|\mu x + (1-\mu)y\|^2 = \mu\|x\|^2 + (1-\mu)\|y\|^2 - \mu(1-\mu)\|x-y\|^2$;

(ii) $\|x \pm y\|^2 = \|x\|^2 \pm 2\langle x, y\rangle + \|y\|^2$ *for all* $x, y \in H$;

(iii) $\|x + y\|^2 \leq \|x\|^2 + 2\langle y, x + y\rangle$ *for all* $x, y \in H$.

Lemma 2.2. ([14]) *Let $g : H \to \mathbb{R}\cup\{\infty\}$ be a proper, lower semi-continuous and convex function. Let $\{x_k\}$ and $\{z_k\}$ be a two sequences in H such that $z_k \in \partial g(x_k)$ for all $k \in \mathbb{N}$. If $x_k \rightharpoonup x$ and $z_k \to z$, then $z \in \partial g(x)$.*

Lemma 2.3. [16] *Let $\{a_k\}$ be a sequence of nonnegative real numbers and $\{b_k\}$ a sequence of real numbers. Let $\{\zeta_k\}$ be asequence of real numbers in $(0,1)$ such that $\sum_{n=1}^{\infty}\zeta_k = \infty$. Assume that*

$$a_{k+1} \leq (1-\zeta_k)a_k + \zeta_k b_k, \; k \in \mathbb{N}.$$

If $\limsup_{i\to\infty} b_{k_i} \leq 0$ for every subsequence $\{a_{k_i}\}$ of $\{a_k\}$ satisfying

$$\liminf_{i\to\infty}(a_{k_{i+1}} - a_k) \geq 0,$$

then $\lim_{n\to\infty} a_k = 0$.

Proposition 2.1. [15] *Suppose $g : \mathbb{R}^n \to \mathbb{R}$ is strongly convex with convexity parameter $\sigma > 0$ and continuously differentiable function such that ∇g is Lipschitz continuous with constant L_g. Then, the mapping $I - \sigma\nabla g$ is contraction for all $\sigma \leq \frac{2}{L_{h_1}+\rho}$, where I is the identity operator. That is $\|x - \sigma\nabla g(x) - (y - \sigma\nabla g(y))\| \leq \sqrt{1 - \frac{2\sigma\rho L_{h_1}}{\rho + L_{h_1}}}\|x - y\|$, for all $x, y \in \mathbb{R}^n$.*

3. Main results

In this section, by using Linesearch(3), we propose a new accelerated forward-backward algorithm based on the viscosity approximation method with an inertial technique for solving the convex bi-level optimization problem. We thoroughly analyze and establish a strong convergence result of our proposed algorithm.

We shift our attention to Problem (1.2) under the assumptions (AI) and (AII). To simplify notation, we denote $FB_\alpha := prox_{\alpha_k h_2}(I - \alpha_k\nabla h_1)$ for $\alpha > 0$ and let $\mathbf{h} := h_1 + h_2$. Denote that, Γ is the set of minimizers of $\mathbf{h}$. Also, we assume that $\Gamma \neq \emptyset$.

Algorithm 6 Linesearch(3)$(x_k, \sigma, \theta, \delta)$

(1) **Input** $x \in H$, $\sigma > 0$, $\delta > 0$, and $\theta \in (0,1)$.
(2) **Set** $\alpha = \sigma$.
(3) **While**

$$\frac{\alpha}{2}\{\|\nabla h_1(FB^2_\alpha(x)) - \nabla h_1(FB_\alpha(x))\| + \|\nabla h_1(FB_\alpha(x)) - \nabla h_1(x)\|\}$$
$$> \delta(\|FB^2_\alpha(x) - FB_\alpha(x)\| + \|FB_\alpha(x) - x\|),$$

do $\alpha = \theta\alpha$
(4) **End**
(5) **Output** α.

It is evident that the condition for the while loop in Linesearch(3) is weaker than that Linesearch(2). Consequently, based on the well-definedness of Linesearch(2), we can conclude that Linesearch(3) also stops after finitely many steps, see ([9], Lemma 3.2]).

Employing Linesearch(3), we introduce a novel accelerated forward-backward algorithm incorporating the inertial term, formulated as follows.

Algorithm 7

(1) **Initialization:** Take $x_1 = y_0 \in$ dom h_2 arbitrarily, $\sigma > 0$, $\delta \in (0, 1/8)$, $t_1 = 0$ and $\theta \in (0,1)$. Let $S : H \to H$ be a contraction with a coefficient $\eta \in (0,1)$.
Pick $\{\gamma_k\}, \{\tau_k\} \subset \mathbb{R}_{++}$, and let $\{\mu_k\} \subset \mathbb{R}_+$ be a bounded sequence. Let $S : H \to H$ be a contraction with coefficient $\eta \in (0,1)$.
(2) **For** $k \geq 1$, set

(3.7)
$$\beta_k = \begin{cases} \min\{\frac{t_k-1}{t_{k+1}}, \frac{\tau_k}{\|y_k - y_{k-1}\|}\}, & \text{if } y_k \neq y_{k-1}, \\ \frac{t_k-1}{t_{k+1}}, & \text{otherwise}, \end{cases}$$

where $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.
(3) **Step1.** Calculate the forward-backward step:

(3.8)
$$z_k = FB_{\alpha_k}(x_k) = prox_{\alpha_k h_2}(x_k - \alpha_k \nabla h_1(x_k)),$$

(3.9)
$$y_k = FB_{\alpha_k}(z_k) = prox_{\alpha_k h_2}(z_k - \alpha_k \nabla h_1(z_k)),$$

where $\alpha_k := Linesearch(3)(x_k, \sigma, \theta, \delta)$.
(4) **Step2.** Compute u_k and x_{k+1} using:

(3.10)
$$u_k = y_k + \beta_k(y_k - y_{k-1}),$$

(3.11)
$$x_{k+1} = (1-\gamma_k)u_k + \gamma_k S(y_k),$$

Then, update $k := k + 1$ and return to Step 1.

Lemma 3.4. *Let $\{x_k\}$ be a sequence generated by Algorithm 7 and $p \in H$ Then the following holds:*

$$\|x_k - p\|^2 - \|y_k - p\|^2 \geq 2\alpha_k[\mathbf{h}(y_k) + \mathbf{h}(z_k) - 2\mathbf{h}(p)]$$
$$+ (1 - 8\delta)(\|x_k - z_k\|^2 + \|z_k - y_k\|^2), \ \forall k \in \mathbb{N}.$$

Proof. From (2.6) and definition of z_k, y_k, we get

$$\frac{x_k - z_k}{\alpha_k} - \nabla h_1(x_k) \in \partial h_2(z_k) \text{ and } \frac{z_k - y_k}{\alpha_k} - \nabla h_1(z_k) \in \partial h_2(y_k).$$

Let $p \in H$. By definitions of subdifferential of h_2, the above expressions give

$$h_2(p) - h_2(z_k) \geq \langle \frac{x_k - z_k}{\alpha_k} - \nabla h_1(x_k), p - y_k \rangle$$

$$= \frac{1}{\alpha_k}\langle x_k - z_k, p - z_k\rangle + \langle \nabla h_1(x_k), z_k - p\rangle \tag{3.12}$$

and

$$h_2(p) - h_2(y_k) \geq \langle \frac{z_k - y_k}{\alpha_k} - \nabla h_1(z_k), p - z_k \rangle$$

$$= \frac{1}{\alpha_k}\langle z_k - y_k, p - y_k\rangle + \langle \nabla h_1(z_k), y_k - p\rangle \tag{3.13}$$

By (AI), we obtain the fact

$$h_1(x) - h_1(y) \geq \langle \nabla h_1(y), x - y\rangle, \ \forall x, y \in H. \tag{3.14}$$

From (3.14), we get

$$h_1(p) - h_1(x_k) \geq \langle \nabla h_1(x_k), p - x_k\rangle \tag{3.15}$$

and

$$h_1(p) - h_1(z_k) \geq \langle \nabla h_1(z_k), p - z_k\rangle \tag{3.16}$$

Combining (3.12), (3.13), (3.15) and (3.16), we have

$$\begin{aligned}
2\mathbf{h}(p) - \mathbf{h}(z_k) - h_2(y_k) - h_1(x_k) \geq\; & \langle \nabla h_1(x_k), z_k - p\rangle + \langle \nabla h_1(z_k), y_k - p\rangle + \langle \nabla h_1(x_k), p - x_k\rangle \\
& + \langle \nabla h_1(z_k), p - z_k\rangle + \frac{1}{\alpha_k}[\langle x_k - z_k, p - z_k\rangle + \langle z_k - y_k, p - y_k\rangle] \\
=\; & \langle \nabla h_1(x_k), z_k - x_k\rangle + \langle \nabla h_1(z_k), y_k - z_k\rangle \\
& + \frac{1}{\alpha_k}[\langle x_k - z_k, p - z_k\rangle + \langle z_k - y_k, p - y_k\rangle] \\
=\; & \langle \nabla h_1(x_k) - \nabla h_1(z_k), z_k - x_k\rangle + \langle \nabla h_1(z_k), z_k - x_k\rangle \\
& + \langle \nabla h_1(y_k), y_k - z_k\rangle + \langle \nabla h_1(z_k) - \nabla h_1(y_k), y_k - z_k\rangle \\
& + \frac{1}{\alpha_k}[\langle x_k - z_k, p - z_k\rangle + \langle z_k - y_k, p - y_k\rangle] \\
\geq\; & \langle \nabla h_1(z_k), z_k - x_k\rangle + \langle \nabla h_1(y_k), y_k - z_k\rangle \\
& - \|\nabla h_1(x_k) - \nabla h_1(z_k)\| \|z_k - x_k\| \\
& - \|\nabla h_1(z_k) - \nabla h_1(y_k)\| \|y_k - z_k\| \\
& + \frac{1}{\alpha_k}[\langle x_k - z_k, p - z_k\rangle + \langle z_k - y_k, p - y_k\rangle].
\end{aligned}$$

Again, applying (3.14), the above inequality becomes

$$\begin{aligned}2\mathbf{h}(p)-\mathbf{h}(z_k)-h_2(y_k)-h_1(x_k) &\geq h_1(y_k)-h_1(x_k)\\ &-\|\nabla h_1(x_k)-\nabla h_1(z_k)\|\|z_k-x_k\|\\ &-\|\nabla h_1(z_k)-\nabla h_1(y_k)\|\|y_k-z_k\|\\ &+\frac{1}{\alpha_k}[\langle x_k-z_k,p-z_k\rangle+\langle z_k-y_k,p-y_k\rangle]\\ &\geq h_1(y_k)-h_1(x_k)\\ &-\|\nabla h_1(x_k)-\nabla h_1(z_k)\|(\|y_k-z_k\|+\|z_k-x_k\|)\\ &-\|\nabla h_1(z_k)-\nabla h_1(y_k)\|(\|y_k-z_k\|+\|z_k-x_k\|)\\ &+\frac{1}{\alpha_k}[\langle x_k-z_k,p-z_k\rangle+\langle z_k-y_k,p-y_k\rangle]\\ &=h_1(y_k)-h_1(x_k)\\ &+\frac{1}{\alpha_k}[\langle x_k-z_k,p-z_k\rangle+\langle z_k-y_k,p-y_k\rangle]\\ &-(\|\nabla h_1(x_k)-\nabla h_1(z_k)\|+\\ &\|\nabla h_1(z_k)-\nabla h_1(y_k)\|)(\|y_k-z_k\|+\|z_k-x_k\|). \qquad (3.17)\end{aligned}$$

From $\alpha_k :=$ Linesearch$(3)(x_k,\sigma,\theta,\delta)$, we get

$$\frac{\alpha_k}{2}\{\|\nabla h_1(y_k)-\nabla h_1(z_k)\|+\|\nabla h_1(z_k)-\nabla h_1(x_k)\|\}\leq\delta(\|y_k-z_k\|+\|z_k-x_k\|). \qquad (3.18)$$

From (3.17) and (3.18), we have

$$\begin{aligned}\frac{1}{\alpha_k}[\langle x_k-z_k,z_k-p\rangle+\langle z_k-y_k,y_k-p\rangle] &\geq \mathbf{h}(y_k)+\mathbf{h}(z_k)-2\mathbf{h}(p)\\ &-(\|\nabla h_1(x_k)-\nabla h_1(z_k)\|+\\ &\|\nabla h_1(z_k)-\nabla h_1(y_k)\|)(\|y_k-z_k\|+\|z_k-x_k\|)\\ &\geq \mathbf{h}(y_k)+\mathbf{h}(z_k)-2\mathbf{h}(p)\\ &-\frac{2\delta}{\alpha_k}(\|y_k-z_k\|+\|z_k-x_k\|)^2\\ &\geq \mathbf{h}(y_k)+\mathbf{h}(z_k)-2\mathbf{h}(p)\\ &-\frac{4\delta}{\alpha_k}(\|y_k-z_k\|^2+\|z_k-x_k\|^2). \qquad (3.19)\end{aligned}$$

By Lemma 2.1(ii), we get

$$\langle x_k-z_k,z_k-p\rangle=\frac{1}{2}(\|x_k-p\|^2-\|x_k-z_k\|^2-\|z_k-p\|^2), \qquad (3.20)$$

and

$$\langle z_k-y_k,y_k-p\rangle=\frac{1}{2}(\|z_k-p\|^2-\|z_k-y_k\|^2-\|y_k-p\|^2). \qquad (3.21)$$

Hence, we can conclude from (3.19)-(3.21) that

$$\|x_k-p\|^2-\|y_k-p\|^2\geq 2\alpha_k[\mathbf{h}(y_k)+\mathbf{h}(z_k)-2\mathbf{h}(p)] \qquad (3.22)$$

$$+(1-8\delta)(\|x_k-z_k\|^2+\|z_k-y_k\|^2). \qquad (3.23)$$

□

Theorem 3.1. *Let $\{x_k\}\subset H$ be a sequence generated by Algorithm 7. Then:*

(i) *For* $p \in \Gamma$, *we have*

$$\|x_{k+1} - p\| \leq \max\left\{\|x_k - p\|, \frac{\frac{\beta_k}{\gamma_k}\|y_k - y_{k-1}\| + \|S(p) - p\|}{1-\eta}\right\}, \ \forall k \in \mathbb{N}. \tag{3.24}$$

(ii) *If the sequences* $\{\alpha_k\}$, $\{\gamma_k\}$ *and* $\{\tau_k\}$ *satisfy the following condition:*
 (Ci) $\sup_k \alpha_k \geq \alpha$ *for some* $\alpha \in \mathbb{R}_{++}$*;*
 (Cii) $\gamma_k \in (0,1)$ *such that* $\lim_{k\to\infty} \gamma_k = 0$ *and* $\sum_{k=1}^{\infty} \gamma_k = \infty$
 (Cii) $\lim_{n\to\infty} \tau_k/\gamma_k = 0$

Then $\{x_k\}$ *converges strongly to an element* $p^* \in \Gamma$, *where* $p^* = P_\Gamma S(p*)$.

Proof. Let $p \in \Gamma$. Applying Lemma 3.4, we have

$$\begin{aligned}\|x_k - p\|^2 - \|y_k - p\|^2 &\geq 2\alpha_k[\mathbf{h}(y_k) + \mathbf{h}(z_k) - 2\mathbf{h}(p)] \\ &\quad + (1-8\delta)(\|x_k - z_k\|^2 + \|z_k - y_k\|^2) \\ &\geq (1-8\delta)(\|x_k - z_k\|^2 + \|z_k - y_k\|^2) \qquad (3.25) \\ &\geq 0. \qquad (3.26)\end{aligned}$$

From (3.25), we get

$$\|y_k - p\|^2 \leq \|x_k - p\|^2 - (1-8\delta)(\|x_k - z_k\|^2 + \|z_k - y_k\|^2)$$

From above inequality, we get

$$\|y_k - p\| \leq \|x_k - p\| \tag{3.27}$$

By (3.10), we have

$$\begin{aligned}\|u_k - p\| &= \|y_k + \beta_k(y_k - y_{k-1}) - p\| \\ &\leq \|y_k - p\| + \beta_k\|y_k - y_{k-1}\| \qquad (3.28)\end{aligned}$$

From (3.11) and (3.27), we have

$$\begin{aligned}\|x_{k+1} - p\| &= \|(1-\gamma_k)u_k + \gamma_k S(y_k) - p\| \\ &\leq \gamma_k\|S(y_k) - S(p)\| + \gamma_k\|S(p) - p\| + (1-\gamma_k)\|u_k - p\| \\ &\leq \gamma_k\eta\|y_k - p\| + \gamma_k\|S(p) - p\| + (1-\gamma_k)\|u_k - p\| \\ &\leq (1-\gamma_k(1-\eta))\|y_k - p\| + \gamma_k(\frac{\beta_k}{\gamma_k}\|y_k - y_{k-1}\| + \|S(p) - p\|) \\ &\leq (1-\gamma_k(1-\eta))\|x_k - p\| + \gamma_k(\frac{\beta_k}{\gamma_k}\|y_k - y_{k-1}\| + \|S(p) - p\|) \\ &\leq \max\left\{\|x_k - p\|, \frac{\frac{\beta_k}{\gamma_k}\|y_k - y_{k-1}\| + \|S(p) - p\|}{1-\eta}\right\}.\end{aligned}$$

Therefore, (i) is obtained. By (3.7) and condition (Cii), we have $\frac{\beta_k}{\gamma_k}\|y_k - y_{k-1}\| \to 0$ as $k \to \infty$. There exist $M > 0$ such that $\frac{\beta_k}{\gamma_k}\|y_k - y_{k-1}\| < M$ for all $k \in \mathbb{N}$, and hence

$$\|x_{k+1} - p\| \leq \max\left\{\|x_k - p\|, \frac{M + \|S(p) - p\|}{1-\eta}\right\}.$$

By mathematical induction, we deduce that

$$\|x_k - p\| \leq \max\left\{\|x_1 - p\|, \frac{M + \|S(p) - p\|}{1-\eta}\right\} \ \forall k \in \mathbb{N}.$$

Hence, $\{x_k\}$ is bounded. One can see that the operator $P_\Gamma S$ is a contraction. By the Banach contraction principle, there is a unique point $p^* \in \Gamma$ such that $p^* = P_\Gamma S(p^*)$. It follows from the characterization of P_Γ that

$$\langle S(p^*) - p^*, p - p^*\rangle \leq 0, \ \forall p \in \Gamma. \tag{3.29}$$

By definition of $\{u_k\}$, we have

$$\begin{aligned} \|u_k - p^*\|^2 &\leq \|y_k + \beta_k(y_k - y_{k-1}) - p^*\|^2 \\ &= \|y_k - p^*\|^2 + 2\langle y_k - p^*, \beta_k(y_k - y_{k-1})\rangle + \beta_k^2\|y_k - y_{k-1}\|^2 \\ &\leq \|y_k - p^*\|^2 + 2\beta_k\|y_k - p^*\|\|y_k - y_{k-1}\| + \beta_k^2\|y_k - y_{k-1}\|^2. \end{aligned} \tag{3.30}$$

Using Lemma 2.1(i),(iii) and (3.25),we have

$$\begin{aligned} \|x_{k+1} - p^*\|^2 &= \|(1-\gamma_k)u_k + \gamma_k S(y_k) - p^*\|^2 \\ &= \|(1-\gamma_k)(u_k - p^*) + \gamma_k(S(y_k) - S(p^*)) + \gamma_k(S(p^*) - p^*)\|^2 \\ &\leq \|(1-\gamma_k)(u_k - p^*) + \gamma_k(S(y_k) - p^*)\|^2 + 2\gamma_k\langle S(p^*) - p^*, x_{k+1} - p^*\rangle \\ &\leq (1-\gamma_k)\|u_k - p^*\|^2 + \gamma_k\|S(y_k) - p^*\|^2 + 2\gamma_k\langle S(p^*) - p^*, x_{k+1} - p^*\rangle \\ &\leq (1-\gamma_k)[\|y_k - p^*\|^2 + 2\beta_k\|y_k - p^*\|\|y_k - y_{k-1}\| + \beta_k^2\|y_k - y_{k-1}\|^2] \\ &\quad + \gamma_k\eta\|y_k - p^*\|^2 + 2\gamma_k\langle S(p^*) - p^*, x_{k+1} - p^*\rangle + \gamma_k\|S(p^*) - p^*\| \\ &\leq (1-\gamma_k(1-\eta))\|y_k - p^*\|^2 + 2\beta_k\|y_k - p^*\|\|y_k - y_{k-1}\| + \beta_k^2\|y_k - y_{k-1}\|^2 \\ &\quad + 2\gamma_k\langle S(p^*) - p^*, x_{k+1} - p^*\rangle + \gamma_k\|S(p^*) - p^*\| \\ &\leq (1-\gamma_k(1-\eta))\|x_k - p^*\|^2 + 2\beta_k\|y_k - p^*\|\|y_k - y_{k-1}\| + \beta_k^2\|y_k - y_{k-1}\|^2 \\ &\quad + 2\gamma_k\langle S(p^*) - p^*, x_{k+1} - p^*\rangle + \gamma_k\|S(p^*) - p^*\| \\ &\quad - (1-\gamma_k(1-\eta))(1-8\delta)(\|x_k - z_k\|^2 + \|z_k - y_k\|^2) \\ &= (1-\gamma_k(1-\eta))\|x_k - p^*\|^2 + \gamma_k(1-\eta)b_k \\ &\quad - (1-\gamma_k(1-\eta))(1-8\delta)(\|x_k - z_k\|^2 + \|z_k - y_k\|^2), \end{aligned} \tag{3.31}$$

where

$$\begin{aligned} b_k :&= \frac{1}{1-\eta}\left(2\langle S(p^*) - p^*, x_{k+1} - p^*\rangle + 2\frac{\beta_k}{\gamma_k}\|y_k - p^*\|\|y_k - y_{k-1}\|\right) \\ &\quad + \frac{1}{1-\eta}\left(\frac{\beta_k^2}{\gamma_k}\|y_k - y_{k-1}\|^2 + \|S(p^*) - p^*\|\right). \end{aligned}$$

It follows that

$$\begin{aligned} (1-\gamma_k(1-\eta))(1-8\delta)(\|x_k - z_k\|^2 + \|z_k - y_k\|^2) \leq \|x_k - p^*\|^2 - \|x_{k+1} - p^*\|^2 \\ + \gamma_k(1-\eta)M^*, \end{aligned} \tag{3.32}$$

where $M^* = \sup\{b_k : k \in \mathbb{N}\}$.

Next, we show that $\{x_k\}$ converge to p^*. Set $a_k := \|x_k - p^*\|^2$ and $\zeta_k := \gamma_k(1-\eta)$. From (3.31), we have the following inequality:

$$a_{k+1} \leq (1-\zeta_k)a_k + \zeta_k b_k.$$

To apply Lemma 2.3, we have to show that $\limsup_{i\to\infty} b_{k_i} \leq 0$ whenever a subsequence $\{a_{k_i}\}$ of $\{a_k\}$ satisfies

$$\liminf_{i\to\infty}(a_{k_i+1} - a_{k_i}) \geq 0 \tag{3.33}$$

To do this, suppose that $\{a_{k_i}\} \subseteq \{a_k\}$ is a subsequence satisfying (3.33). Then, by (3.32) and (Cii), we have

$$\begin{aligned}\limsup_{i\to\infty}(1-\gamma_{k_i}(1-\eta))(1-8\delta)(\|x_{k_i}-z_{k_i}\|^2+\|z_{k_i}-y_{k_i}\|^2) &\le \limsup_{i\to\infty}(a_{k_i}-a_{k_i+1}) \\ &\quad + (1-\eta)M^*\lim_{i\to\infty}\gamma_{k_i} \\ &= -\liminf_{i\to\infty}(a_{k_i+1}-a_{k_i}) \\ &\le 0,\end{aligned}$$

which implies

$$\lim_{i\to\infty}\|x_{k_i}-z_{k_i}\| = \lim_{i\to\infty}\|z_{k_i}-y_{k_i}\| = 0. \tag{3.34}$$

Using (Cii), (Ciii) and (3.34), we have

$$\begin{aligned}\|x_{k_i+1}-x_{k_i}\| &= \|(1-\gamma_{k_i})u_{k_i}+\gamma_{k_i}S(y_{k_i})-x_{k_i}\| \\ &\le \gamma_{k_i}\|S(y_{k_i})-x_{k_i}\| + \|u_{k_i}-x_{k_i}\| \\ &\le \gamma_{k_i}\|S(y_{k_i})-x_{k_i}\| + \|u_{k_i}-z_{k_i}\| + \|z_{k_i}-x_{k_i}\| \\ &\le \gamma_{k_i}\|S(y_{k_i})-x_{k_i}\| + \|y_{k_i}-z_{k_i}\| + \frac{\beta_{k_i}}{\gamma_{k_i}}\|y_{k_i}-y_{k_i-1}\| + \|z_{k_i}-x_{k_i}\| \\ &\to 0, \text{ as } i\to\infty. \end{aligned} \tag{3.35}$$

We next show that $\limsup_{i\to\infty} b_{k_i} \le 0$. Clearly, it suffices to show that

$$\limsup_{i\to\infty}\langle S(p^*)-p^*, x_{k_i+1}-p^*\rangle \le 0.$$

Since $\{x_{k_i}\}$ is bounded, we can choose a subsequence of $\{x_{k_i}\}$ such that

$$\lim_{j\to\infty}\langle S(p^*)-p^*, x_{k_{i_j}}-p^*\rangle = \limsup_{i\to\infty}\langle S(p^*)-p^*, x_{k_i}-p^*\rangle. \tag{3.36}$$

and $x_{k_{i_j}} \rightharpoonup \bar{p}$ as $j\to\infty$ for some $\bar{p}\in H$. Thus, we also have $z_{k_{i_j}} \rightharpoonup \bar{p}$ as $j\to\infty$. From (AI), we have $\|\nabla h_1(x_{k_{i_j}}) - \nabla h_1(z_{k_{i_j}})\| \to 0$ as $j\to\infty$. This together with (3.34) and (Ci) yields

$$\lim_{j\to\infty}\left\|\frac{x_{k_{i_j}}-z_{k_{i_j}}}{\alpha_{k_{i_j}}} + \nabla h_1(z_{k_{i_j}}) - \nabla h_1(x_{k_{i_j}})\right\| = 0. \tag{3.37}$$

By (2.6), we get

$$\frac{x_{k_{i_j}}-z_{k_{i_j}}}{\alpha_{k_{i_j}}} + \nabla h_1(z_{k_{i_j}}) - \nabla h_1(x_{k_{i_j}}) \in \partial h_2(z_{k_{i_j}}) + \nabla h_1(z_{k_{i_j}}) = \partial \mathbf{h}(z_{k_{i_j}}). \tag{3.38}$$

Now, by (3.37), (3.38) and $z_{k_{i_j}} \rightharpoonup \bar{p}$, it follows from Lemma 2.2 that $0\in\partial\mathbf{h}(\bar{p})$. Hence, $\bar{p}\in\Gamma$. From (3.35) and (3.38), we have

$$\begin{aligned}\limsup_{i\to\infty}\langle S(p^*)-p^*, x_{k_i+1}-p^*\rangle &\le \limsup_{i\to\infty}\langle S(p^*)-p^*, x_{k_i+1}-x_{k_i}\rangle \\ &\quad + \limsup_{i\to\infty}\langle S(p^*)-p^*, x_{k_i}-p^*\rangle \\ &= \limsup_{j\to\infty}\langle S(p^*)-p^*, x_{k_{i_j}}-p^*\rangle \\ &= \limsup_{j\to\infty}\langle S(p^*)-p^*, \bar{p}-p^*\rangle \\ &\le 0.\end{aligned}$$

By Lemma 2.3, we conclude that $\{x_k\}$ converges to p^*. The proof is now complete. □

Now, we employ Algorithm 7 for solving Problem (1.1). We obtain the following result as a consequence of Theorem 3.1.

Theorem 3.2. *Let $g : \mathbb{R}^n \to \mathbb{R}$ be a strongly convex function with a parameter $\rho > 0$. Assume that g is continuously differentiable and its gradient ∇g is Lipschitz continuous with a constant L_g. Suppose that h_1 and h_2 satisfy the assumptions of Problem (1.2). Let $\{x_k\}$ be a sequence generated by Algorithm 7. Then $\{x_k\}$ converges strongly to $\bar{p} \in \mathcal{A}$ where $\mathcal{A}$ is the set of all solutions of Problem (1.1).*

Proof. By Theorem 3.1, we get that $\{x_k\}$ converges to $\bar{p} \in \Gamma = X^* = arg\min_{x\in\mathbb{R}^n}(h_1(x) + h_2(x))$ and $\bar{p} = P_{x^*}S(p^*)$. By Proposition 2.1, $S = I - s\nabla g(x)$ is a k-contraction with parameter $k = \sqrt{1 - \frac{2s\rho L_g}{\rho + L_g}}$, whenever $s \in (0, 2/(\rho + L_g))$. It remains to show that $\bar{p} = arcmin_{x\in X^*} g(x)$. By using $\bar{p} = P_{X^*}S(\bar{p})$ and (2.5), we have, for $z \in X^*$,

$$\begin{aligned}
\bar{p} = P_\Gamma S(\bar{p}) &\Leftrightarrow \langle S(\bar{p}) - \bar{p}, z - \bar{p}\rangle \leq 0 \\
&\Leftrightarrow \langle \bar{p} - s\nabla g(\bar{p}) - \bar{p}, z - \bar{p}\rangle \leq 0 \\
&\Leftrightarrow \langle s\nabla g(\bar{p}), z - \bar{p}\rangle \geq 0 \\
&\Leftrightarrow s\langle \nabla g(\bar{p}), z - \bar{p}\rangle \geq 0 \\
&\Leftrightarrow \langle \nabla g(\bar{p}), z - \bar{p}\rangle \geq 0.
\end{aligned}$$

Thus, $\bar{p}$ is an optimal solution for the Problem (1.1). That is, $x_k \to \bar{p} \in \mathcal{A}$. □

4. Applications

In this section, we utilize Algorithm 7 as a machine learning algorithm applying for a Single Hidden Layer Feedforward Neural Networks for classifying data, leveraging a model of SLFNs (Single Hidden Layer Feedforward Neural Networks) and Extrem Learning Machine. The experiments are performed using the MATLAB computing environment on an Intel Core-i5 8th with 8 GB RAM.

We begin by revisiting fundamental concepts of Extreme Learning Machine (ELM) concerning data classification problems. Subsequently, we propose our algorithm for addressing these problems and conduct a comparative performance evaluation involving another algorithm.

Extreme Learning Machine (ELM) [7] is defined as follows: Let $D = \{(x_d, q_d) : x_d \in \mathbb{R}^n, q_d \in \mathbb{R}^m, d = 1, 2, ..., N\}$ represent a training set comprising N distinct samples, where x_d is an input data and q_d is a target. A standard Single Hidden Layer Feedforward Neural Networks (SLFN) with M hidden nodes and activation function $\varphi(x)$ is given by:

$$\sum_{j=1}^{M} \xi_j \varphi(\langle p_j, x_d\rangle + c_j) = o_d,\ d = 1, ..., N,$$

where ξ_j denotes the weight vector connecting the j-th hidden node to the output node, p_j represents the weight vector connecting the j-th hidden node to the input node, and c_j is the bias term. The objective of SLFNs is to predict the N outputs in a manner that minimizes the error, expressed as $\sum_{d=1}^{N} |o_d - q_d| = 0$. That is,

$$\sum_{j=1}^{M} \xi_j \varphi(\langle p_j, x_d\rangle + c_j) = q_d, d = 1, ..., N. \tag{4.39}$$

We can rewrite above system of linear equation by the following matrix equation:

$$R\xi = Q, \tag{4.40}$$

where

$$R = \begin{bmatrix} \varphi(\langle p_1, x_1\rangle + c_1) & \cdots & \varphi(\langle p_M, x_1\rangle + c_M) \\ \vdots & \ddots & \vdots \\ \varphi(\langle p_1, x_N\rangle + c_1) & \cdots & \varphi(\langle p_M, x_N\rangle + c_M) \end{bmatrix}_{N\times M}$$

$$\xi = [\xi_1^T, ..., \xi_M^T]^T_{m\times M},\ Q = [q_1^T, ..., q_N^T]^T_{m\times N}.$$

The objective of a SLFNs is estimating ξ_j, p_j and c_j for solving (4.39) while ELM aims to find only ξ_j with randomly p_j and c_j.

The Problem (4.40) can be considered as the following convex minimization problem:

$$\min_{\xi} \|R\xi - Q\|_2^2 + \lambda \|\xi\|_1,$$

where $\lambda > 0$ is called regularization parameter. In Algorithm 7, we set $h_1(\xi) = \|R\xi - Q\|_2^2$ and $h_2(\xi) = \lambda \|\xi\|_1$. We employ Algorithm 7 to solve a convex bi-level optimization Problem (1.1) and (1.2) while the outer level function is give by $g(\xi) = \frac{1}{2}\|\xi\|_2^2$.

We implement our proposed Algorithm 7 for data classification and conduct a performance comparison with other methods. Our experimental setup incorporates five datasets obtained from "https://archive.ics.uci.edu/, accessed on 7 January 2023" and "https://www.kaggle.com/, accessed on 7 January 2023" as follows:

Breast Cancer dataset [24]: This dataset consists of 11 attributes, and its classification involves distinguishing data into 2 distinct classes.

Heart Disease UCI dataset [5]: With 14 attributes, this dataset also focuses on classifying data into 2 distinct classes.

Diabetes dataset [18]: Comprising 9 attributes, this dataset involves the classification of data into two distinct classes.

Parkinsons dataset [11]: With 23 attributes, this dataset involves categorizing information into two distinct classes.

We set all control conditions for each algorithms as in Table 1.

TABLE 1. Algorithms and their setting control conditions.

Methods	Setting
Algorithm 7	$s = 0.01$, $\sigma = 2$, $\delta = 0.1$, $\theta = 0.1$, $\eta = 0.99$, $t_1 = 0$, $\gamma_k = \frac{1}{60k}$, $\tau_k = \frac{10^{60}}{k^2}$
BiG-SAM	$\alpha = 3$, $s = 0.01$, $c_k = \frac{1}{L_{h_1}}$, $\alpha_k = \frac{2(0.1)}{1-\frac{2+c_k L_{h_1}}{4}}$, $\gamma_k = \frac{\alpha_k}{n^{0.01}}$
iBiG-SAM	$\alpha = 3$, $s = 0.01$, $c_k = \frac{1}{L_{h_1}}$, $\alpha_k = \frac{2(0.1)}{1-\frac{2+c_k L_{h_1}}{4}}$, $\gamma_k = \frac{\alpha_k}{n^{0.01}}$
aiBiG-SAM	$\alpha = 3$, $s = 0.01$, $c_k = \frac{1}{L_{h_1}}$, $\alpha_k = \frac{1}{k+2}$, $\gamma_k = \frac{\alpha_k}{n^{0.01}}$
Algorithm 4	$\sigma = 2$, $\delta = 0.1$, $\theta = 0.9$
Algorithm 5	$\sigma = 2$, $\delta = 0.1$, $\theta = 0.9$

In Table 2, we give the details of attributes for each dataset and the corresponding data distribution into training and testing sets. The training set encompasses approximately 70% of the data, while the remaining 30% is designated for the testing set.

We have done our experiments by using the set of control parameters as detailed in Table 1, the number of hidden nodes $M = 100$ and a sigmoid function as an activation function. For every dataset listed in Table 2, we conduct training on the respective training

TABLE 2. Training and Testing sets of dataset.

Dataset	Attributes	Sample Train	Sample Test
Breast Cancer	11	488	211
Heart Disease	14	213	90
Diabetes	9	538	230
Parkinson	23	135	60

set. The accuracy, precision, recall, and F1-score of the output data were computed using the following formulas, respectively.

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

$$\text{Precision(Pre)} = \frac{\text{TP}}{\text{TP+FP}},$$

$$\text{Recall(Rec)} = \frac{\text{TP}}{\text{TP+FP}},$$

$$\text{F1-score(F1)} = \frac{\text{TP}}{\text{TP+1/2(FP+FN)}},$$

where TP is the number of samples correctly predicted as positive, TN denotes the number of samples correctly predicted as negative, FN represents the number of samples wrongly predicted as negative. and FP means number of samples wrongly predicted as positive.

In Table 3 and Table 4, we provide a comparative analysis showcasing the training accuracy, testing accuracy, training precision, testing precision, training recall, testing recall, training F1-score, testing F1-score and iteration number of Algorithm 7 with other algorithms for each dataset.

TABLE 3. The iteration number of each algorithm with the best accuracy on each dataset.

Dataset	Algorithm	Iteration no.	Accuracy train	Accuracy test
Breast Cancer	Algorithm 7	986	96.55	99.50
	BIG-SAM	1700	96.55	98.49
	iBIG-SAM	1700	96.55	98.49
	aiBIG-SAM	1700	96.55	98.49
	Algorithm 4	100	96.55	98.99
	Algorithm 5	52	96.55	98.99
Heart Disease	Algorithm 7	98	86.67	83.87
	BIG-SAM	1800	86.19	82.80
	iBIG-SAM	1756	86.19	82.80
	aiBIG-SAM	2501	86.67	82.80
	Algorithm 4	630	86.19	82.80
	Algorithm 5	567	86.19	83.87
Diabetes	Algorithm 7	96	77.11	81.98
	BIG-SAM	700	76.01	81.08
	iBIG-SAM	696	74.54	81.08
	aiBIG-SAM	1300	76.92	80.18
	Algorithm 4	898	74.36	80.18
	Algorithm 5	582	75.46	81.98
Parkinson	Algorithm 7	454	94.16	81.03
	BIG-SAM	659	86.13	77.59
	iBIG-SAM	660	86.13	77.59
	aiBIG-SAM	2240	87.59	77.59
	Algorithm 4	540	87.59	77.59
	Algorithm 5	391	88.32	77.59

TABLE 4. The precision, recall, and F1-score of each algorithm with the best accuracy on each dataset.

Dataset	Algorithm	Pre train	Pre test	Rec train	Rec test	F1 train	F1 test	Acc test
Breast Cancer	Algorithm 7	0.9810	0.9928	0.9688	1	0.9748	0.9964	99.50
	BIG-SAM	0.9810	0.9787	0.9688	1	0.9748	0.9892	98.49
	iBIG-SAM	0.9810	0.9787	0.9688	1	0.9748	0.9892	98.49
	aiBIG-SAM	0.9810	0.9787	0.9688	1	0.9748	0.9892	98.49
	Algorithm 18	0.9810	0.9857	0.9688	1	0.9748	0.9928	98.99
	Algorithm 19	0.9810	0.9857	0.9688	1	0.9748	0.9928	98.99
Heart Disease	Algorithm 7	0.8425	0.7966	0.9304	0.9400	0.8843	0.8624	83.87
	BIG-SAM	0.8413	0.7833	0.9217	0.9400	0.8797	0.8545	82.80
	iBIG-SAM	0.8413	0.7833	0.9217	0.9400	0.8797	0.8545	82.80
	aiBIG-SAM	0.8425	0.7833	0.9304	0.9400	0.8843	0.8545	82.80
	Algorithm 18	0.8468	0.7931	0.9130	0.9200	0.8787	0.8519	82.80
	Algorithm 19	0.8413	0.7966	0.9217	0.9400	0.8797	0.8624	83.87
Diabetes	Algorithm 7	0.6767	0.7500	0.5641	0.6667	0.6128	0.7059	81.98
	BIG-SAM	0.7305	0.7586	0.5282	0.6111	0.6131	0.6769	81.08
	iBIG-SAM	0.7324	0.5333	0.7586	0.6111	0.6172	0.6769	81.08
	aiBIG-SAM	0.7447	0.5385	0.7500	0.5833	0.6250	0.6562	80.18
	Algorithm 18	0.7037	0.7500	0.4872	0.5833	0.5758	0.6562	80.18
	Algorithm 19	0.7259	0.5026	0.7857	0.6111	0.5939	0.6875	81.98
Parkinson	Algorithm 7	0.9439	0.9783	0.9806	0.8182	0.9619	0.8911	81.03
	BIG-SAM	0.8684	0.9612	0.9375	0.8182	0.9124	0.8738	77.59
	iBIG-SAM	0.8684	0.9612	0.9375	0.8182	0.9124	0.8738	77.59
	aiBIG-SAM	0.8839	0.9612	0.9375	0.8182	0.9209	0.8738	77.59
	Algorithm 18	0.8839	0.9612	0.9375	0.8182	0.9217	0.8738	77.59
	Algorithm 19	0.8850	0.9709	0.9375	0.8182	0.9259	0.8738	77.59

From the observations in Table 3 and Table 4, it is evident that Algorithm 7 exhibits But Table 3 shows that Algorithm 7 needs way more number of iterations than Algorithm 5 for Breast Cancer. across all conducted experiments. Furthermore, Algorithm 7 requires the lowest number of iterations to the highest comparable accuracy compared to the other studied algorithms.

5. CONCLUSIONS

In this work, we study and discuss the convex bi-level optimization problem. The challenge of removing the Lipschitz continuity assumption on the gradient of the objective function attracts us to study the concept of the linesearch method. We use linesearch from [19] and introduce an accelerated forward-backward algorithm with an inertial technique whose stepsize does not depend on any Lipschitz constant for solving the considered problem without any Lipschitz continuity condition on the gradient. We prove that the sequence generated by our proposed method converges strongly to an optimal solution of the convex bi-level optimization problems under some mild control conditions. As applications, we apply our method to solving data classification of non-communicable diseases. The comparative experiments show that our algorithm has a higher efficiency than the others.

Acknowledgements. This research has been financially supported by Fundamental Fund 2024, Chiang Mai University (FF030/2567). The first author expresses gratitude to the Science Achievement Scholarship of Thailand (SAST) for their generous financial assistance.

REFERENCES

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

[2] Beck, A.; Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, **2** (2009), no. 1, 183–202.

[3] Bello Cruz, J. Y.; Nghia, T.-T. On the convergence of the forward–backward splitting method with linesearches. *Optimization Methods and Software* **31** (2016), no. 6, 1209–1238.

[4] Boyd, S.; Vandenberghe, L. *Convex Optimization. Cambridge University Pass*. New York, USA, 2004.

[5] Detrano, R.; Janosi, A.; Steinbrunn, W.; Pfisterer, M.; Schmid, J.-J.; Sandhu, S.; Guppy, K.-H.; Lee, S.; Froelicher, V. International application of a new probability algorithm for the diagnosis of coronary artery disease. *Am. J. Cardiol.* **64** (1989), no. 5, 304–310.

[6] Duan, P.; Zhang, Y. Alternated and multi-step inertial approximation methods for solving convex bilevel optimization problems. *Optimization* **72** (2022), no. 10, 2517–2545.

[7] Huang, G.-B.; Zhu, Q.Y.; Siew, C.-K. Extreme learning machine: Theory and applications. *Neurocomputing.* **70** (2006), no. (1–3), 489–501.

[8] Jailoka, P.; Suantai, S.; Hanjing, A. A fast viscosity forward-backward algorithm for convex minimization problems with an application in image recovery. *Carpathian J. Math.*, **37** (2021), no. 3, 449–461.

[9] Kankam, K.; Pholasa, N.; Cholamjiak, P. On convergence and complexity of the modified forward-backward method involving new linesearches for convex minimization. *Mathematical Methods in the Applied Sciences* **42** (2019), no. 5, 1352–1362.

[10] Lions, P.-L.; Mercier, B. Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal.* **16** (1979), no. 6, 964–979.

[11] Little, M.-A.; McSharry, P.-E.; Roberts, S.-J.; Costello, D.-A.; Moroz., I- M. Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *Nature Precedings.* (2007), 1–1.

[12] Polyak, B. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, **4** (1964), no. 5, 1–17.

[13] Puangpee, J.; Suantai, S. A New Accelerated Viscosity Iterative Method for an Infinite Family of Nonexpansive Mappings with Applications to Image Restoration Problems. *Mathematics*, **8** (2020), no. 4, 615.

[14] Rockafellar, R. On the maximal monotonicity of subdifferential mappings. *Pacific J. Math.* **33** (1970), no. 1, 209–216.

[15] Sabach, S.; Shtern, S. A first order method for solving convex bilevel optimization problems. *SIAM J. Optim.* **27** (2017), no. 2, 640–660.

[16] Saejung, S.; Yotkaew, P. Approximation of zeros of inverse strongly monotone operators in Banach spaces. *Nonlinear Anal.* **75** (2012), no. 2, 724–750.

[17] Shehu, Y.; Vuong, P.-T.; Zemkoho, A. An inertial extrapolation method for convex simple bilevel optimization. *Optim Methods Softw.* **36** (2019), no. 1, 1–19.

[18] Smith, J.-W.; Everhart, J.-E.; Dickson, W.-C.; Knowler, W.-C.; Johannes, R.-S.; Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. *Proc. Symp. Comput. Appl. Med. Care.* 1998.

[19] 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.

[20] Takahashi, W. *Introduction to Nonlinear and Convex Analysis Yokohama Publishers*. Yokohama, Japan, 2009.

[21] Takahashi, W. *Nonlinear Functional Analysis*, Yokohama Publishers Yokohama, Japan, 2000.

[22] Thongsri, P.; Panyanak, B.; Suantai, S. A New Accelerated Algorithm Based on Fixed Point Method for Convex Bilevel Optimization Problems with Applications. *Mathematics* **11** (2023), no. 3, 702.

[23] Tseng, P. A modified forward-backward splitting method for maximal monotone mappings. *SIAM Journal on Control and Optimization* **38** (2000), no. 2, 431–446.

[24] Wolberg, W.-H.; Mangasarian, O.-L. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proc. Natl. Acad. Sci. USA.* **87** (1990), no. 23, 9193–9196.

[25] Xu, H.-K. Viscosity approximation methods for nonexpansive mappings. *J. Math. Anal. Appl.* **298** (2004), no. 1, 279–291.

Department of Mathematics, Faculty of Science
Chiang Mai University
239 Huaykaew Rd, 50200, Chiang Mai, Thailand
Email address: suthep.s@cmu.ac.th

PhD Degree Program in Mathematics
Department of Mathematics, Faculty of Science
Chiang Mai University
239 Huaykaew Rd, 50200, Chiang Mai, Thailand
Email address: piti_tho@cmu.ac.th