EFFICIENT OPTIMIZATION ALGORITHMS FOR VARIATIONAL INCLUSION PROBLEMS AND APPLICATIONS

PRONPAT PEEYADA

A Dissertation Submitted to University of Phayao

in Partial Fulfillment of the Requirements

for the Doctor of Philosophy in Mathematics

November 2023

Thesis

Title

Efficient Optimization Algorithms for Variational Inclusion Problems and Applications

Submitted by Pronpat Peeyada

Approved in partial fulfillment of the requirements for the

Doctor of Philosophy in Mathematics

University of Phayao

.....Chairman

(Professor Dr.Suthep Suantai)

.....Committee

(Professor Dr.Rabian Wangkeeree)

.....Committee

(Associate Professor Dr.Watcharaporn Cholamjiak)

.....Committee

(Associate Professor Dr.Tanakit Thianwan)

.....Committee

(Associate Professor Dr.Prasit Cholamjiak)

.....Committee

(Associate Professor Dr.Damrongsak Yambangwai)

Approved by

.....

(Associate Professor Dr.Sitthisak Pinmongkhonkul)

Dean of School of Science

November 2023

ACKNOWLEDGEMENT

I would like to express my sincere thanks to my thesis advisors, Associate Professor Dr.Watcharaporn Cholamjiak and Associate Professor Dr.Prasit Cholamjiak for the continuous support of my Ph.D. study and related research, for their motivation and immense knowledge. I could not have imagined having a better advisors and mentors for my Ph.D. study.

I would like to take this opportunity to express my thankfulness to Assistant Professor Dr.Yann Savoye when I was doing research at Liverpool John Moores University, Liverpool, UK and my gratefulness to Professor Dr.Choonkil Park when I was doing research at Hanyang University, Seoul, South Korea for his kindness, helpful guidance and academic support.

My sincere gratitude also goes to my thesis committees, Professor Dr.Suthep Suantai, Professor Dr.Rabian Wangkeeree, Associate Professor Dr.Tanakit Thianwan, and Associate Professor Dr.Damrongsak Yambangwai. It is my pleasure to express my thanks to all of them for their generous.

I appreciate mentioning that my graduate study was financially supported by International Research Network in Digital Image Processing and Machine Learning (IRN).

Finally, I most gratefully acknowledge my family, friends and brothers and sisters for their support throughout the period of this thesis.

Pronpat Peeyada

เรื่อง: อัลกอริทึมการหาค่าเหมาะที่สุดที่มีประสิทธิภาพสำหรับปัญหาการรวมแปรผันและการประยุกต์

ผู้วิจัย: ภรภัทร ปียะดา, วิทยานิพนธ์: ปร.ด. (คณิตศาสตร์), มหาวิทยาลัยพะเยา 2566

ประธานที่ปรึกษา: รองศาสตราจารย์ ดร.วัชรภรณ์ ช่อลำเจียก

กรรมการที่ปรึกษา: รองศาสตราจารย์ ดร.ธนกฤต เทียนหวาน, รองศาสตราจารย์ ดร.ประสิทธิ์ ช่อลำเจียก

คำสำคัญ: ปัญหาการรวมแปรผัน, ปริภูมิฮิลเบิร์ต, การจำแนกประเภทข้อมูล, การประมวลผลสัญญาณ, การประมวลผลภาพ

บทคัดย่อ

อัลกอริทึมการหาค่าเหมาะที่สุดช่วยแก้ปัญหามากมายในด้านวิศวกรรม เศรษฐศาสตร์ วิทยาการคอมพิวเตอร์ การแพทย์และกลศาสตร์ ในทางคณิตศาสตร์ปัญหาสำคัญประการหนึ่งในการหาค่าเหมาะที่สุดก็คือปัญหาการรวมแบบแปรผัน ซึ่งสามารถจำลองเป็นปัญหาเชิงปฏิบัติที่มีอยู่จริงมากมาย เช่น การประมวลผลสัญญาณ การประมวลผลภาพ การจัดจำแนกข้อมูล และการประยุกต์ในสาขาอื่นๆ จุดมุ่งหมายของเราในวิทยานิพนธ์นี้คือเพื่อสร้างอัลกอริทึมใหม่แบบเร่งสี่อัลกอริทึมสำหรับการแก้ปัญหาการรวมแปรผัน มีดังนี้ (1) อัลกอริทึมแบบแยกข้างหน้า ข้างหลังของมานน์แบบเฉื่อย ที่ถูกนำไปประยุกต์เพื่อแก้ปัญหาการจำแนกโรคมะเร็งเต้านมและการประมวลผลสัญญาณ และสามารถแก้ปัญหาที่แตกต่างกันได้โดยการเลือกใช้ขนาดขั้นตอนที่เหมาะสม (2) อัลกอริทึมแบบแยกข้างหน้า ข้างหลังแบบเฉื่อยที่ถูกนำไปประยุกต์แก้ปัญหาการจำแนกโรคมะเร็งเต้านม (3) อัลกอริทึมภาพฉายแบบใหม่ที่ถูกนำไปประยุกต์แก้ปัญหาการจำแนกความเสี่ยงของโรคมะเร็งปากมดลูก โดยการพิจารณาแบบจำลองการทำให้เป็นมาตรฐานหกรูปแบบเพื่อหลีกเลี่ยงการเรียนรู้ของเครื่องที่ไม่เหมาะสม (4) อัลกอริทึมเส้นเกรเดียนต์ย่อยแบบเฉื่อยคู่ขนานแบบไฮบริดสำหรับการแก้ปัญหาอสมการแปรผันที่ถูกลดรูปมาจากปัญหาการรวมแปรผัน ที่ถูกนำไปประยุกต์ใช้ในการแก้ปัญหาการประมวลผลภาพในกรณีที่ภาพตั้งต้นถูกเบลอโดยตัวดำเนินการเบลอมากมายเป็นจำนวนจำกัด ทฤษฎีบทการลู่เข้าของอัลกอริทึมที่เสนอทั้งหมดได้ถูกพิสูจน์ภายใต้เงื่อนไขที่เหมาะสมในปริภูมิฮิลเบิร์ต

Title: EFFICIENT OPTIMIZATION ALGORITHMS FOR VARIATIONAL INCLUSION PROBLEMS AND APPLICATIONS

Author: Pronpat Peeyada, Thesis: Ph.D. (Mathematics), University of Phayao, 2023

Advisor: Associate Professor Dr.Watcharaporn Cholamjiak

Co-advisor: Associate Professor Dr.Tanakit Thianwan, Associate Professor Dr.Prasit Cholamjiak

Keywords: variational inclusion problem, Hilbert space, data classification, signal recovery, image recovery

ABSTRACT

Optimization algorithms can solve many problems in engineering, economics, computer science, medicine, and mechanics. In mathematics, one of the critical problems in optimization is the variation inclusion problem. It can be modelled for many practical problems, such as signal processing, image processing, data classification, and other applied fields. Our aim in this thesis is to construct four new accelerated algorithms for solving variational inclusion problems as follow: (i) inertial Mann forward–backward splitting algorithm, it can be applied to solve breast cancer classification and signal recovery, and solved different problems by choosing suitable different step sized; (ii) modified inertial forward–backward splitting algorithm which is applied to solve breast cancer classification; (iii) new projection algorithm which is applied to solve cervical cancer behavior risk classification by considering in six regularization models for avoiding overfitting; (iv) hybrid inertial parallel subgradient extragradient–line algorithm for solving variational inequality which is reduced from variational inclusion problems, it can be applied to solve image recovery in the case of an unknown image is blurred by finitely blur operators. Under some suitable conditions in Hilbert spaces, the convergence theorems of the proposed algorithms are proved.

LIST OF CONTENTS

Chapter **Page**

LIST OF TABLES

Tables **Page**

LIST OF TABLES

Tables **Page**

LIST OF FIGURES

Figures **Page**

LIST OF FIGURES

Figures **Page**

CHAPTER 1

INTRODUCTION

Optimization has resulted in new algorithms and theories being developed that can be applied to solve real-world problems in both pure and applied science. Important problems exist in engineering, economics, computer science, physics, and mechanics can be formulated in mathematical optimization. One of critical problems in optimization is variation inclusion problem, which plays a central role in nonlinear analysis around known mathematical models such as minimization problems, split feasibility problems, convex programming, and variational inequality, etc., with applications in signal recovery, image processing, machine learning and others. In this work, we study the variational inclusion problem, which has received much attention from many authors whose works on theoretical results as well as iterative algorithms.

In 1970, Martinet [56] introduced the popular iteration method known as the proximal point algorithm. In 1976, Lions and Mercier [52] proposed the Peaceman-Rachford and Douglas-Rachford splitting algorithms for problems involving the sum of two monotone operators. These algorithms are well known in the linear case and are here extended to the case of multivalued monotone operators. In 2000, Tseng [81] proposed the forward-backward splitting method for finding a zero of the sum of two maximal monotone mappings. This method is known to converge when the inverse of the forward mapping is strongly monotone. One way to study the convergence of algorithms is finding a method that makes an algorithm converges faster. Polyak [64] introduced the so-called heavy ball method, a two step iterative method for minimizing a smooth convex function. In 2001, Alvarez and Attouch [3] translated the idea of the heavy ball method to the setting of a general maximal monotone operator using the framework of the

proximal point algorithm. In 2003, Moudafi and Oliny [58] introduced a forward-backward inertial procedure for solving the problem of finding a zero of the sum of two maximal monotone operators is proposed and its convergence is established under a cocoercivity condition with respect to the solution set. In 2015, Lorenz and Pock [54] proposed an inertial forward-backward splitting algorithm to compute a zero of the sum of two monotone operators, with one of the two operators being co-coercive. In 2018, Gibali and Thong [37] introduced two modifications of the forward-backward splitting method with a new step size rule for inclusion problems in real Hilbert spaces. The modifications are based on Mann and viscosity-ideas. Under standard assumptions, such as Lipschitz continuity and monotonicity, establish strong convergence of the proposed algorithms.

The aim of this thesis is to design new efficient optimization algorithms for solving variational inclusion problems. We prove the convergence theorems under some suitable conditions in Hilbert spaces. Finally, we apply the proposed algorithm to solve data classification, signal recovery, and image recovery.

CHAPTER 2

REVIEW OF RELATED LITERATURE AND RESEARCH

Let H be a real Hilbert space. In this research, the objective of our investigation is to solve the following variational inclusion problem: find an element $x^* \in H$ such that

$$0 \in (F + G)x^*, \tag{2.1.1}$$

where $G : H \to 2^H$ is a multi-valued maximal monotone mapping and $F : H \to H$ is a monotone and Lipschitz continuous mapping. The solution set of variational inclusion problems is denoted by $(F+G)^{-1}(0)$. Approximating a solution of variational inclusion problem (2.1.1) has a variety of specific applications since many problems can be seen as variational inclusion problem (2.1.1), for example, as variational inequality problems, split feasibility problems, and convex minimization problems with applications in signal and image processing, machine learning and others, see, [10, 21, 24, 46, 33].

The resolvent mapping $J_\gamma^G : H \to H$ associated with the multi-value mapping G is defined by

$$J_\gamma^G(x) := (I + \gamma G)^{-1}(x), \ \forall x \in H,$$

for some $\gamma > 0$, where I stands for the identity operator on H. It is well known that the variational inclusion problem (2.1.1) is equivalent to the fixed point equation $x^* = J_\gamma^G(I - \gamma F)x^*$. This concept has been used to modify many splitting algorithms for approximating a solution of the variational inclusion problem (2.1.1). One of the famous is the forward-backward splitting method [24, 53]

which is defined by the following method: x_1 and

$$x_{n+1} = J^G_{\gamma_n}(x_n - \gamma_n F x_n),\ n \geq 1,$$

where $\gamma_n > 0$ is the stepsize. This technique includes, as special cases the proximal point algorithm [16, 60, 85] and the gradient method [87, 88].

One way to study the convergence of algorithms is finding a method that makes an algorithm converges faster. Polyak [64] first introduced the inertial technique which was called heavy ball method in 1964 for convergence speeding up, this algorithm was generated for solving convex minimization.

In 2001, Alvarez and Attouch [3] extended the heavy ball method to encompass a broader context involving a general maximal monotone operator. This extension was achieved by strategically integrating the proximal point algorithm framework, resulting in the creation of the advanced inertial proximal point algorithm **(IPA)**, formally represented as:

$$x_{n+1} = J^G_{\gamma_n}(x_n + \theta_n(x_n - x_{n-1})), n \geq 1. \tag{2.1.2}$$

They proved that under the specified condition

$$\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty, \tag{2.1.3}$$

where $\{\theta_n\} \subset [0, 1)$ and $\{\gamma_n\}$ is nondecreasing, the algorithm (2.1.2) weakly converges to a zero of G. In particular, condition (2.1.3) is true for $\theta_n < \frac{1}{3}$. Here θ_n is an extrapolation factor and the inertial is represented by the term $\theta_n(x_n - x_{n-1})$. It is proved that the inertial terminology greatly improves the performance of the algorithm and has a nice convergence properties [4, 25, 31, 59].

For solving a zero-finding problem of the sum of two monotone operators.

The inertial proximal algorithm is the one of using the inertial technique with the forward-backward algorithm. This following inertial forward-backward algorithm **(IFBA)** have been proposed by Moudafi and Oliny [58]. Let $x_0, x_1 \in H$, $\gamma_n \in (0, \frac{2}{L})$ with L the Lipschitz constant of F, for all $n \geq 1$:

$$\begin{aligned} y_n &= x_n + \theta_n(x_n - x_{n-1}), \\ x_{n+1} &= J^G_{\gamma_n}(y_n - \gamma_n F x_n). \end{aligned}$$

Based on the condition generated in term of the sequence $\{x_n\}$ and parameter θ_n under a cocoercivity condition F with respect to the solution set, the weak convergence of the iterative sequence was established.

In 2015, Lorenz and Pock [54] proposed a modification of the inertial forward-backward splitting algorithm **(IFBSA)**. The scheme was generated as follows:

$$\begin{aligned} y_n &= x_n + \theta_n(x_n - x_{n-1}), \\ x_{n+1} &= J^G_{\gamma_n}(I - \gamma_n F)y_n, \end{aligned}$$

where $\theta_n \in [0, 1)$ is an extrapolation factor and γ_n is a step size parameter in positive real interval. Under conditions $G : H \to 2^H$ is maximal monotone mapping and $F : H \to H$ is single valued and cocoercive, then the sequence $\{x_n\}$ generated by IFBSA converges weakly to $(F + G)^{-1}(0)$.

The best choices for the inertial parameter θ_n is the interesting topic for many mathematicians. The well-known inertial parameter was introduced by Beck and Teboulle [12], it is embed in the well-known fast iterative shrinkage-thresholding algorithm **(FISTA)**. The FISTA was introduce for convex minimization problem which was generalized by the problem (2.1.1). A convex mini-

mization problem is defined as follow:

$$\min_{x\in H}\{f(x)+g(x)\}, \tag{2.1.4}$$

where f and g are two proper, lower semicontinuous and convex functions from H to the set of extended real numbers $\bar{\mathbb{R}} := \mathbb{R}\cup\{+\infty\}$ such that f is differentiable with Lipschitz continuous gradient ∇f and g is subdifferentiable with its computable proximal mapping. By setting $G \equiv \partial g$ and $F \equiv \nabla f$, then the variational inclusion problem (2.1.1) is reduced to the convex minimization problem (2.1.4) and $J_\lambda^{\partial g}(x) = (I+\lambda\partial g)^{-1}(x) = prox_{\lambda g}(x)$. The problem (2.1.4) has received a lot of attention by many authors to solve some data classification, for more information on the importance and development of data classifications and methods (see in [62, 63, 72, 73]).

The FISTA algorithm is designed by choosing $x_1 = y_0 \in H$, $t_1 = 1$, $\gamma > 0$ and compute

$$\begin{aligned} y_n &= prox_{\gamma g}(x_n - \gamma\nabla f x_n),\\ t_{n+1} &= \frac{1+\sqrt{1+4t_n^2}}{2},\\ \theta_n &= \frac{t_n-1}{t_{n+1}},\\ x_{n+1} &= y_n + \theta_n(y_n - y_{n-1}). \end{aligned}$$

FISTA has received a great deal of attention due to its excellent computational effect and important applications [16]. After that, Laing and Schonlieb [51] presented the proof of weak convergence theorem of modifying FISTA by setting $t_{n+1} = \frac{p+\sqrt{q+r(t_n)^2}}{2}$, where $p, q > 0$ and $0 < r \leq 4$.

Recently, Verma et al. [83] used the idea of the inertial technique of [64] with viscosity method and forward-backward algorithm for getting strong convergence for the convex minimization problem (2.1.4), which is called the

viscosity-base inertial forward-backward algorithm **(VIFBA)**:

$$\begin{aligned} y_n &= x_n + \theta_n(x_n - x_{n-1}), \\ z_n &= \alpha_n V(y_n) + (1+\alpha_n)y_n, \\ x_{n+1} &= prox_{\gamma_n g}(z_n - \gamma_n \nabla f z_n), \end{aligned}$$

where $L > 0$, $\theta_n \geq 0$, $\alpha_n \in (0,1)$, $\gamma_n \in (0, \frac{2}{L})$ and V is contractive mapping with constant c. In this algorithm, the inertial term was used flexibly under the suitable of its condition. A strong convergence has been prove under conditions on the parameters α_n, γ_n and the inertial term $\theta_n(x_n - x_{n-1})$.

Another way to study the convergence of algorithms is modifying the step size parameter to avoid calculating the norm operator of F. The famous one of algorithms was introduced by Gibali and Thong [37] in 2018, the algorithm was modified by Tseng type method combining with Mann algorithm for solving monotone variational inclusion. Let $x_0 \in H$, $\gamma_n > 0$, and $\mu \in (0,1)$. This algorithm is defined as follow:

$$\begin{aligned} y_n &= J^G_{\gamma_n}(I - \gamma_n F)x_n, \\ z_n &= y_n - \gamma_n(Fy_n - Fx_n), \\ x_{n+1} &= (1 - \alpha_n - \beta_n)x_n + \beta_n z_n, \end{aligned}$$

by using the step size:

$$\gamma_{n+1} = \begin{cases} \min\{\frac{\mu\|x_n - y_n\|}{\|Fx_n - Fy_n\|}, \gamma_n\}, & \text{if } Fx_n - Fy_n \neq 0, \\ \gamma_n, & \text{otherwise.} \end{cases}$$

A strong convergence has been proved under mild assumptions in Hilbert spaces.

Very recently, Inthakon et al. [45], proposed a modified step size which

is well-known that the linesearch technique calling Linesearch CN. The algorithm was generated by VIFBA algorithm with Linesearch CN. The method have been called a new machine learning algorithm **(NMLA)** based on optimization method. Their algorithm is of the form

$$\begin{aligned} y_n &= x_n + \theta_n(x_n - x_{n-1}), \\ z_n &= \alpha_n V(y_n) + (1-\alpha_n)y_n, \\ \gamma_n &= \text{Linesearch CN}(z_n, \sigma, \eta, \delta), \\ x_{n+1} &= prox_{\gamma_n g}(z_n - \gamma_n \nabla f z_n), \end{aligned}$$

where $\sigma > 0$, $\eta \in (0,1)$, $\delta \in (0, \frac{1}{2})$, $\theta_n \geq 0$, $\alpha_n \in (0,1)$ and c-contractive mapping V. The strong convergence theorem have been obtained.

The variational inequality problem that is to find a point $x^* \in C$ such that

$$\langle Fx^*, x - x^* \rangle \geq 0, \quad \forall x \in C, \tag{2.1.5}$$

where $F : C \to H$ is a nonlinear monotone operator. We denote $VI(C, F)$ is the solution set of variational inequality problem (2.1.5). In other words, variational inequality problems are a special case of the problem of finding zeros of the sum of two monotone operators. Note that the resolvent of the normal cone is nothing but the projection operator.

It is well known that the variational inequality problem (2.1.5) is equivalent to the fixed point problem, which consists of finding a point $x^* \in C$ such that

$$x^* = P_C(x^* - \gamma F x^*),$$

where γ is any positive real number. The variational inequality problem (2.1.5), which is a fundamental problem in nonlinear analysis and optimization theory,

finds many real application, such as signal recovery, image recovery, transportation problems, economics, engineering, see [10, 48, 69, 5, 80] and the references therein.

Recently, projection-based methods have been extensively investigated to solve variational inequality problem (2.1.5), see [18, 19]. An important projection method, which is called the Extragradient Method **(EGM)** was proposed by Korpelevich [49] in 1976, see also [6]. The method is generated by giving the current iterate x_n, compute

$$\begin{aligned} y_n &= P_C(x_n - \gamma F x_n), \\ x_{n+1} &= P_C(x_n - \gamma F y_n), \end{aligned} \tag{2.1.6}$$

where $\gamma \in (0, \frac{1}{L})$ and P_C denotes the metric projection from H onto C.

In recent years, the EGM (2.1.6) has received great attention from many authors, who improved it in various ways; see, e.g., [18, 19, 20] and the references therein. In 2011, Censor et al. [20] improved the EGM (2.1.6) in Hilbert spaces. Their method, called the subgradient extragradient method **(SEGM)**. Their method is of the form:

$$\begin{aligned} y_n &= P_C(x_n - \gamma F x_n), \\ T_n &= \{w \in H : \langle x_n - \gamma F x_n - y_n, w - y_n \rangle \leq 0\}, \\ x_{n+1} &= P_{T_n}(x_n - \gamma F y_n). \end{aligned} \tag{2.1.7}$$

In (2.1.7), the second projection P_C of the EGM (2.1.6) was replaced with a projection onto a half-space T_n which can be calculated easier more than a projection onto a complex closed convex set C. Under the assumptions of monotonicity and continuity of the operator F, Censor et al. [20] obtained weak convergence results based on (2.1.7).

Recently, Alvarez and Attouch [3], and Censor et al. [20] used the inertial extrapolation term to speed up the rate of convergence of the SEGM for solving (2.1.5) in Hilbert spaces. Their algorithm, called inertial subgradient extragradient method **(ISEGM)**. The algorithm is designed by choosing $x_0, x_1 \in H$ and compute

$$\begin{aligned} w_n &= x_n + \theta_n(x_n - x_{n-1}), \\ y_n &= P_C(w_n - \gamma F w_n), \\ T_n &= \{x \in H | \langle w_n - \gamma F w_n - y_n, x - y_n \rangle \leq 0\}, \\ x_{n+1} &= P_{T_n}(w_n - \gamma F y_n), \end{aligned}$$

where $\gamma > 0$, $\theta_n \geq 0$ are suitable parameters. Under several appropriate conditions imposed on these parameters, weak convergence result was established. It deserves mentioning that, in the above algorithm, the Lipschitz constant is known.

Our interest in this work is to study common solutions of variational inequality problems **(CVIP)**. The CVIP is stated as follows: Let C be a nonempty closed and convex subset of H. Let $F_i : H \to H$, $i = 1, 2, ..., N$ be mappings. The CVIP is to find $x^* \in C$ such that

$$\langle F_i x^*, x - x^* \rangle \geq 0, \ \forall x \in C, \ i = 1, 2, ..., N. \tag{2.1.8}$$

If $N = 1$, CVIP (2.1.8) becomes (2.1.5).

Recently, Suantai et al. [71] motivated the viscosity-type subgradient extragradient-line method which introduced by Shehu and Iyiola [49] to solve the CVIP (2.1.8). This algorithm was called the parallel viscosity-type subgradient extragradient-line method **(PVSEGM)**. The strong convergence theorem was proved when each of the operator F_i is Lipschitz continuous monotone mapping

that the Lipschitz constant is unknown. This algorithm starts with $x_1 \in H$ and computes

$$\begin{aligned} y_n^i &= P_C(x_n - \gamma_n^i F_i x_n), \quad \gamma_n^i = \rho^{l_n^i}, \\ \gamma_n^i \|F_i x_n - F_i y_n^i\| &\leq \mu \|r_{\rho^{l_n^i}}(x_n)\|), \\ z_n^i &= P_{T_n^i}(x_n - \gamma_n^i F y_n^i), \\ x_{n+1} &= \alpha_n^0 f(x_n) + \sum_{i=1}^{N} \alpha_n^i z_n^i, \quad n \geq 1, \end{aligned} \tag{2.1.9}$$

where $T_n^i = \{z \in H : \langle x_n - \gamma_n^i F_i x_n - y_n^i, z - y_n^i \rangle \leq 0\}$ with $\rho, \mu \in (0,1)$ and $\{\alpha_n\}_{n=1}^{\infty} \subseteq (0,1)$. The sequence $\{x_n\}_{n=1}^{\infty}$ generated by (2.1.9) was proved that it converges strongly to $x^* \in VI(C,F)$, where $x^* = P_{VI(C,F)} f(x^*)$ is the unique solution of the variational inequality

$$\left\langle (I-f)x^*, x - x^* \right\rangle \geq 0, \forall x \in VI(C,F).$$

Since f is a strict contraction, its Lipschitz constant k is, in fact, strictly less than 1 under the following conditions

$$(C_1) \quad \lim_{n\to\infty} \alpha_n^0 = 0 \quad \text{and} \quad (C_2) \quad \sum_{n=1}^{\infty} \alpha_n^0 = \infty.$$

The advantage of the PVSEGM was presented to solve the problem of multiblur effects in an image restoration. The image quality was improved sharper by using the PVSEGM in the resolution of common resolution variational inequality problem.

In this thesis, we design a forward-backward splitting algorithm with the inertial technique to solve the variational inclusion problem and show stepsize modification to generate another efficient algorithm. The weak convergence theorem are established under some suitable conditions in Hilbert spaces. Moreover,

we give an example and numerical results for supporting our main theorem in infinitely dimensional spaces. Finally, we apply our main result to solve a data classification problem, signal recovery, and image recovery. We then compare the performance of our algorithm with other algorithms.

CHAPTER 3

PRELIMINARIES

In this section, we provide some basic concepts, definitions, and lemmas that will be used in the following sections.

3.1 Fundamentals

Definition 3.1.1 [1]**(Normed space)** Let X be a vector space over field $\mathbb{S}$ ($\mathbb{R}$ or $\mathbb{C}$) and $\|\cdot\| : X \to [0, \infty)$ be a function. Then $\|\cdot\|$ is said to be a norm if the following properties hold: for all $x, y \in X$ and $\alpha \in \mathbb{S}$,

1. $\|x\| \geq 0$;
2. $\|x\| = 0 \Leftrightarrow x = 0$;
3. $\|\alpha x\| = |\alpha|\|x\|$;
4. $\|x + y\| \leq \|x\| + \|y\|$.

$\|x\|$ is called the norm of x. $(X, \|\cdot\|)$ denotes the normed space just defined.

Example 3.1.2 $\mathbb{R}^n$ is a normed space with the following norms:

$$\begin{aligned}
\|x\|_1 &= \sum_{i=1}^{n} |x_i| \text{ for all } x = (x_1, x_2, .., x_n) \in \mathbb{R}^n; \\
\|x\|_p &= \Big(\sum_{i=1}^{n} |x_i|^p\Big)^{1/p} \text{ for all } x = (x_1, x_2, .., x_n) \in \mathbb{R}^n \text{ and } p \in (1, \infty); \\
\|x\|_\infty &= \max_{1 \leq i \leq n} |x_i| \text{ for all } x = (x_1, x_2, .., x_n) \in \mathbb{R}^n.
\end{aligned}$$

Example 3.1.3 Let $X = \ell_1$, the linear space whose elements consist of all absolutely convergent sequences $(x_1, x_2, ..., x_i, ...)$ of scalars ($\mathbb{R}$ *or* $\mathbb{C}$),

$$\ell_1 = \{x : x = (x_1, x_2, ..., x_i, ...) \text{ and } \sum_{i=1}^{\infty} |x_i| < \infty\}.$$

Then ℓ_1 is a normed space with the norm defined by $\|x\|_1 = \sum_{i=1}^{\infty} |x_i|$.

Example 3.1.4 Let $X = \ell_p$ $(1 < p < \infty)$ be the linear space whose elements consist of all presumable sequences $(x_1, x_2, ..., x_i, ...)$ of scalars ($\mathbb{R}$ *or* $\mathbb{C}$),

$$\ell_p = \{x : x = (x_1, x_2, ..., x_i, ...) \text{ and } \sum_{i=1}^{\infty} |x_i|^p < \infty\}.$$

Then ℓ_p is a normed space with the norm defined by $\|x\|_p = (\sum_{i=1}^{\infty} |x_i|^p)^{1/p}$.

Example 3.1.5 Let $X = \ell_\infty$ be the linear space whose elements consist of all bounded sequences $(x_1, x_2, ..., x_i, ...)$ of scalars ($\mathbb{R}$ *or* $\mathbb{C}$),

$$\ell_\infty = \{x : x = (x_1, x_2, ..., x_i, ...) \text{ and } \{x_i\}_{i=1}^{\infty} \text{ is bounded}\}.$$

Then ℓ_∞ is a normed space with the norm defined by $\|x\|_\infty = \sup_{i \in \mathbb{N}} |x_i|$.

Example 3.1.6 Let $X = L_2[a, b]$ be the linear space of all continuous real-valued functions on $[a, b]$ forms a normed space X with norm defined by

$$\|x\| = \Big(\int_a^b x(t)^2 dt \Big)^{\frac{1}{2}}.$$

Definition 3.1.7 [1]**(Cauchy sequence)** A sequence $\{x_n\}$ in a normed space X is said to be Cauchy if $\lim_{m,n \to \infty} \|x_m - x_n\| = 0$, i.e., for $\varepsilon > 0$, there exists an integer $n_0 \in \mathbb{N}$ such that $\|x_m - x_n\| < \varepsilon$ for all $m, n \geq n_0$.

Definition 3.1.8 [1]**(Convergent sequence)** A sequence $\{x_n\}$ in a normed space X is said to be convergent to x if $\lim_{n \to \infty} \|x_n - x\| = 0$. In this case, we write $x_n \to x$ or $\lim_{n \to \infty} x_n = x$.

Definition 3.1.9 [74]**(Strong convergence)** Let H be an inner product space and let $x \in H$. A sequence $\{x_n\}$ in H is said to be converges strongly to x, denoted by $x_n \to x$, if $\|x_n - x\| \to 0$.

Definition 3.1.10 [50]**(Weak convergence)** A sequence $\{x_n\}$ in a normed space X is said to be weakly convergent if there is an $x \in X$ such that for every $f \in X^{'}$,

$$\lim_{n\to\infty} f(x_n) = f(x).$$

Definition 3.1.11 [1]**(Completeness)** The space X is said to be complete if every Cauchy sequence in X converges strongly.

Example 3.1.12 The Euclidean space $\mathbb{R}^n$ is complete with

$$d(x,y) = \sqrt{(x_1-y_1)^2 + (x_2-y_2)^2 + ... + (x_n-y_n)^2}$$

where $x = (x_1, x_2, ..., x_n)$, $y = (y_1, y_2, ..., y_n) \in \mathbb{R}^n$.

Example 3.1.13 The sequence space ℓ_∞ is complete.

Example 3.1.14 The sequence space ℓ_p is complete.

Definition 3.1.15 [1]**(Inner product space)** Let X be a vector space over field $\mathbb{S}$ ($\mathbb{R}$ or $\mathbb{C}$) and $\langle ., . \rangle : X \times X \to \mathbb{S}$ be a function. Then $\langle ., . \rangle$ is said to be an inner product if the following properties hold: for all $x, y \in X$ and $\alpha \in \mathbb{S}$,

1. $\langle x, x \rangle \geq 0$;
2. $\langle x, x \rangle = 0 \Leftrightarrow x = 0$;
3. $\langle \alpha x, y \rangle = \alpha \langle x, y \rangle$;
4. $\langle x, y \rangle = \overline{\langle y, x \rangle}$;
5. $\langle x + y, z \rangle = \langle x, z \rangle + \langle z, y \rangle$.

$\langle x, y \rangle$ is called the inner product of x and y, and $\overline{\langle y, x \rangle}$ is conjugate symmetry of $\langle x, y \rangle$. $(X, \|\cdot\|)$ denotes the inner product space just defined.

Definition 3.1.16 [1]**(Hilbert space)** An inner product space H is said to be a Hilbert space if it is complete, i.e., every Cauchy sequence is strongly convergent sequence in H.

Example 3.1.17 The Euclidean space $\mathbb{R}^n$ is a Hilbert space with inner product defined by

$$\langle x, y\rangle = \sum_{i=1}^{n} x_i y_i,$$

where $x = (x_1, x_2, ..., x_n)$, $y = (y_1, y_2, ..., y_n) \in \mathbb{R}^n$.

Example 3.1.18 The space l_2 is a Hilbert space with inner product defined by

$$\langle x, y\rangle = \sum_{i=1}^{\infty} x_i \overline{y_i},$$

where $x, y \in l_2$.

Example 3.1.19 The space $L_2[a, b]$ is a Hilbert space with inner product defined by

$$\langle x, y\rangle = \int_a^b x(t)y(t)dt,$$

where $a, b \in [-\infty, +\infty]$ and $a < b$.

Proposition 3.1.20 [1] Let X be an inner product space. Then the function $\|.\| : X \to [0, +\infty)$ defined by

$$\|x\| = \sqrt{\langle x, x\rangle}, \quad x \in X$$

is a norm on X.

Proposition 3.1.21 [17]**(The Cauchy-Schwarz inequality)** Let X be an inner product space. The following inequality holds for all $x, y \in X$:

$$|\langle x, y\rangle| \leq \|x\|\|y\|.$$

Proposition 3.1.22 [17]**(Properties of the inner product)** The following equalities hold: for all $x, y \in H$ and $\alpha \in [0, 1]$,

1. $\|x + y\|^2 = \|x\|^2 + \|y\|^2 + 2\langle x, y\rangle$,

2. $\|x-y\|^2 = \|x\|^2 + \|y\|^2 - 2\langle x, y\rangle$,
3. $\langle x+y, x-y\rangle = \|x\|^2 - \|y\|^2$,
4. $\|x+y\|^2 + \|x-y\|^2 = 2(\|x\|^2 + \|y\|^2)$,
5. $\|\alpha x + (1-\alpha)y\|^2 = \alpha\|x\|^2 + (1-\alpha)\|y\|^2 - \alpha(1-\alpha)\|x-y\|^2$.

Definition 3.1.23 [9]**(Bounded sequence)** Let H be an inner product space. A sequence $\{x_n\}$ in H is said to be bounded if there is $M > 0$ such that for all $n \in \mathbb{N}$,

$$\|x_n\| \leq M.$$

Definition 3.1.24 [1]**(Bounded linear operator)** Let X and Y be normed spaces and $T : X \to Y$ be a linear operator. The operator T is said to be bounded if there is a real number $M > 0$ such that for all $x \in X$,

$$\|Tx\| \leq M\|x\|.$$

Definition 3.1.25 [1]**(Convex subsets)** Let H be a Hilbert space. A subset $C \subseteq H$ is said to be convex, if $(1-\lambda)x + \lambda y \in C$ for all $x, y \in C$ and for all $\lambda \in [0,1]$.

Definition 3.1.26 [1]**(Closed set)** Let H be an inner product space. A subset C of H is said to be closed if for each a sequence $\{x_n\}$ in C with $x_n \to x$ implies that $x \in C$.

Definition 3.1.27 [10]**(Weak convergence in a Hilbert space)** A sequence $\{x_n\}$ in a Hilbert space H is said to converge weakly to a point x in H if

$$\langle x_n, y\rangle \to \langle x, y\rangle$$

for all $y \in H$ and denote that $x_n \rightharpoonup x$.

Proposition 3.1.28 [17] Let H be a Hilbert space. Then every bounded sequence $\{x_n\}$ in H, there exists a weakly convergent subsequence $\{x_{n_k}\}$ of $\{x_n\}$.

Definition 3.1.29 [17] Let H be a Hilbert space. Let $F : H \to H$ be an operator. Then

1. The operator F is called L-Lipschitz continuous with $L > 0$ if

$$\|Fx - Fy\| \leq L\|x - y\|, \quad \forall x, y \in H.$$

If $L = 1$, then F is called nonexpansive.

2. The operator F is called monotone if

$$\langle Fx - Fy, x - y\rangle \geq 0, \quad \forall x, y \in H.$$

3. The operator F is called firmly nonexpansive if

$$\|Fx - Fy\|^2 \leq \|x - y\|^2 - \|(I - F)x - (I - F)y\|^2,$$

or equivalently

$$\langle Fx - Fy, x - y\rangle \geq \|Fx - Fy\|^2, \quad \forall x, y \in C.$$

4. β-cocoercive or β-inverse strongly monotone if βF is firmly nonexpansive when $\beta > 0$.

Proposition 3.1.30 [89] *Let H be a Hilbert space and let $F : H \to H$ be a β-inverse-strongly monotone mapping, then*

1. F is an $\frac{1}{\beta}$-Lipschitz continuous and monotone mapping.

2. If γ is any constant in $(0, 2\beta]$, then the mapping $I - \gamma F$ is nonexpansive, where I is the identity mapping on H.

Definition 3.1.31 [17] Let C be a nonempty subset of H and $x \in H$. If there exists a point $x^* \in C$ such that

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

then x^* is called a metric projection of x on C, denoted by $P_C x$. It $P_C x$ exists and is unique for all x, then the function P_C of H onto C is called the metric projection.

Theorem 3.1.32 [35] Let C be a nonempty closed convex subset of H. Then, for any $x \in H$ there exists a metric projection $P_C x$ onto C and it is unique.

Proposition 3.1.33 [74] Let C be a nonempty convex subset of H and let $x \in H$, $x^* \in C$. Then,

$$x^* = P_C x \Leftrightarrow \langle x - x^*, y - x^* \rangle \leq 0, \ \forall y \in C.$$

Proposition 3.1.34 [89] *Let C be a nonempty closed convex subset of a Hilbert space X and P_C the metric projection from X onto C. Then the following hold:*

1. *P_C is idempotent:*

$$P_C(P_C(x)) = P_C(x) \quad \forall x \in X.$$

2. *P_C is firmly nonexpansive:*

$$\langle x - y, P_C(x) - P_C(y) \rangle \geq \|P_C(x) - P_C(y)\|^2 \quad \forall x, y \in X.$$

3. *P_C is nonexpansive:*

$$\|P_C(x) - P_C(y)\| \leq \|x - y\| \quad \forall x, y \in X.$$

4. P_C is monotone:

$$\langle P_C(x) - P_C(y), x - y\rangle \geq 0 \quad \forall x, y \in X.$$

5. P_C is demiclosed:

$$x_n \rightharpoonup x_0 \text{ and } P_C(x_n) \to y_0 \Rightarrow P_C(x_0) = y_0.$$

Definition 3.1.35 [10] Let H be a real Hilbert space and let $f : H \to R$, function f is said to be lower semi-continuous at x if $x_n \to x$, then

$$f(x) \leq \liminf_{n\to\infty} f(x_n).$$

Definition 3.1.36 [10]**(Proximal operate)** Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a closed proper convex function, The proximal operator of f is defined by

$$\text{prox}_f(y) = \arg\min_x \Big(f(x) + \frac{1}{2}\|x - y\|_2^2\Big),$$

and the proximal operator of the scalar function γf, where $\gamma > 0$, which can be expressed as

$$\text{prox}_{\gamma f}(y) = \arg\min_x \Big(f(x) + \frac{1}{2\gamma}\|x - y\|_2^2\Big),$$

then $\text{prox}_{\gamma f}$ is call the proximal operator of f with parameter γ.

Let $g : H \to (-\infty, +\infty]$ be a proper, lower semicontinuous and convex function. We denote the domain of g by $\text{dom}g = \{x \in H | g(x) < +\infty\}$. For any $x \in \text{dom}g$, the subdifferential of g at x is defined by

$$\partial g(x) = \{v \in H | \langle v, y - x\rangle \leq g(y) - g(x),\ y \in H\}.$$

Recall that the proximal operator $\text{prox}_g : \text{dom}(g) \to H$ is defined by $\text{prox}_g(x) = (I + \partial g)^{-1}(z),\ z \in H$. It is well-known that the proximal operator is single-valued. Furthermore, we have

$$\frac{z - \text{prox}_{\gamma g}(z)}{\gamma} \in \partial g(\text{prox}_{\gamma g}(z)) \qquad \text{for all } z \in H,\ \lambda > 0. \tag{3.1.1}$$

A differentiable function f is convex if and only if there holds the inequality

$$f(z) \geq f(x) + \langle \nabla f(x), z - x \rangle,\ \forall z \in H. \tag{3.1.2}$$

Proposition 3.1.37 [66] *If $g : H \to (-\infty, +\infty]$ is a proper lower semicontinuous convex function, then the subdiferential ∂g of g is a maximal monotone operator. In particular, if g is a constant function, then $\partial g = 0$ is maximal monotone and $J_\gamma^0 = I$ for $\gamma > 0$, where 0 is a zero operator.*

Example 3.1.38 Let C be a nonempty closed convex subset of H. An indicator function $i_C : H \to (-\infty, +\infty]$ of C is defined by

$$\partial i_C(x) = \begin{cases} 0, & \text{if } x \in C, \\ \infty, & \text{if } x \notin C. \end{cases}$$

Thus, i_C is proper, lower semicontinuous and convex, and hence ∂i_C is maximal monotone such that

$$\partial i_C(x) = \begin{cases} \{y \in H : \langle y, z - x \rangle \leq 0,\ \forall z \in C\}, & \text{if } x \in C, \\ \varnothing, & \text{if } x \notin C. \end{cases}$$

and $J_\gamma^{\partial i_C} = (I + \gamma \partial i_C)^{-1} = P_C$ for all $\gamma > 0$.

3.2 Lemmas

Lemma 3.2.1 [53] *Let $F : H \to H$ be a β-cocoercive mapping and $G : H \to 2^H$ a maximal monotone mapping. Then, we have*

1. *for $\gamma > 0$, $Fix\left(J_\gamma^G(I - \gamma F)\right) = (F + G)^{-1}(0)$;*

2. *for $0 < \gamma \leq \bar{\gamma}$ and $x \in H$, $\|x - J_\gamma^G(I - \gamma F)x\| \leq 2\|x - J_{\bar{\gamma}}^G(I - \bar{\gamma}F)x\|$.*

Lemma 3.2.2 [53] *Let H be a real Hilbert space. Assume that F is a β-inverse strongly monotone operator. Then, given $r > 0$, we have*

$$\begin{aligned}\|J_\gamma^G(I - \gamma F)x - J_\gamma^G(I - \gamma F)y\|^2 \quad \leq \quad & \|x - y\|^2 - \gamma(2\beta - \gamma)\|Fx - Fy\|^2 \\ & -\|(I - J_\gamma^G)(I - \gamma F)x - (I - J_\gamma^G)(I - \gamma F)y\|^2,\end{aligned}$$

for all $x, y \in G_\gamma = \{z \in H : \|z\| \leq \gamma\}$.

Lemma 3.2.3 [3] *Let $\{\varphi_n\}$, $\{\delta_n\}$ and $\{\alpha_n\}$ be the sequences in $[0, +\infty)$ such that $\varphi_{n+1} \leq \varphi_n + \alpha_n(\varphi_n - \varphi_{n-1}) + \delta_n$ for all $n \geq 1$, $\sum_{n=1}^{\infty} \delta_n < +\infty$ and there exists a real number α with $0 \leq \alpha_n \leq \alpha < 1$ for all $n \geq 1$. Then the followings hold:*

1. *$\Sigma_{n\geq 1}[\varphi_n - \varphi_{n-1}]_+ < +\infty$, where $[t]_+ = \max\{t, 0\}$;*

2. *there exists $\varphi^* \in [0, +\infty)$ such that $\lim_{n\to+\infty} \varphi_n = \varphi^*$.*

Lemma 3.2.4 [15] *Let C be a nonempty closed convex subset of a uniformly convex space X and T a nonexpansive mapping with the fixed point set of T is nonempty ($F(T) \neq \emptyset$). If $\{x_n\}$ is a sequence in C such that $x_n \rightharpoonup x$ and $(I - T)x_n \to t$, then $(I - T)x = t$. In particular, if $t = 0$, then $x \in F(T)$.*

Lemma 3.2.5 [70] *Let X be a Banach space satisfying Opial's condition and let $\{x_n\}$ be a sequence in X. Let p, $q \in X$ be such that*

$$\lim_{n\to\infty} \|x_n - p\| \quad \text{and} \quad \lim_{n\to\infty} \|x_n - q\| \quad \text{exist.}$$

If $\{x_{n_n}\}$ *and* $\{x_{m_n}\}$ *are subsequences of* $\{x_n\}$ *which converge weakly to* p *and* q*, respectively, then* $p = q$*.*

Lemma 3.2.6 [8] *Let* $\{a_n\}$ *and* $\{b_n\}$ *be nonnegative sequences of real numbers satisfying* $\sum_{n=1}^{\infty} b_n < \infty$ *and* $a_{n+1} \leq a_n + b_n$*. Then,* $\{a_n\}$ *is a convergent sequence.*

Lemma 3.2.7 [11] *Let* Ω *be a nonempty set of* H *and* $\{x_n\}$ *be a sequence in* H*. Suppose the following assertions hold.*

1. *For every* $x \in \Omega$*, the sequence* $\{\|x_n - x\|\}$ *converges.*

2. *Every weak sequential cluster point of* $\{x_n\}$ *belongs to* Ω*.*

Then $\{x_n\}$ *weakly converges to a point in* Ω*.*

CHAPTER 4

MAIN RESULTS

4.1 Inertial Mann forward-backward splitting algorithm for variational inclusion problems

In this section, let F be a β-cocoercive mapping on a real Hilbert space H and G be maximal monotone operator of H into 2^H such that $(F+G)^{-1}(0) \neq \emptyset$.

Algorithm 4.1.1 Inertial Mann forward-backward splitting algorithm (IMFBSA)

Initialization: *Select* $x_0, x_1 \in H, \{\alpha_n\} \subset (0,1), \{\gamma_n\} \subset (0, 2\beta)$ *and* $\{\theta_n\} \subset [0, \infty)$ *satisfies the condition such that*

$$0 < \liminf_{n\to\infty} \gamma_n \leq \limsup_{n\to\infty} \gamma_n < 2\beta \text{ and } \sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty.$$

Iterative step: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

$$y_n = x_n + \theta_n(x_n - x_{n-1}),$$

and

$$z_n = y_n + \alpha_n(x_n - y_n).$$

Step 2. *Compute*

$$x_{n+1} = J^G_{\gamma_n}(I - \gamma_n F)z_n.$$

Replace n *by* $n+1$ *and then repeat* ***Step 1.***

Theorem 4.1.2 *The sequence $\{x_n\}$ generated by IMFBSA weakly converges to a solution of $(F+G)^{-1}(0)$.*

Proof. Let $p \in (F+G)^{-1}(0)$. Since $J^G_{\gamma_n}$ is firmly nonexpansive mapping and F is β-cocoercive mapping, we deduce the following:

$$\begin{aligned}
\|x_{n+1}-p\|^2 &= \|J^G_{\gamma_n}(I-\gamma_n F)z_n - J^G_{\gamma_n}(I-\gamma_n F)p\|^2 \\
&\leq \|z_n - p - \gamma_n(Fz_n - Fp)\|^2 - \|z_n - x_{n+1} - \gamma_n(Fz_n - Fp)\|^2 \\
&= \|z_n-p\|^2 + \gamma_n^2\|Fz_n - Fp\|^2 - 2\gamma_n\langle z_n - p, Fz_n - Fp\rangle \\
&\quad -\|z_n - x_{n+1} - \gamma_n(Fz_n - Fp)\|^2 \\
&\leq \|z_n-p\|^2 - \gamma_n(2\beta-\gamma_n)\|Fz_n - Fp\|^2 \\
&\quad -\|z_n - x_{n+1} - \gamma_n(Fz_n - Fp)\|^2. \qquad (4.1.1)
\end{aligned}$$

Indeed, we have

$$\begin{aligned}
\|z_n - p\| &= \|y_n - p + \alpha_n(x_n - y_n)\| \\
&= \|\alpha_n(x_n - p) + (1-\alpha_n)(y_n - p)\| \\
&\leq \alpha_n\|x_n - p\| + (1-\alpha_n)\|y_n - p\| \\
&\leq \|x_n - p\| + (1-\alpha_n)\theta_n\|x_n - x_{n-1}\| \\
&\leq \|x_n - p\| + \theta_n\|x_n - x_{n-1}\|. \qquad (4.1.2)
\end{aligned}$$

By the condition of the sequence $\{\gamma_n\}$, there are $\gamma > 0$ and $n_0 \in \mathbb{N}$ such that $\gamma \leq \gamma_n < 2\beta$ for all $n \geq n_0$. It follows from (4.1.1) and (4.1.2) that, for all $n \geq n_0$,

$$\|x_{n+1} - p\| \leq \|z_n - p\| \leq \|x_n - p\| + \theta_n\|x_n - x_{n-1}\|. \qquad (4.1.3)$$

Applying Lemma 3.2.6 to the inequality (4.1.3) with $\sum_{n=1}^{\infty}\theta_n\|x_n - x_{n-1}\| < \infty$, we derive the sequence $\{\|x_n - p\|\}$ converges and so $\lim_{n\to\infty}\|x_n - p\| = \lim_{n\to\infty}\|z_n - p\|$.

Again, by (4.1.1), we obtain

$$\lim_{n\to\infty} \|Fz_n - Fp\| = \lim_{n\to\infty} \|z_n - x_{n+1} - \gamma_n(Fz_n - Fp)\| = 0.$$

This implies that

$$\lim_{n\to\infty} \|z_n - x_{n+1}\| = 0. \tag{4.1.4}$$

On the other hand, we gain

$$\|z_n - x_n\| = (1-\alpha_n)\|y_n - x_n\| \leq \theta_n \|x_n - x_{n-1}\| \to 0 \ \text{ as } \ n \to \infty. \tag{4.1.5}$$

From $\gamma \in (0, 2\beta)$, we have that the mapping $J_\gamma^G(I - \gamma F)$ is nonexpansive. Due to (4.1.4), (4.1.5) and Lemma 3.2.1 (2), the following result is obtained:

$$\begin{aligned} \|x_n - J_\gamma^G(I-\gamma F)x_n\| &\leq \|x_n - z_n\| + \|z_n - J_\gamma^G(I-\gamma F)z_n\| \\ &\quad + \|J_\gamma^G(I-\gamma F)z_n - J_\gamma^G(I-\gamma F)x_n\| \\ &\leq 2\|x_n - z_n\| + 2\|z_n - x_{n+1}\| \to 0 \ \text{ as } \ n \to \infty. \end{aligned}$$

Next, let $\bar{x}$ be a weak sequential cluster point of $\{x_n\}$. Using Lemma 3.2.1 (1) and Lemma 3.2.4, we deduce that $\bar{x} \in Fix\left(J_\gamma^G(I-\gamma F)\right) = (F+G)^{-1}(0)$. Finally, by Opial's lemma (Lemma 3.2.7), we can conclude that $\{x_n\}$ weakly converges to a solution of $(F+G)^{-1}(0)$. □

From Algorithm 4.1.1, we see that the parameter $\{\gamma_n\}$ needs to belong in $(0, 2\beta)$ for the convergence of the algorithm. So, the step size $\{\gamma_n\}$ can be considered in many ways. The update step is one of that, it has been updated every step using the previous iterative scheme. We expect that the update step size algorithm has received a great deal of attention due to its excellent computational effect and important applications: for example, see in [37, 68]. In this

work, we introduce a new update step size as follows:

$$\textbf{Stepsize UP}(\gamma, a_1, ..., a_N, k_1, ..., k_N) := \begin{cases} \min\{\gamma, a_1k_1, ..., a_Nk_N\}, \\ \text{if } a_ik_i < \infty \ \exists i \in \{1, 2, ..., N\}, \\ \gamma, \qquad\qquad \text{otherwise,} \end{cases} \tag{4.1.6}$$

where $\gamma \in (0, 2\beta)$ and $a_1, ..., a_N, k_1, ..., k_N \in [0, \infty)$. Inspired by Gibali [37], we provide the different algorithm based on the **Stepsize UP**. The relation between Algorithm 4.1.1 as following:

Algorithm 4.1.3 Initialization: *Select* $x_0, x_1 \in H, \{\alpha_n\} \subset (0, 1), \mu_1, \mu_2, \mu_3 \in (0, 2), \{\gamma_1\} \in (0, 2\beta)$ *and* $\{\theta_n\} \subset [0, \infty)$ *satisfies the condition such that*

$$\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty.$$

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

$$y_n = x_n + \theta_n(x_n - x_{n-1}),$$

and

$$z_n = y_n + \alpha_n(x_n - y_n).$$

Step 2. *Compute*

$$x_{n+1} = J^G_{\gamma_n}(I - \gamma_n F)z_n.$$

$$\gamma_{n+1} = \textbf{Stepsize UP}\Big(\gamma_n, \frac{\|y_n - x_{n+1}\|}{\|F(y_n) - F(x_{n+1})\|}, \frac{\|z_n - x_{n+1}\|}{\|F(z_n) - F(x_{n+1})\|}, \frac{\|z_n - y_n\|}{\|F(z_n) - F(y_n)\|}, \mu_1, \mu_2, \mu_3\Big).$$

Replace n with $n+1$ and then repeat ***Step 1****.*

Lemma 4.1.4 *The sequence $\{\gamma_n\}$ generated by* ***Stepsize UP*** *in Algorithm 4.1.3 is a nonincreasing sequence and*

$$\lim_{n\to\infty} \gamma_n = \gamma \geq \min\{\gamma_1, \mu_1\beta, \mu_2\beta, \mu_3\beta\}.$$

Proof. By the definition of the sequence $\{\gamma_n\}$, it is obvious that $\{\gamma_n\}$ is nonincreasing. On the other hand, we consider

$$\frac{\mu_1\|y_n - x_{n+1}\|}{\|F(y_n) - F(x_{n+1})\|} \geq \mu_1\beta,$$

$$\frac{\mu_2\|z_n - x_{n+1}\|}{\|F(z_n) - F(x_{n+1})\|} \geq \mu_2\beta,$$

and

$$\frac{\mu_3\|z_n - y_n\|}{\|F(z_n) - F(y_n)\|} \geq \mu_3\beta.$$

This shows that if $\|F(y_n) - F(x_{n+1})\| \neq 0$ or $\|F(z_n) - F(x_{n+1})\| \neq 0$ or $\|F(z_n) - F(y_n)\| \neq 0$, then the sequence $\{\gamma_n\}$ has the lower bound $\min\{\gamma_1, \mu_1\beta, \mu_2\beta, \mu_3\beta\}$. □

Remark 4.1.5 From Lemma 4.1.4, we see that the condition $0 < \liminf_{n\to\infty} \gamma_n \leq \limsup_{n\to\infty} \gamma_n < 2\beta$ in Algorithm 4.1.3 is satisfied. Thus, we obtain that the sequence $\{x_n\}$ generated by Algorithm 4.1.3 weakly converges to a solution of $(F+G)^{-1}(0)$.

4.2 Modified inertial forward-backward splitting methods for variational inclusion problems

In this section, we generate a new modified inertial forward-backward splitting algorithm for solving variational inclusion problems (2.1.1) and prove weak convergence theorem using some suitable conditions in Hilbert spaces.

Algorithm 4.2.1 Modified inertial forward-backward splitting algorithm (MIFBSA)

Initialization: *Select* $x_0, x_1 \in H$, $\{\gamma_n\} \subset (0, 2\beta)$, $\{\theta_n\} \subset [0, \theta]$ *for some* $\theta \in [0, 1)$, $\{\alpha_n\}$ *and* $\{\beta_n\}$ *are sequences in* $[0, 1]$.

Iterative step: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Compute*

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

Step 2. *Compute*

$$z_n = (1 - \beta_n)x_n + \beta_n J^G_{\gamma_n}(I - \gamma_n F)y_n.$$

Step 3. *Compute*

$$x_{n+1} = (1 - \alpha_n)J^G_{\gamma_n}(I - \gamma_n F)x_n + \alpha_n J^G_{\gamma_n}(I - \gamma_n F)z_n.$$

where $J^G_{\gamma_n} = (I + \gamma_n F)^{-1}$. *Set* $n = n + 1$ *and return to* **Step 1**.

Theorem 4.2.2 *The sequence* $\{x_n\}$ *generated by Algorithm 4.2.1 weakly converges to a solution of* $(F + G)^{-1}(0)$.

Assume that the following conditions hold:

1. $\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty$;
2. $\liminf_{n\to\infty} \alpha_n > 0$ *and* $\liminf_{n\to\infty} \beta_n > 0$;
3. $0 < \liminf_{n\to\infty} \gamma_n \leq \limsup_{n\to\infty} \gamma_n < 2\beta$.

Proof. Let $p \in (F+G)^{-1}(0)$. For each $n \in \mathbb{N}$. Since $J^G_{\gamma_n}(I-\gamma_n F)$ is nonexpansive when $\{\gamma_n\} \subset (0, 2\beta)$, we have

$$\|x_{n+1} - p\| \leq (1 - \alpha_n)\|J^G_{\gamma_n}(I - \gamma_n F)x_n - p\| + \alpha_n\|J^G_{\gamma_n}(I - \gamma_n F)z_n - p\|$$

$$\begin{aligned}
&\leq (1-\alpha_n)\|x_n-p\|+\alpha_n\|z_n-p\| \\
&\leq (1-\alpha_n)\|x_n-p\|+\alpha_n\big((1-\beta_n)\|x_n-p\|+\beta_n\|J^G_{\gamma_n}(I-\gamma_nF)t_n-p\|\big) \\
&\leq (1-\alpha_n)\|x_n-p\|+\alpha_n\big((1-\beta_n)\|x_n-p\|+\beta_n\|t_n-p\|\big) \\
&\leq \|x_n-p\|+\theta_n\|x_n-x_{n-1}\|. \qquad (4.2.1)
\end{aligned}$$

From Lemma 3.2.3 and the assumption (1), we obtain $\lim_{n\to\infty}\|x_n-p\|$ exists. This implies that $\{x_n\}$ is bounded and also $\{y_n\}$ and $\{z_n\}$. By Lemma 3.2.2, we have

$$\begin{aligned}
\|x_{n+1}-p\|^2 &\leq (1-\alpha_n)\|J^G_{\gamma_n}(I-\gamma_nF)x_n-p\|^2+\alpha_n\|J^G_{\gamma_n}(I-\gamma_nF)z_n-p\|^2 \\
&\leq (1-\alpha_n)\|x_n-p\|^2+\alpha_n\|z_n-p\|^2 \\
&\leq (1-\alpha_n)\|x_n-p\|^2+\alpha_n\big((1-\beta_n)\|x_n-p\|^2 \\
&\quad +\beta_n\|J^G_{\gamma_n}(I-\gamma_nF)y_n-p\|^2\big) \\
&\leq (1-\alpha_n)\|x_n-p\|^2+\alpha_n\big((1-\beta_n)\|x_n-p\|^2\big(\|y_n-p\|^2 \\
&\quad +\beta_n-\gamma_n(2\beta-\gamma_n)\|Fy_n-Fp\|^2 \\
&\quad -\varphi_q(\|y_n-\gamma_nFy_n-J^G_{\gamma_n}(I-\gamma_nF)y_n+\gamma_nFp\|)\big)\big) \\
&\leq \|x_n-p\|^2+2\alpha_n\beta_n\theta_n\langle x_n-x_{n-1},y_n-p\rangle \\
&\quad -\alpha_n\beta_n\big(\gamma_n(2\beta-\gamma_n)\|Fy_n-Fp\|^2 \\
&\quad -\varphi_q(\|y_n-\gamma_nF_n-J^G_{\gamma_n}(I-\gamma_nF)y_n+\gamma_nFp\|)\big).
\end{aligned}$$

This implies that

$$\begin{aligned}
&\alpha_n\beta_n\big(\gamma_n(2\beta-\gamma_n)\|Fy_n-Fp\|^2+\varphi_q(\|y_n-\gamma_nFy_n-J^G_{\gamma_n}(I-\gamma_nF)y_n+\gamma_nFp\|)\big) \\
&\leq \|x_n-p\|^2-\|x_{n+1}-p\|^2+2\alpha_n\beta_n\theta_n\langle x_n-x_{n-1},y_n-p\rangle. \qquad (4.2.2)
\end{aligned}$$

Since $\lim_{n\to\infty}\|x_n-p\|$ exists, it follows from (4.2.2) and the assumptions (1)-(3) that

$$\lim_{n\to\infty}\|Fy_n-Fp\|=\lim_{n\to\infty}\|y_n-\gamma_nFy_n-J^G_{\gamma_n}(I-\gamma_nF)y_n+\gamma_nFp\|=0. \quad (4.2.3)$$

This gives, by the triangle inequality, that

$$\lim_{n\to\infty} \|J^G_{\gamma_n}(I-\gamma_n F)y_n - y_n\| = 0. \tag{4.2.4}$$

Since $\liminf_{n\to\infty} \gamma_n > 0$, there is $\gamma > 0$ such that $\gamma_n \geq \gamma$ for all $n \geq 1$. Lemma 3.2.1 (2) yields that

$$\|J^G_{\gamma}(I-\gamma F)y_n - y_n\| \leq 2\|J^G_{\gamma_n}(I-\gamma_n F)y_n - y_n\|. \tag{4.2.5}$$

Then, by (4.2.4) and (4.2.5), we obtain

$$\lim_{n\to\infty} \|J^G_{\gamma}(I-\gamma F)y_n - y_n\| = 0. \tag{4.2.6}$$

By the definition of $\{x_n\}$ and our assumption (1), we have

$$\lim_{n\to\infty} \|y_n - x_n\| = \lim_{n\to\infty} \theta_n\|x_n - x_{n-1}\| = 0. \tag{4.2.7}$$

Since $\{x_n\}$ is bounded and H is reflexive, $\omega_w(x_n) = \{x \in H : x_{n_i} \rightharpoonup x, \{x_{n_i}\} \subset \{x_n\}\}$ is nonempty. Let $q \in \omega_w(x_n)$ be an arbitrary element. Then there exists a subsequence $\{x_{n_i}\} \subset \{x_n\}$ which converges weakly to q. Let $p \in \omega_w(x_n)$ and $\{x_{n_m}\} \subset \{x_n\}$ be such that $x_{n_m} \rightharpoonup p$. From (4.2.7), we also have $y_{n_i} \rightharpoonup q$ and $y_{n_m} \rightharpoonup p$. Since $J^G_{\gamma_n}(I-\gamma_n F)$ is nonexpansive and (4.2.6), by Lemma 3.2.4, we have $p, q \in (F+G)^{-1}(0)$. Applying Lemma 3.2.5, we obtain $p = q$. □

Remark 4.2.3 We can apply our new inertial forward-backward splitting algorithm to solve variational inequality problem, convex minimization problem, split feasibility problem and constrained linear system, see [23].

From Theorem 4.2.2, we see that the Algorithm 4.2.1 needs the condition (3) for being the nonexpansiveness of the mapping $J^G_{\gamma_n}(I-\gamma_n F)$. Using this concept, we can modify the step size $\{\gamma_n\}$ in many ways. One of that, we are

interested in the linesearh step size which has been updated every step. We expect that the linesearch step size algorithm has received a great deal of attention due to its excellent computational effect and important applications: for example, see in [13, 45]. In this paper, the new Stepsize N(a, b, c, μ, δ) step size $\{\gamma_n\}$ is chosen as follows:

$$\begin{cases} \gamma & = \mu, \\ \eta & = max\{10^2\|Fa\|, 10^2\|Fb\|, 10^2\|Fc\|, \frac{1}{\gamma}\}, \\ \gamma & = \frac{\delta}{\eta}, \end{cases} \tag{4.2.8}$$

where $\mu \in (0, 2\beta]$ and $\delta \in (0, 1)$. Based on the **Stepsize N**, the relation between Algorithm 4.2.1 with the other work is as following:

Algorithm 4.2.4 Initialization: *Select* $x_0, x_1 \in H$, $\mu \in (0, 2\beta]$, $\delta \in (0, 1)$, $\{\theta_n\} \subset [0, \theta]$ *for some* $\theta \in [0, 1)$, $\{\alpha_n\}$ *and* $\{\beta_n\}$ *are sequences in* $[0, 1]$ *and* N *is a stop number of iteration.*

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

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

Step 2. *Compute*

$$z_n = (1 - \beta_n)x_n + \beta_n J^G_{\gamma_n}(I - \gamma_n F)y_n.$$

Step 3. *Compute*

$$x_{n+1} = (1 - \alpha_n)J^G_{\gamma_n}(I - \gamma_n F)x_n + \alpha_n J^G_{\gamma_n}(I - \gamma_n F)z_n.$$

$$\gamma_{n+1} = \begin{cases} \textbf{\textit{Stepsize}}\ \boldsymbol{N}(y_n, x_n, z_n, \mu, \delta), & if\ 1 \le n < N, \\ \gamma_N, & otherwise, \end{cases}$$

where $J_{\gamma_n}^G = (I + \gamma_n^G)^{-1}$. *Replace* n *with* $n+1$ *and then repeat* ***Step 1***.

Theorem 4.2.5 *The sequence* $\{x_n\}$ *generated by Algorithm 4.2.4 weakly converges to* $z \in (F+G)^{-1}(0)$. *Assume that the following conditions hold:*

1. $\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty$;
2. $\liminf_{n\to\infty} \alpha_n > 0$ *and* $\liminf_{n\to\infty} \beta_n > 0$.

Remark 4.2.6 (1) From(4.2.8), we see that the sequence $\{\gamma_n\}$ which introduced by **Stepsize N** is nonincreasing and the limit of the sequence $\{\gamma_n\}$ is less than or equal to γ_N, so the condition (3) in Theorem 4.2.5 can be reduced.

(2) Since the sequence $\{\gamma_n\}$ of Theorem 4.2.5 satisfies the condition (3), hence Theorem 4.2.2 generalizes Theorem 4.2.5.

4.3 New projection algorithm for variational inclusion problems

In this section, we introduce a new projection algorithm for solving variational inclusion problem and the following conditions are assumed for the convergence of the method.

Algorithm 4.3.1 New projection algorithm (NPA)

Initialization: *Select* $x_0, x_1 \in H$, $\{\gamma_n\} \subset (0, 2\beta)$, $\{\theta_n\} \subset [0, \infty)$, $\{\eta_n\}$ *and* $\{\alpha_n\}$ *are sequences in* $(0, 1)$.

Iterative step: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

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

Step 2. *Compute*

$$z_n = y_n + \eta_n(x_n - y_n).$$

Step 3. *Compute*

$$x_{n+1} = P_C(\alpha_n y_n + (1-\alpha_n)J^G_{\gamma_n}(I-\gamma_n F)z_n).$$

Replace n *by* $n+1$ *and then repeat* ***Step 1.***

Theorem 4.3.2 *Assume that the following conditions hold:*

1. $\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty$;
2. $0 < \liminf_{n\to\infty} \gamma_n \leq \limsup_{n\to\infty} \gamma_n < 2\beta$;
3. $\limsup_{n\to\infty} \alpha_n > 1$.

Then the sequence $\{x_n\}$ *weakly converges to* $z \in (F+G)^{-1}(0) \cap C$.

Proof. Let $p \in (F+G)^{-1}(0) \cap C$. For each $n \in \mathbb{N}$, since $J^G_{\gamma_n}(I-\gamma_n F)$ is nonexpansive when $\{\gamma_n\} \subset (0, 2\beta)$ and F is β-inverse strongly monotone, we have

$$\begin{aligned}
\|x_{n+1} - p\| &= \|P_C(\alpha_n y_n + (1-\alpha_n)J^G_{\gamma_n}(I-\gamma_n F)z_n) - p\| \\
&\leq (1-\alpha_n)\|y_n - p\| + \alpha_n\|y_n - p\| \\
&\leq (1-\alpha_n)\|z_n - p\| + \alpha_n(\|x_n - p\| + \theta_n\|x_n - x_{n-1}\|) \\
&\leq (1-\alpha_n)((1-\eta_n)\|y_n - p\| + \eta_n\|x_n - p\|) \\
&\quad +\alpha_n(\|x_n - p\| + \theta_n\|x_n - x_{n-1}\|) \\
&\leq (1-\alpha_n)(\|x_n - p\| + (1-\eta_n)\theta_n\|x_n - x_{n-1}\|) \\
&\quad +\alpha_n(\|x_n - p\| + \theta_n\|x_n - x_{n-1}\|) \\
&\leq \|x_n - p\| + \theta_n\|x_n - x_{n-1}\|. \qquad (4.3.1)
\end{aligned}$$

From Lemma 3.2.3 and assumption (1), $\lim_{n\to\infty} \|x_n - p\|$ exists. This implies that $\{x_n\}$ is bounded. Since $J^G_{\gamma_n}$ is a firmly nonexpansive mapping, we have

$$\|x_{n+1} - p\|^2 = \|P_C(\alpha_n y_n + (1-\alpha_n)J^G_{\gamma_n}(I-\gamma_n F)z_n) - p\|^2$$

$$
\begin{aligned}
&\leq (1-\alpha_n)\|J^G_{\gamma_n}(I-\gamma_n F)z_n - p\|^2 + \alpha_n\|y_n - p\|^2 \\
&\leq (1-\alpha_n)(\|(I-\gamma_n F)z_n - (I-\gamma_n F)p\|^2 \\
&\quad -\|(I-J^G_{\gamma_n})(I-\gamma_n F)z_n - (I-J^G_{\gamma_n})(I-\gamma_n F)p\|^2) \\
&\quad +\alpha_n(\|x_n - p\|^2 + \theta_n\|x_n - x_{n-1}\|^2) \\
&= (1-\alpha_n)(\|z_n - p\|^2 + \gamma_n^2\|Fz_n - Fp\|^2 - 2\gamma_n\langle z_n - p, Fz_n - Fp\rangle \\
&\quad -\|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n - \gamma_n(Fz_n - Fp)\|^2) \\
&\quad +\alpha_n(\|x_n - p\|^2 + \theta_n\|x_n - x_{n-1}\|^2) \\
&\leq (1-\alpha_n)(\|z_n - p\|^2 + \gamma_n^2\|Fz_n - Fp\|^2 - 2\beta\gamma_n\|Fz_n - Fp\|^2 \\
&\quad -\|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n - \gamma_n(Fz_n - Fp)\|^2) \\
&\quad +\alpha_n(\|x_n - p\|^2 + \theta_n\|x_n - x_{n-1}\|^2) \\
&\leq (1-\alpha_n)[(1-\eta_n)(\|x_n - p\|^2 + \theta_n\|x_n - x_{n-1}\|^2) + \eta_n\|x_n - p\|^2 \\
&\quad -\gamma_n(2\beta-\gamma_n)\|Fz_n - Fp\|^2 - \|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n - \gamma_n(Fz_n - Fp)\|^2] \\
&\quad +\alpha_n(\|x_n - p\|^2 + \theta_n\|x_n - x_{n-1}\|^2) \\
&= \|x_n - p\|^2 + ((1-\eta_n) + \alpha_n\eta_n)\theta_n\|x_n - x_{n-1}\|^2 \\
&\quad -\gamma_n(2\beta-\gamma_n)(1-\alpha_n)\|Fz_n - Fp\|^2 \\
&\quad -(1-\alpha_n)\|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n - \gamma_n(Fz_n - Fp)\|^2.
\end{aligned}
$$

This implies that

$$
\begin{aligned}
&(1-\alpha_n)\|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n - \gamma_n(Fz_n - Fp)\|^2 + \gamma_n(2\beta-\gamma_n)\|Fz_n - Fp\|^2 \\
&\leq \|x_n - p\|^2 - \|x_{n+1} - p\|^2 + ((1-\eta_n) + \alpha_n\eta_n)\theta_n\|x_n - x_{n-1}\|^2. \qquad (4.3.2)
\end{aligned}
$$

Since $\lim_{n\to\infty}\|x_n - p\|$ exists, if follows from (4.3.2) and assumption (1)-(3) that

$$
\lim_{n\to\infty}\|Fz_n - Fp\| = \lim_{n\to\infty}\|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n - \gamma_n(Fz_n - Fp)\| = 0.
$$

This gives, by the triangle inequality, that

$$\lim_{n\to\infty} \|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n\| = 0. \tag{4.3.3}$$

Since $\liminf_{n\to\infty} \gamma_n > 0$, there is $\gamma > 0$ such that $\gamma_n > \gamma$. Lemma 3.2.1 (2), the following result is obtained:

$$\|z_n - J^G_{\gamma}(I-\gamma F)z_n\| \leq 2\|z_n - J^G_{\gamma_n}(I-\gamma_n F)z_n\|. \tag{4.3.4}$$

Then, by (4.3.3) and (4.3.4), we obtain

$$\lim_{n\to\infty} \|z_n - J^G_{\gamma}(I-\gamma F)z_n\| = 0.$$

Hence,

$$\begin{aligned} \lim_{n\to\infty} \|z_n - x_n\| &= (1-\eta_n) \lim_{n\to\infty} \|y_n - x_n\| \\ &= (1-\eta_n) \lim_{n\to\infty} \theta_n \|x_n - x_{n-1}\| = 0. \end{aligned} \tag{4.3.5}$$

Next, let $\bar{x}$ be a weak sequential cluster point of x_n. According to (4.3.5), we can observe that $\bar{x}$ is also a weak sequential cluster point of z_n. By applying Lemma 3.2.1 (1) and Lemma 3.2.4, we can get that $\bar{x} \in \mathrm{Fix}(J^G_{\gamma}(I-\gamma F)) = (F+G)^{-1}(0)$. Since x_n is a sequence in C and C is closed, it follows that $\bar{x} \in (F+G)^{-1}(0) \cap C$. By utilizing Opial's lemma (Lemma 3.2.7), we can obtain that x_n weakly converges to an element in $(F+G)^{-1}(0) \cap C$. □

4.4 Hybrid inertial parallel subgradient extragradient-line algorithm for variational inequality problems

In this section, we propose the hybrid inertial parallel subgradient extragradient-line method for solving CVIP (2.1.8). Let H be a real Hilbert

space and C be a nonempty closed convex subset of H. Let $F_i : H \to H$ be monotone mappings and L_i-Lipschitz continuous on H but L_i is unknown for all $i = 1, 2, ..., N$ such that $\Upsilon = \bigcap_{i=1}^{N} VI(C, F_i) \neq \emptyset$. Suppose $\{x_n\}_{n=1}^{\infty}$ is generated in the following:

Algorithm 4.4.1 Hybrid inertial parallel subgradient extragradient-line algorithm (HIPSEA)

Initialization: *Take* $\rho \in (0, 1)$, $\mu \in (0, 1)$. *Select arbitrary points* $x_0, x_1 \in H$ *and* $\{\theta_n\} \subseteq [0, \theta]$ *for some* $\theta \in [0, 1)$. *Set* $n := 1$.

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Compute*

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

Step 2. *Compute* y_n^i *for all* $i = 1, 2, ..., N$ *by*

$$y_n^i = P_C(t_n - \gamma_n^i F_i t_n),$$

where $\gamma_n^i = \rho^{k_n^i}$ *and* k_n^i *is the smallest nonnegative integer such that*

$$\gamma_n^i \parallel F_i t_n - F_i y_n^i \parallel \leq \mu \parallel t_n - y_n^i \parallel . \tag{4.4.1}$$

Step 3. *Compute*

$$z_n^i = P_{T_n^i}(t_n - \gamma_n^i F_i y_n^i),$$

where $T_n^i := \{z \in H : \langle t_n - \gamma_n^i F_i t_n - y_n^i, z - y_n^i \rangle \leq 0\}$.

Step 4. *Compute*

$$\bar{u}_n = \alpha_n^0(t_n) + \sum_{i=1}^{N} \alpha_n^i z_n^i, \tag{4.4.2}$$

$$\textit{where } \alpha_n^i \in (0,1),\ \ \forall i = 1,2,...,N \textit{ and } \sum_{i=0}^{N} \alpha_n^i = 1,\ \forall n \in N.$$

Step 5. *Compute*

$$x_{n+1} = P_{C_{n+1}} x_1,$$

$$\textit{where } C_{n+1} := \{z \in C_n :\| \bar{u}_n - z \| \leq \| t_n - z \|\}.$$

Set $n+1 \to n$ *and go to* **Step 1**.

Lemma 4.4.2 *There exists a nonnegative integer* k_n^i *satisfying (4.4.1).*

Proof. We first show that $\{t_n\}$ is bounded. Since Υ is a nonempty, closed and convex subset of H, there exists a unique $v \in \Upsilon$ such that $v = P_\Upsilon x_1$. From $x_n = P_{C_n} x_1$ and $x_{n+1} \in C_n$, for all $n \geq 1$, we obtain

$$\| x_n - x_1 \| \leq \| x_{n+1} - x_1 \| . \tag{4.4.3}$$

On the other hand, as $\Upsilon \subset C_n$, we obtain

$$\| x_n - x_1 \| \leq \| v - x_1 \| . \tag{4.4.4}$$

It follows from (4.4.3) and (4.4.4) that $\{x_n\}$ is bounded. By the definition of $\{x_n\}$, we obtain that $\{t_n\}$ is also bounded. For each $i = 1,2,...,N$ and $n \in \mathbb{N}$, we let $y_{k^i}^i = P_C(t_n - \rho^{k^i} F_i t_n)$ for all $k^i \in \mathbb{N}$. We divide the proof into two cases as follows:

case I: for each $i = 1,2,...,N$, if $\| t_n - y_{n_0^i}^i \| = 0$ for some $n_0^i \geq 1$, then there exists k_n^i such that $k_n^i \leq n_0^i$ satisfies (4.4.1).

case II: for each $i = 1,2,...,N$, if $\| t_n - y_{n_1^i}^i \| \neq 0$ for all $n_1^i \geq 1$, then we assume the contrary that

$$\rho^{n_1^i} \| F_i t_n - F_i y_{n_1^i}^i \| > \mu \| t_n - y_{n_1^i}^i \| .$$

From [34] Lemma 6.3 and the fact that $\rho \in (0,1)$, we obtain

$$\begin{aligned} \| F_i t_n - F_i y^i_{n^i_1} \| &> \frac{\mu}{\rho^{n^i_1}} \| t_n - y^i_{n^i_1} \| \\ &\geq \frac{\mu}{\rho^{n^i_1}} \min\{1, \rho^{n^i_1}\} \| t_n - y^i_0 \| \\ &= \mu \| t_n - y^i_0 \| . \end{aligned} \tag{4.4.5}$$

By using the continuity of P_C and the fact that $\{t_n\}$ is bounded, we have that

$$y^i_{n^i_1} = P_C(t_n - \rho^{n^i_1} F_i t_n) \to P_C(t_n), \ n^i_1 \to \infty \ \text{ for all } \ i = 1, 2, ..., N.$$

We consider two cases: $t_n \in C$ and $t_n \notin C$.

(i) If $t_n \in C$, then $t_n = P_C(t_n)$. Now, since $\| t_n - y^i_{n^i_1} \| \neq 0$ and $0 < \rho^{n^i_1} \leq 1$, it follows from [34] Lemma 6.3 that

$$\begin{aligned} 0 &< \| t_n - y^i_{n^i_1} \| \leq \max\{1, \rho^{n^i_1}\} \| t_n - y^i_0 \| \\ &= \| t_n - y^i_0 \| . \end{aligned} \tag{4.4.6}$$

Taking $n^i_1 \to \infty$ in (4.4.5) for each $i = 1, 2, ..., N$, we have that

$$0 = \| F_i t_n - F_i t_n \| \geq \mu \| t_n - y^i_0 \| > 0.$$

This is a contradiction and hence (4.4.1) is well defined.

(ii) If $t_n \notin C$, then for each $i = 1, 2, ..., N$, $\rho^{n^i_1} \| F_i t_n - F_i y^i_n \| \to 0$, as $n^i_1 \to \infty$ while

$$\begin{aligned} \lim_{n^i_1 \to \infty} \mu \| t_n - y^i_{n^i_1} \| &= \mu \lim_{n^i_1 \to \infty} \| t_n - P_C(t_n - \rho^{n^i_1} F_i t_n) \| \\ &= \mu \| t_n - P_C(t_n) \| > 0. \end{aligned}$$

This is a contradiction because $t_n \neq P_C(t_n)$. Therefore, linesearch in Algorithm

4.4.1 is well defined and implementable. □

Theorem 4.4.3 *Assume that the conditions hold:*

1. $\sum_{n=1}^{\infty} \theta_n \parallel x_n - x_{n-1} \parallel < \infty$.
2. $\liminf_{n\to\infty} \alpha_n^i > 0$ *for all* $i = 1, 2, ..., N$.

Then the sequence $\{x_n\}$ *generated by Algorithm 4.4.1 converges strongly to* $z \in \Upsilon$.

Proof. We split the proof into five steps.

Step 1. Show that $\{x_n\}$ is well defined. From $C_1 = C$, we see that C_1 is closed and convex. Assume that C_n is closed and convex. From the definition of C_{n+1} and Lemma 1.3 in [57], we obtain that C_{n+1} is closed and convex. Let $x^* \in \Upsilon$ and $s_n^i = t_n - \gamma_n^i F_i y_n^i, \forall n \geq 1, i = 1, 2, .., N$. Then,

$$\begin{aligned} \parallel z_n^i - x^* \parallel^2 &= \parallel P_{T_n^i}(s_n^i) - x^* \parallel^2 \\ &= \parallel P_{T_n^i}(s_n^i) - s_n^i \parallel^2 + 2\langle P_{T_n^i}(s_n^i) - s_n^i, s_n^i - x^* \rangle \\ &\quad + \parallel s_n^i - x^* \parallel^2 . \end{aligned} \tag{4.4.7}$$

From $x^* \in \Upsilon \subseteq C \subseteq T_n^i$ and the characterization of the metric projection $P_{T_n^i}$, we have

$$\begin{aligned} &2 \parallel s_n^i - P_{T_n^i}(s_n^i) \parallel^2 + 2\langle P_{T_n^i}(s_n^i) - s_n^i, s_n^i - x^* \rangle \\ &= 2\langle s_n^i - P_{T_n^i}(s_n^i), x^* - P_{T_n^i}(s_n^i) \rangle \leq 0. \end{aligned} \tag{4.4.8}$$

This implies that

$$\parallel s_n^i - P_{T_n^i}(s_n^i) \parallel^2 + 2\langle P_{T_n^i}(s_n^i) - s_n^i, s_n^i - x^* \rangle \leq - \parallel s_n^i - P_{T_n^i}(s_n^i) \parallel^2 . \tag{4.4.9}$$

By the definition of Algorithm 4.4.1 the inequalities (4.4.7) and (4.4.8), we have

$$\begin{aligned}
\| z_n^i - x^* \|^2 &\leq \| s_n^i - x^* \|^2 - \| s_n^i - z_n^i \|^2 \\
&= \| (t_n - x^*) - \gamma_n^i F_i y_n^i \|^2 - \| (t_n - z_n^i) - \gamma_n^i F_i y_n^i \|^2 \\
&= \| t_n - x^* \|^2 - \| t_n - z_n^i \|^2 + 2\gamma_n^i \langle -t_n + x^*, F_i y_n^i \rangle \\
&\quad + 2\gamma_n^i \langle t_n - z_n^i, F_i y_n^i \rangle \\
&= \| t_n - x^* \|^2 - \| t_n - z_n^i \|^2 + 2\gamma_n^i \langle x^* - z_n^i, F_i y_n^i \rangle. \quad (4.4.10)
\end{aligned}$$

By the monotonicity of the operator F_i, we have

$$\begin{aligned}
0 &\leq \langle F_i y_n^i - F_i x^*, y_n^i - x^* \rangle \\
&= \langle F_i y_n^i, y_n^i - x^* \rangle - \langle F_i x^*, y_n^i - x^* \rangle \\
&\leq \langle F_i y_n^i, y_n^i - x^* \rangle \\
&= \langle F_i y_n^i, y_n^i - z_n^i \rangle + \langle F_i y_n^i, z_n^i - x^* \rangle.
\end{aligned}$$

Thus

$$\langle x^* - z_n^i, F_i y_n^i \rangle \leq \langle F_i y_n^i, y_n^i - z_n^i \rangle. \quad (4.4.11)$$

Using (4.4.11) in (4.4.10), we obtain

$$\begin{aligned}
\| z_n^i - x^* \|^2 &\leq \| t_n - x^* \|^2 - \| t_n - z_n^i \|^2 + 2\gamma_n^i \langle F_i y_n^i, y_n^i - z_n^i \rangle \\
&= \| t_n - x^* \|^2 - \| t_n - y_n^i \|^2 - \| y_n^i - z_n^i \|^2 - 2\langle t_n - y_n^i, y_n^i - z_n^i \rangle \\
&\quad + 2\gamma_n^i \langle F_i y_n^i, y_n^i - z_n^i \rangle \\
&= \| t_n - x^* \|^2 - \| t_n - y_n^i \|^2 - \| y_n^i - z_n^i \|^2 \\
&\quad + 2\langle t_n - \gamma_n^i F_i y_n^i - y_n^i, z_n^i - y_n^i \rangle. \quad (4.4.12)
\end{aligned}$$

Observe that

$$\begin{aligned}\left\langle t_n - \gamma_n^i F_i y_n^i - y_n^i, z_n^i - y_n^i \right\rangle &= \left\langle t_n - \gamma_n^i F_i t_n - y_n^i, z_n^i - y_n^i \right\rangle \\ &\quad + \left\langle \gamma_n^i F_i t_n - \gamma_n^i F_i y_n^i, z_n^i - y_n^i \right\rangle \\ &\leq \left\langle \gamma_n^i F_i t_n - \gamma_n^i F_i y_n^i, z_n^i - y_n^i \right\rangle.\end{aligned}$$

Using the last inequality in (4.4.12), we have that

$$\begin{aligned}\| z_n^i - x^* \|^2 &\leq \| t_n - x^* \|^2 - \| t_n - y_n^i \|^2 - \| y_n^i - z_n^i \|^2 \\ &\quad + 2\left\langle \gamma_n^i F_i t_n - \gamma_n^i F_i y_n^i, z_n^i - y_n^i \right\rangle \\ &\leq \| t_n - x^* \|^2 - \| t_n - y_n^i \|^2 - \| y_n^i - z_n^i \|^2 \\ &\quad + 2\gamma_n^i \| F_i t_n - F_i y_n^i \| \| z_n^i - y_n^i \| \\ &\leq \| t_n - x^* \|^2 - \| t_n - y_n^i \|^2 - \| y_n^i - z_n^i \|^2 \\ &\quad + 2\mu \| t_n - y_n^i \| \| z_n^i - y_n^i \| \\ &\leq \| t_n - x^* \|^2 - \| t_n - y_n^i \|^2 - \| y_n^i - z_n^i \|^2 \\ &\quad + \mu(\| t_n - y_n^i \|^2 + \| z_n^i - y_n^i \|^2) \\ &= \| t_n - x^* \|^2 - (1-\mu)(\| t_n - y_n^i \|^2 + \| y_n^i - z_n^i \|^2), \quad (4.4.13)\end{aligned}$$

which implies that

$$\begin{aligned}\| \bar{u}_n - x^* \|^2 &\leq \alpha_n^0 \| t_n - x^* \|^2 + \sum_{i=1}^{N} \alpha_n^i \| z_n^i - x^* \|^2 \\ &\leq \| t_n - x^* \|^2 .\end{aligned}$$

This shows that $\| \bar{u}_n - x^* \| \leq \| t_n - x^* \|$. Hence, $x^* \in C_n$, $\forall n \geq 1$. This implies that $\{x_n\}$ is well-defined.

Step 2. Show that $x_n \to \omega \in C$ as $n \to \infty$. For $k > j$, since $x_k = P_{C_k} x_1 \in C_k \subset$

C_j, we have

$$\| x_k - x_j \|^2 \leq \| x_k - x_1 \|^2 - \| x_j - x_1 \|^2 .$$

By (4.4.3), (4.4.4), and the fact that $\{x_n\}$ is bounded and nonincreasing, we have that $\lim_{n\to\infty} \| x_n - x_1 \|$ exists. Hence, $\| x_k - x_j \| \to 0$, as $k, j \to \infty$, which means that $\{x_n\}$ is a Cauchy sequence. Hence, there exists $\omega \in C$ such that $x_n \to \omega$ as $n \to \infty$. In particular, we have

$$\lim_{n\to\infty} \| x_{n+1} - x_n \| = 0.$$

Step 3. Show that $\lim_{n\to\infty} \| x_n - y_n^i \| = \lim_{n\to\infty} \| y_n^i - z_n^i \| = 0$ for all $i = 1, 2, ..., N$. Let $x^* \in \Upsilon$. Then, we have from (4.4.2), (4.4.13) and Lemma 2.1 in [22] that

$$\begin{aligned}
\| \bar{u}_n - x^* \|^2 &= \| \alpha_n^0(t_n) + \sum_{i=1}^{N} \alpha_n^i z_n^i - x^* \|^2 \\
&\leq \alpha_n^0 \| t_n - x^* \|^2 + \sum_{i=1}^{N} \alpha_n^i \| z_n^i - x^* \|^2 \\
&= \| t_n - x^* \|^2 - (1-\mu) \sum_{i=1}^{N} \alpha_n^i (\| t_n - y_n^i \|^2 + \| y_n^i - z_n^i \|^2) \\
&= \| x_n - x^* \|^2 + \theta_n^2 \| x_n - x_{n-1} \|^2 + 2\langle x_n - x^*, \theta_n(x_n - x_{n-1}) \rangle \\
&\quad -(1-\mu) \sum_{i=1}^{N} \alpha_n^i (\| x_n - y_n^i \|^2 + \theta_n^2 \| x_n - x_{n-1} \|^2 \\
&\quad +2\langle x_n - y_n^i, \theta_n(x_n - x_{n-1}) \rangle + \| y_n^i - z_n^i \|^2). \qquad (4.4.14)
\end{aligned}$$

Since $x_{n+1} \in C_{n+1} \subset C_n$, we have

$$\begin{aligned}
\| \bar{u}_n - x_{n+1} \| &\leq \| t_n - x_{n+1} \| \\
&\leq \| t_n - x_n \| + \| x_n - x_{n+1} \| \\
&= \theta_n \| x_n - x_{n-1} \| + \| x_n - x_{n+1} \| \to 0, \text{ as } n \to \infty.
\end{aligned}$$

This implies that

$$\| \bar{u}_n - x_n \| \leq \| \bar{u}_n - x_{n+1} \| + \| x_{n+1} - x_n \| \to 0, \text{ as } n \to \infty. \quad (4.4.15)$$

It follows form (4.4.14) that

$$\begin{aligned}(1-\mu)\sum_{i=1}^{N}\alpha_n^i\big(\| x_n - y_n^i \|^2 + \| y_n^i - z_n^i \|^2 \big) \leq\ & \| x_n - x^* \|^2 - \| \bar{u}_n - x^* \|^2 \\ & +\theta_n^2 \| x_n - x_{n-1} \|^2 \\ & +2\big\langle x_n - x^*, \theta_n(x_n - x_{n-1})\big\rangle \\ & -(1-\mu)\sum_{i=1}^{N}\alpha_n^i\big(\theta_n^2 \| x_n - x_{n-1} \|^2 \\ & +2\big\langle x_n - y_n^i, \theta_n(x_n - x_{n-1})\big\rangle\big).\end{aligned}$$

By our assumptions (1), (2) and (4.4.15), we obtain

$$\lim_{n\to\infty} \| y_n^i - z_n^i \| = \lim_{n\to\infty} \| x_n - y_n^i \| = 0, \ \forall i = 1, 2, ..., N. \quad (4.4.16)$$

Step 4. Show that $\omega \in \Upsilon$. Since $\| x_n - t_n \| = \theta_n \| x_n - x_{n-1} \| \to 0$ and $x_n - y_n^i \to 0$, we have $t_n - \omega$ and $y_n^i \to \omega$. Since $y_n^i \in C$, we obtain $\omega \in C$. For all $x \in C$ using the property of the projection P_C, we have (Since F_i is monotone)

$$\begin{aligned}0 &\leq \big\langle y_n^i - t_n + \gamma_n^i F_i t_n, x - y_n^i \big\rangle \\ &= \big\langle y_n^i - t_n, x - y_n^i \big\rangle + \big\langle \gamma_n^i F_i t_n, x - x_n \big\rangle + \big\langle \gamma_n^i F_i t_n, x_n - y_n^i \big\rangle \\ &= \big\langle y_n^i - x_n, x - y_n^i \big\rangle + \theta_n \big\langle x_n - x_{n-1}, x - y_n^i \big\rangle. \quad (4.4.17)\end{aligned}$$

From [75] Remark 3.2, we know that $\inf_{n\geq 1} \gamma_n^i > 0$ for all $i = 1, 2, ..., N$. Taking $n \to \infty$ in (4.4.17) yields that $\langle F_i\omega, x - \omega\rangle \geq 0$, $\forall x \in C$. This implies that $\omega \in VI(C, F_i)$ for all $i = 1, 2, ..., N$. This completes the proof. □

Remark 4.4.4 The condition (1) is easily implemented in numerical computation since the value $\|x_n - x_{n-1}\|$ is known before choosing θ_n. Indeed, the parameter θ_n can be chosen such that

$$\theta_n = \begin{cases} \frac{\epsilon_n}{\|x_n - x_{n-1}\|} & if\ x_n \neq x_{n-1}, \\ \tau & otherwise, \end{cases} \tag{4.4.18}$$

where $\sum_{n=1}^{\infty} \epsilon_n < \infty$ and $\tau \geq 0$.

Base on the choice of the inertial parameter θ_n and the relation between Algorithm 4.4.1 where $F_i = F$ for all $i = 1, 2, ..., N$, Algorithm 4.4.1 is reduced to the following hybrid inertial subgradient extragradient algorithm.

Algorithm 4.4.5 Initialization: *Take* $\rho \in (0, 1)$, $\mu \in (0, 1)$. *Select arbitrary points* $x_0, x_1 \in H$ *and* $\{\theta_n\} \subseteq [0, \theta]$ *for some* $\theta \in [0, 1)$. *Set* $n := 1$.

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Compute*

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

Step 2. *Compute* y_n *by*

$$y_n = P_C(t_n - \gamma_n F t_n),$$

where $\gamma_n = \rho^{k_n}$ *and* k_n *is the smallest nonnegative integer such that*

$$\gamma_n \parallel Ft_n - Fy_n \parallel \leq \mu \parallel t_n - y_n \parallel . \tag{4.4.19}$$

Step 3. *Compute*

$$z_n = P_{T_n}(t_n - \gamma_n F y_n),$$

where $T_n := \{z \in H : \langle t_n - \gamma_n F t_n - y_n, z - y_n \rangle \leq 0\}$.

Step 4. *Compute*

$$\bar{u}_n = \alpha_n t_n + (1 - \alpha_n) z_n, \tag{4.4.20}$$

where $\alpha_n \in (0, 1)$.

Step 5. *Compute*

$$x_{n+1} = P_{C_{n+1}} x_1,$$

where $C_{n+1} := \{z \in C_n : \| \bar{u}_n - z \| \leq \| t_n - z \|\}$.

Set $n + 1 \to n$ *and go to* **Step 1**.

4.5 Numerical example

We now give an example in infinitely dimensional spaces $L_2[0, 1]$ such that $\|.\|$ is L_2-norm defined by $\|x\| = \sqrt{\int_0^1 |x(t)|^2 dt}$ where $x(t) \in L_2[0, 1]$ to support Theorem 4.2.2 and Theorem 4.2.5.

Example 4.5.1 Let $F : L_2[0, 1] \to L_2[0, 1]$ be defined by $Fx(t) = 4x(t)$ where $x(t) \in L_2[0, 1]$. Let $G : L_2[0, 1] \to L_2[0, 1]$ be defined by $Gx(t) = 2x(t)$ where $x(t) \in L_2[0, 1]$. We can choose $x_0(t) = \frac{\sin(t)}{2}$ and $x_1(t) = \sin(t)$. The stopping criterion is defined by $\|x_n - x_{n-1}\| < 10^{-2}$.

We start computation by comparing of the Algorithm 4.2.1 with different parameters θ where

$$\theta_n = \begin{cases} \frac{1}{n^2 \|x_n - x_{n-1}\|} & if x_n \neq x_{n-1} \text{ and } n > N, \\ \theta & otherwise, \end{cases} \tag{4.5.1}$$

where N is a number of iterations that we want to stop. We choose $\gamma_n = 0.1$ and $\alpha_n = \beta_n = \frac{n}{2n+1}$. Then, the results are presented in Table 1.

Table 1: Numerical results of the different θ.

θ	0.1	0.3	0.5	0.7	0.9
CPU Time	1.6621	1.1552	1.0693	1.1088	1.1106
Iteration No.	4	4	4	4	4

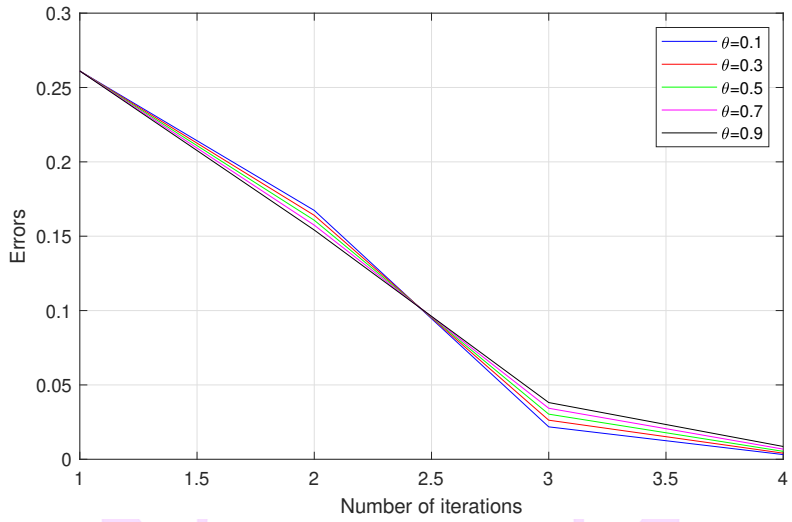


Figure 1: The Cauchy error plotting number of iterations for different θ.

We compare the performance of the Algorithm 4.2.1 and Algorithm 4.2.4 with different parameters γ_n by setting $\theta = 0.5$, $\alpha_n = \beta_n = \frac{n}{2n+1}$ and $\delta = 0.15$. Then, the results are presented in Table 2.

Table 2: Numerical results of the different γ_n.

γ_n	Algorithm 4.2.1					Algorithm 4.2.4
	0.1	0.3	0.5	0.7	0.9	**Stepsize N**, $\gamma_1 = 0.2$
CPU Time	1.6465	1.1856	1.7553	1.7359	1.6324	1.0479
Iteration No.	4	4	4	4	4	4

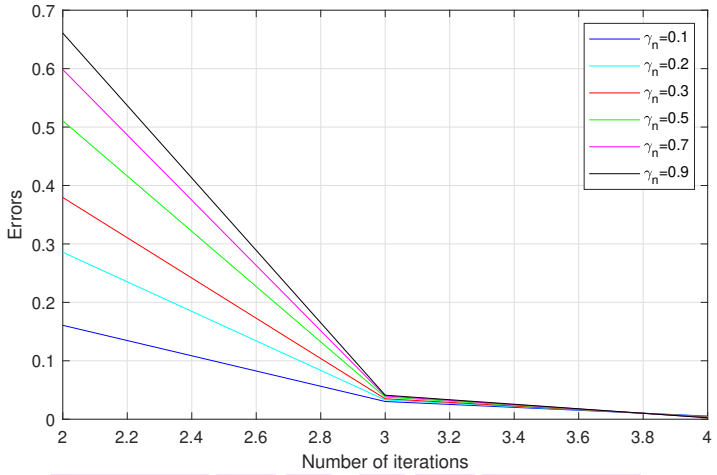


Figure 2: The Cauchy error plotting number of iterations for different γ_n.

We compare the performance of the Algorithm 4.2.4 with different parameters α_n by setting $\theta = 0.5$, $\gamma_n = 0.2$ and $\delta = 0.15$. Then, the results are presented in Table 3.

Table 3: Numerical results of the different α_n.

α_n	$\frac{n}{2n+1}$	$\frac{n}{5n+1}$	$\frac{n}{10n+1}$	$\frac{n}{50n+1}$	$\frac{n}{100n+1}$
CPU Time	1.5889	1.1416	1.0573	1.1312	1.0771
Iteration No.	4	4	4	4	4

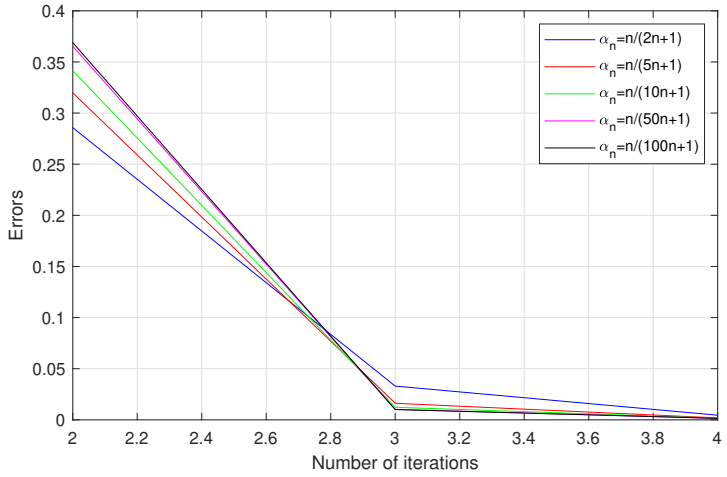


Figure 3: The Cauchy error plotting number of iterations for different α_n.

We compare the performance of the Algorithm 4.2.4 with different parameters β_n by setting $\theta = 0.5$, $\gamma_n = 0.2$ and $\delta = 0.15$. Then, the results are presented in Table 4.

Table 4: Numerical results of the different β_n.

β_n	$\frac{n}{2n+1}$	$\frac{n}{5n+1}$	$\frac{n}{10n+1}$	$\frac{n}{50n+1}$	$\frac{n}{100n+1}$
CPU Time	1.5889	1.1416	1.0573	1.1312	1.0771
Iteration No.	4	4	4	4	4

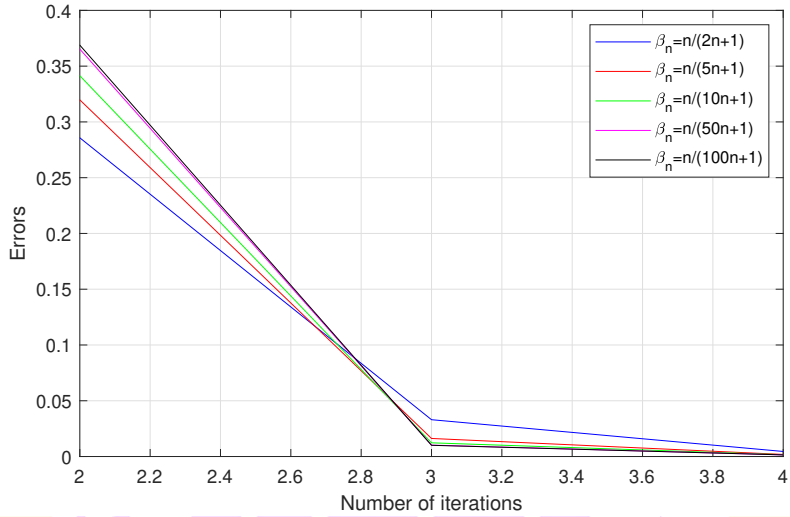


Figure 4: The Cauchy error plotting number of iterations for different β_n.

From Tables 1-4 and Figures 1-4, we noticed that in all cases, choosing $\theta = 0.5$, $\gamma_n = 0.2$, $\alpha_n = \frac{n}{10n+1}$, $\beta_n = \frac{n}{10n+1}$ and $\delta = 0.15$ yield the best results for the initialization $x_0(t) = \frac{sin(t)}{2}$ and $x_1(t) = \sin(t)$.

CHAPTER 5

APPLICATIONS

5.1 Application to Data Classification

For applying our algorithms to data classification, we focus on extreme learning machine **(ELM)** proposed by Huang [44]. Let $S := \{(x_n, t_n) : x_n \in \mathbb{R}^n, t_n \in \mathbb{R}^m, n = 1, 2, ..., N\}$ be a training set of N distinct samples, x_n is an input training data and t_n is a target. The output function of ELM for single-hidden layer feed forward neural networks (SLFNs) with M hidden nodes and activation function A is

$$O_n = \sum_{i=1}^{M} \omega_i A(\langle a_i, x_n \rangle + b_i),$$

where a_i and b_i are parameters of weight and finally the bias at the i-th hidden node, respectively. To find the optimal output weight ω_i at the i-th hidden node, then the hidden layer output matrix $\mathcal{H}$ is defined as follows:

$$\mathcal{H} = \begin{bmatrix} A(\langle a_1, x_1 \rangle + b_1) & \dots & A(\langle a_M, x_1 \rangle + b_M) \\ \vdots & \ddots & \vdots \\ A(\langle a_1, x_N \rangle + b_1) & \dots & A(\langle a_M, x_N \rangle + b_M) \end{bmatrix}.$$

The main goal of ELM is to find optimal output weight $\omega = [\omega_1^T, \omega_2^T, ..., \omega_M^T]^T$ such that $\mathcal{H}\omega = \mathcal{T}$, where $\mathcal{T} = [t_1^T, t_2^T, ..., t_N^T]^T$ is the training target data. In some cases, finding $\omega = \mathcal{H}^\dagger \mathcal{T}$, where $\mathcal{H}^\dagger$ is the Moore-Penrose generalized inverse of $\mathcal{H}$, it may be difficult to find when the matrix $\mathcal{H}$ does not exist. Thus, finding such a solution ω through convex minimization can overcome such difficulty.

We consider regularization of least square problems using techniques like L_1 (Lasso) and L_2 (Ridge) regularization. Regularization is a technique commonly used in machine learning and statistics to prevent overfitting and improve the generalization of models, which can lead to better generalization in classification problems. We conduct a series of experiments on a classification problem. The specific details of these problems are provided below.

1. The regularization of least square problem by L_1 (RLSP-L_1) or well-known called the least absolute shrinkage and selection operator (LASSO):

$$\min_{\omega\in\mathbb{R}^M}\{\frac{1}{2}\|\mathcal{H}\omega-\mathcal{T}\|_2^2+\lambda\|\omega\|_1\}, \tag{5.1.1}$$

where λ is a regularization parameter.

2. The regularization of least square problem by L_2 (RLSP-L_2):

$$\min_{\omega\in\mathbb{R}^M}\{\frac{1}{2}\|\mathcal{H}\omega-\mathcal{T}\|_2^2+\lambda\|\omega\|_2^2\}, \tag{5.1.2}$$

where λ is a regularization parameter.

3. The regularization of least square problem by L_1 with constrained by convex set L_1 (RLSP-CL_1):

$$\min_{\omega\in C}\{\frac{1}{2}\|\mathcal{H}\omega-\mathcal{T}\|_2^2+\lambda\|\omega\|_1\}, \tag{5.1.3}$$

where $C=\{\omega:\|\omega\|_1\leq\rho\}$, λ is a regularization parameter, and $\rho>0$.

4. The regularization of least square problem by L_2 with constrained by convex set L_2 (RLSP-CL_2):

$$\min_{\omega\in C}\{\frac{1}{2}\|\mathcal{H}\omega-\mathcal{T}\|_2^2+\lambda\|\omega\|_2^2\}, \tag{5.1.4}$$

where $C = \{\omega : \|\omega\|_2^2 \leq \rho\}$, λ is a regularization parameter, and $\rho > 0$.

5. The constrained least squares problem by convex set L_1 (CLSP-L_1):

$$\min_{\omega \in C} \frac{1}{2}\|\mathcal{H}\omega - \mathcal{T}\|_2^2, \tag{5.1.5}$$

where $C = \{\omega : \|\omega\|_1 \leq \rho\}$ and $\rho > 0$.

6. The constrained least squares problem by convex set L_2 (CLSP-L_2):

$$\min_{\omega \in C} \frac{1}{2}\|\mathcal{H}\omega - \mathcal{T}\|_2^2, \tag{5.1.6}$$

where $C = \{\omega : \|\omega\|_2^2 \leq \rho\}$ and $\rho > 0$.

For applying our algorithms to solve all of the convex minimization problems as above, we set our operator as in Table 5:

Table 5: Setting operators of our algorithms to solve all of the convex minimization problems (5.1.1)-(5.1.6).

Problems	Setting operators of our algorithms
RLSP-L_1 (5.1.1)	$F(\omega) \equiv \nabla(\frac{1}{2}\|\mathcal{H}\omega - T\|_2^2)$, $G(\omega) \equiv \partial(\lambda\|\omega\|_1)$, $C = H$
RLSP-L_2 (5.1.2)	$F(\omega) \equiv \nabla(\frac{1}{2}\|\mathcal{H}\omega - T\|_2^2)$, $G(\omega) \equiv \partial(\lambda\|\omega\|_2^2)$, $C = H$
RLSP-CL_1 (5.1.3)	$F(\omega) \equiv \nabla(\frac{1}{2}\|\mathcal{H}\omega - T\|_2^2)$, $G(\omega) \equiv \partial(\lambda\|\omega\|_1)$, $C = \{\omega : \|\omega\|_1 \leq \rho\}$
RLSP-CL_2 (5.1.4)	$F(\omega) \equiv \nabla(\frac{1}{2}\|\mathcal{H}\omega - T\|_2^2)$, $G(\omega) \equiv \partial(\lambda\|\omega\|_2^2)$, $C = \{\omega : \|\omega\|_2^2 \leq \rho\}$
CLSP-L_1 (5.1.5)	$F(\omega) \equiv \nabla(\frac{1}{2}\|\mathcal{H}\omega - T\|_2^2)$, $G(\omega) \equiv 0$, $C = \{\omega : \|\omega\|_1 \leq \rho\}$
CLSP-L_2 (5.1.6)	$F(\omega) \equiv \nabla(\frac{1}{2}\|\mathcal{H}\omega - T\|_2^2)$, $G(\omega) \equiv 0$, $C = \{\omega : \|\omega\|_2^2 \leq \rho\}$

The study emphasises assessing crucial performance metrics: accuracy, precision, recall, and F1-score [86]. These metrics are determined by considering specific values, such as true negatives ($\mathsf{T_N}$), false positives ($\mathsf{F_P}$), false negatives ($\mathsf{F_N}$), and true positives ($\mathsf{T_P}$).

$$\begin{aligned} Accuracy &= \frac{\mathsf{T_P}+\mathsf{T_N}}{\mathsf{T_P}+\mathsf{T_N}+\mathsf{F_P}+\mathsf{F_N}} \times 100\%. \\ Precision &= \frac{\mathsf{T_P}}{\mathsf{T_P}+\mathsf{F_P}} \times 100\%. \\ Recall &= \frac{\mathsf{T_P}}{\mathsf{T_P}+\mathsf{F_N}} \times 100\%. \\ F1-score &= \frac{2 \times (Precision \times Recall)}{(Precision + Recall)}. \end{aligned}$$

Binary cross entropy is a loss function used in machine learning and deep learning to measure the difference between predicted binary outcomes and actual binary labels. It quantifies the dissimilarity between probability distributions, aiding model training by penalizing inaccurate predictions. The standard binary cross entropy loss function [43] is the cross entropy loss when only two classes are involved. We can calculate the loss by calculating the following mean:

$$Loss = -\frac{1}{output\ size} \sum_{i=1}^{output\ size} y_i \log \hat{y}_i + (1-y_i) \log(1-\hat{y}_i),$$

where $\hat{y}_i$ is the i-th scalar value in the model output, y_i is the corresponding target value, and output size is the number of scalar values in the model output.

Example 5.1.1 The dataset used in this work is a breast cancer data retrieved from UCI Machine Learning repository [14] for training processing. This dataset was collected periodically over 3 years by Dr. William H. Wolberg from University of Wisconsin Hospitals. This dataset comprises of 699 instances, where the cases are labelled as either benign or malignant and 458 of the cases are benign and 241 are malignant. The dataset is partitioned into two classes 2 and 4, where 2 denotes

the benign class and 4 denotes the malignant class. For the implementation of the ML algorithms, the dataset was partitioned into the training set and testing set: 630 instances were used as a training dataset and the remaining 70 instances were used as a testing dataset. Which is divided into training and testing instances in the ratio of 90:10. There are 16 missing values of features in the dataset. The missing features are substituted by the mean for that feature. Finally, the dataset is randomized to guarantee the correct circulation of data. A comprehensive analysis is presented in Table 6.

Table 6: Breast cancer data attributes information.

Attribute Name	**Max**	**Min**	**Mean**	**Median**	**Standard Deviation**
Clump Thickness	10	1	4.4177	4	2.8157
Uniformity of Cell Size	10	1	3.1345	1	3.0515
Uniformity of Cell Shape	10	1	3.2074	1	2.9719
Marginal Adhesion	10	1	2.8069	1	2.8554
Single Epithelial Cell Size	10	1	3.2160	2	2.2143
Bare Nuclei	10	1	3.5447	1	3.6439
Bland Chromatin	10	1	3.4378	3	2.4384
Normal Nucleoli	10	1	2.8670	1	3.0536
Mitoses	10	1	1.5894	1	1.7151
Class	4	2	2.6896	2	0.9513

For starting our computation, we set $\gamma_n = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$, $\alpha_n = \frac{1}{2n+1}$ for Algorithm 4.1.1 and $\gamma_1 = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$, $\alpha_n = \frac{1}{2n+1}$, $\mu = 1.1$ for Algorithm 4.1.3. The stopping criteria is the binary cross entropy Loss $= 0.119$. We obtain the results of the different parameters $\bar{\theta}_n$ as seen in Table 7 when

$$\theta_n = \begin{cases} \frac{\bar{\theta}_n}{n^2\|x_n - x_{n-1}\|} & if\ x_n \neq x_{n-1}\ and\ n > N, \\ \bar{\theta}_n & otherwise, \end{cases}$$

where N is a number of iterations that we want to stop. We can see that parameters θ_n satisfies the condition in Algorithm 4.1.1 and Algorithm 4.1.3 all of each case of $\bar{\theta}_n$ in Table 7.

Table 7: Numerical results of the different parameter $\bar{\theta}_n$.

	$\bar{\theta}_n$	Iteration No.	Training Time
Algorithm 4.1.1	0	229	0.1221
	$\frac{1}{10n+1}$	309	0.1086
	$\frac{1}{n^2}$	317	0.1294
	$\frac{1}{n^2+1}$	297	0.1020
	$\frac{1}{\|x_n-x_{n-1}\|^2+n^2}$	315	0.1197
	$\frac{1}{\|x_n-x_{n-1}\|^5+n^5}$	229	0.0822
Algorithm 4.1.3	0	239	0.7253
	$\frac{1}{10n+1}$	213	0.3788
	$\frac{1}{n^2}$	213	0.4127
	$\frac{1}{n^2+1}$	213	0.4127
	$\frac{1}{\|x_n-x_{n-1}\|^2+n^2}$	223	0.3842
	$\frac{1}{\|x_n-x_{n-1}\|^5+n^5}$	218	0.4355

We can see that $\bar{\theta}_n = \frac{1}{\|x_n-x_{n-1}\|^5+n^5}$ greatly improves the performance of Algorithm 4.1.1 and $\bar{\theta}_n = \frac{1}{10n+1}$ of Algorithm 4.1.3. Therefore, we choose it as the default inertial parameter for later our calculation.

We next set the inertial parameters $\gamma_n = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$, $\bar{\theta}_n = \frac{1}{\|x_n-x_{n-1}\|^5+n^5}$ for Algorithm 4.1.1, and $\bar{\theta}_n = \frac{1}{\|x_n-x_{n-1}\|^5+n^5}$, $\gamma_1 = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$, $\mu = 1.1$ for Algorithm 4.1.3. For the different parameters α_n, we obtain the results as seen in Table 8 when

$$\alpha_n = \begin{cases} \bar{\alpha}_n & if\ n \leq N, \\ 0.5 & otherwise, \end{cases}$$

where N is a number of iterations that we want to stop. We can see that parameters α_n satisfies the condition in Algorithm 4.1.1 and Algorithm 4.1.3 all of each case of $\bar{\alpha}_n$ in Table 8.

Table 8: Numerical results of the different parameter α_n.

	$\bar{\alpha}_n$	Iteration No.	Training Time
Algorithm 4.1.1	$\frac{1}{n+1}$	229	0.0885
	$\frac{1}{2n+1}$	229	0.0822
	$\frac{1}{10n+1}$	229	0.0861
	$\frac{n}{2n+1}$	229	0.0779
	$\frac{n}{10n+1}$	229	0.0795
Algorithm 4.1.3	$\frac{1}{n+1}$	224	0.3584
	$\frac{1}{2n+1}$	213	0.3788
	$\frac{1}{10n+1}$	222	0.3652
	$\frac{n}{2n+1}$	223	0.4497
	$\frac{n}{10n+1}$	222	0.4205

From Table 8, notice that $\bar{\alpha}_n = \frac{n}{2n+1}$ has the lowest number of iteration for both of Algorithm 4.1.1 and $\bar{\alpha}_n = \frac{1}{2n+1}$ has the lowest number of iteration for both of Algorithm 4.1.3. So we choose it as the default parameter α_n for later our calculation.

Next we will set the inertial parameters $\bar{\theta}_n = \frac{1}{\|x_n - x_{n-1}\|^5 + n^5}$, $\bar{\alpha}_n = \frac{n}{2n+1}$ for Algorithm 4.1.1, and $\bar{\theta}_n = \frac{1}{10n+1}$, $\bar{\alpha}_n = \frac{1}{2n+1}$, $\mu = 1.1$ for Algorithm 4.1.3. For the different step size parameters γ_n of Algorithm 4.1.1 and γ_1 of Algorithm 4.1.3, respectively, we obtain the results as seen in Table 9.

Table 9: Numerical results of the different step size parameter γ_n of Algorithm 4.1.1 and γ_1 of Algorithm 4.1.3, respectively.

		Iteration No.	Training Time
Algorithm 4.1.1, γ_n	$\frac{1}{\|\mathcal{H}\|^2}$	231	0.0796
	$\frac{1}{2\|\mathcal{H}\|^2}$	535	0.1909
	$\frac{1}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$	535	0.1896
	$\frac{1.5}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$	356	0.1202
	$\frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$	229	0.0779
Algorithm 4.1.3, γ_1	$\frac{1}{\|\mathcal{H}\|^2}$	220	0.3993
	$\frac{1}{2\|\mathcal{H}\|^2}$	219	0.3908
	$\frac{1}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$	219	0.3692
	$\frac{1.5}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$	224	0.4220
	$\frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$	213	0.3788

We observe from Table 9 that $\gamma_n = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$ has the lowest number of iterations for both of Algorithm 4.1.1 and Algorithm 4.1.3. We next choose it as the default suitable step size for later our computation.

From Tables 7-9, we choose 1. $\bar{\theta}_n = \frac{1}{\|x_n - x_{n-1}\|^5 + n^5}$, $\bar{\alpha}_n = \frac{n}{2n+1}$ and $\gamma_n = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$ for Algorithm 4.1.1;

2. $\bar{\theta}_n = \frac{1}{10n+1}$, $\bar{\alpha}_n = \frac{1}{2n+1}$, $\mu = 1.1$ and $\gamma_1 = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$ for Algorithm 4.1.3;

3. $\gamma_n = \frac{1}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$ for FISTA;

4. $\bar{\theta}_n = \frac{1}{\|x_n - x_{n-1}\|^5 + n^5}$, $\bar{\alpha}_n = \frac{n}{2n+1}$, $\gamma_n = \frac{1.999}{2(max(eigenvalue(\mathcal{H}^T\times\mathcal{H})))}$ and $f(x) = 0.9999x$ for VIFBA.

For comparison, we set sigmoid as an activation function, hidden nodes $M = 100$ regularization parameter $\lambda = 1$.

Table 10: The performance of each algorithm with the stopping criteria as training accuracy > 90 and testing accuracy > 98.

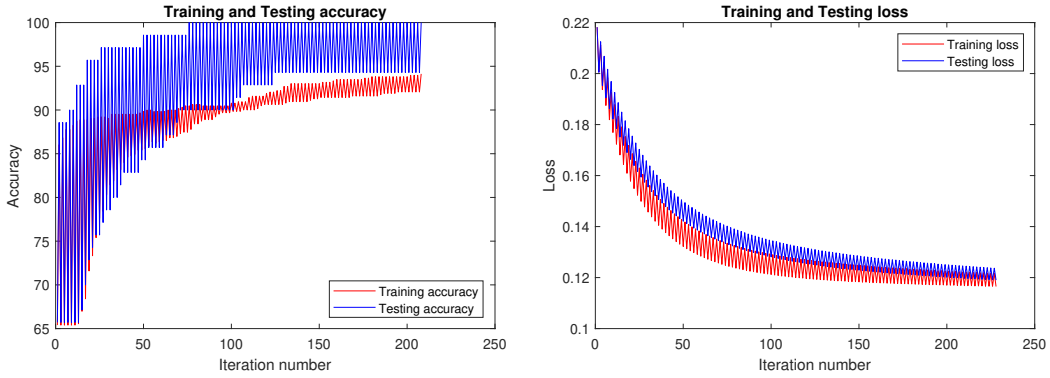
Algorithm	**Iteration No.**	**Training Time**	**Precision**	**Recall**	**Accuracy**
FISTA	74	0.0284	97.87	95.83	98.57
VIFBA	66	0.0249	96.00	97.83	98.57
Algorithm 4.1.1	52	0.0209	96.00	97.83	98.57
Algorithm 4.1.3	390	0.6915	97.97	95.83	98.57

Table 10 shows that our Algorithm 4.1.1 get the good number of iterations more than FISTA and VIFBA when we stopping criteria at the same testing accuracy greater than 98.

Table 11: The performance of each algorithm with the highest accuracy.

Algorithm	**Iteration No.**	**Training Time**	**Precision**	**Recall**	**Accuracy**
FISTA	74	0.0284	97.87	95.83	98.57
VIFBA	80	0.0303	100.00	100.00	100.00
Algorithm 4.1.1	76	0.0265	100.00	100.00	100.00
Algorithm 4.1.3	390	0.6915	97.97	95.83	98.57

Table 11 shows that Algorithm 4.1.1 has the highest efficiency in precision, recall and accuracy. It also has the lowest number of iterations. It has the highest probability of correctly classifying tumours compared to algorithms examinations. We next show the accuracy and loss graphs of training data and testing data for considering over-fitting of Algorithm 4.1.1.



Figures 5-6: Accuracy and Loss plots of the iteration.

From Figures 5-6, we see that our Algorithm 4.1.1 has good fit model this means that our Algorithm 4.1.1 suitably learns the training dataset and generalizes well to a hold out dataset.

Example 5.1.2 In this section, we predict breast cancer by dividing it into training and testing instances in the ratio of 90:10 for both datasets. The dataset was partitioned into training and testing sets: 630 instances were used as a training dataset, and the remaining 70 instances were used as a testing dataset.

For starting our computation, we set the suitable step size $\gamma_n = \frac{1}{2\|\mathcal{H}\|^2}$, $\bar{\alpha}_n = \bar{\beta}_n = \frac{n}{10n+1}$ and the binary cross entropy Loss $= 0.143$ as the stopping criteria, we obtain the results of the different parameters $\bar{\theta}_n$ as seen in Table 12 when

$$\theta_n = \begin{cases} \frac{\bar{\theta}_n}{n^2\|x_n - x_{n-1}\|} & if x_n \neq x_{n-1} \text{ and } n > N, \\ \bar{\theta}_n & otherwise, \end{cases} \tag{5.1.7}$$

where N is a number of iterations that we want to stop. We can see that the condition (1) in Theorem 4.2.2 is satisfied by parameter θ_n (5.1.7) all of each case of $\bar{\theta}_n$ in Table 12.

Table 12: Numerical results of each inertial different parameter $\bar{\theta}_n$.

$\bar{\theta}_n$	Iteration No.	Training Time
0	55	0.0491
$\frac{1}{\|x_n - x_{n-1}\| + n^3 + 1}$	55	0.0533
$\frac{1}{\|x_n - x_{n-1}\| + n^3}$	55	0.0514
$\frac{1}{n^2\|x_n - x_{n-1}\| + 1}$	43	0.0397
$\frac{t_n - 1}{t_{n+1}}$	21	0.0220
$\frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$	12	0.0141

We can see that $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$ greatly improves the performance of our Algorithm. Therefore, we choose it as the default inertial parameter for later our calculation.

We next set the inertial parameters $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$ for θ_n, $\bar{\alpha}_n = \bar{\beta}_n = \frac{n}{10n+1}$, $\delta = 0.99$ and the binary cross entropy Loss $= 0.143$ as the stopping criteria. For the different step size parameters γ_n, we obtain the results as seen in Table 13.

Table 13: Numerical results of the different step size parameters γ_n.

Algorithm	γ_n	Iteration No.	Training Time
Algorithm 4.2.1	$\frac{1}{2\|\mathcal{H}\|^2}$	12	0.0141
	$\frac{0.1}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$	104	0.0738
	$\frac{0.5}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$	21	0.0144
	$\frac{1}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$	12	0.0123
	$\frac{1.5}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$	9	0.0129
	$\frac{1.9}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$	8	0.0077
Algorithm 4.2.4	**Stepsize N**, $\gamma_1 = \frac{1}{\|\mathcal{H}\|^2}$	8	0.0070

We observe from Table 13 that $\gamma_n =$ **Stepsize N**$, \gamma_1 = \frac{1}{\|\mathcal{H}\|^2}$ has the lowest number of iterations. We next choose it as the default suitable step size for later our computation.

Next we will set the inertial parameters $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$ for θ_n, $\gamma_n = \frac{1}{\|\mathcal{H}\|^2}$, $\delta = 0.99$ and the binary cross entropy Loss $= 0.143$ as the stopping criteria. For the different parameter α_n, we obtain the results as seen in Table 14 when

$$\alpha_n = \begin{cases} \bar{\alpha}_n & if\ n \leq N, \\ 0.5 & otherwise, \end{cases} \tag{5.1.8}$$

where N is a number of iterations that we want to stop. We can see that parameters α_n satisfies the condition (2) in Theorem 4.2.5 all of each case of $\bar{\alpha}_n$ in Table 14.

Table 14: Numerical results of the different parameters α_n.

$\bar{\alpha}_n$	Iteration No.	Training Time
$\frac{n}{n+1}$	48	0.0486
$\frac{n}{2n+1}$	28	0.0210
$\frac{n}{5n+1}$	11	0.0099
$\frac{n}{10n+1}$	8	0.0105
$\frac{n}{50n+1}$	6	0.0072
$\frac{n}{100n+1}$	6	0.0063

From Table 14, notice that $\bar{\alpha}_n = \frac{n}{100n+1}$ has the lowest number of iterations. So we choose it as the default parameter α_n for later our calculation.

We next set the inertial parameters $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$ for θ_n, $\gamma_n = \frac{1}{\|\mathcal{H}\|^2}$, $\delta = 0.99$ and the binary cross entropy Loss $= 0.143$ as the stopping criteria. For the different parameter β_n, we obtain the results as seen in Table 15 when

$$\beta_n = \begin{cases} \bar{\beta}_n & if\ n \leq N, \\ 0.5 & otherwise, \end{cases} \tag{5.1.9}$$

where N is a number of iterations that we want to stop. We can see that parameters β_n satisfies the condition (2) in Theorem 4.2.5 all of each case of $\bar{\beta}_n$ in Table 15.

Table 15: Numerical results of the different parameters β_n.

$\bar{\beta}_n$	Iteration No.	Training Time
$\frac{n}{n+1}$	48	0.0486
$\frac{n}{2n+1}$	28	0.0210
$\frac{n}{5n+1}$	11	0.0099
$\frac{n}{10n+1}$	8	0.0105
$\frac{n}{50n+1}$	6	0.0072
$\frac{n}{100n+1}$	6	0.0063

From Table 15, notice that $\bar{\beta}_n = \frac{n}{100n+1}$ has the lowest number of iterations. So we choose it as the default parameter β_n for later our calculation.

From Table 12-15, we choose 1. $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$, $\bar{\alpha}_n = \bar{\beta}_n = \frac{n}{100n+1}$, $\gamma_n = \frac{1}{\|\mathcal{H}\|^2}$ and $\delta = 0.99$ for Algorithm 4.2.4;

2. $\gamma_n = \frac{1.9}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$ for FBSA;

3. $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$ and $\gamma_n = \frac{1}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$ for IFBSA;

4. $\gamma_n = \frac{1}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$ for FISTA;

5. $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$, $\bar{\alpha}_n = \frac{1}{n}$, $\gamma_n = \frac{1}{2(max(eigenvalue(\mathcal{H}^T \times \mathcal{H})))}$ and $f(x) = 0.999x$ for VIFBA;

6. $\bar{\theta}_n = \frac{10^{10}}{\|x_n - x_{n-1}\|^3 + n^3 + 10^{10}}$, $\bar{\alpha}_n = \frac{1}{n}$, $\delta = 0.1$, $\sigma = 0.49$, $\eta = 0.1$ and $f(x) = 0.999x$ for NMLA.

For comparison, We set sigmoid as an activation function, hidden nodes $M = 100$ and regularization parameter $\lambda = 1 \times 10^{-5}$.

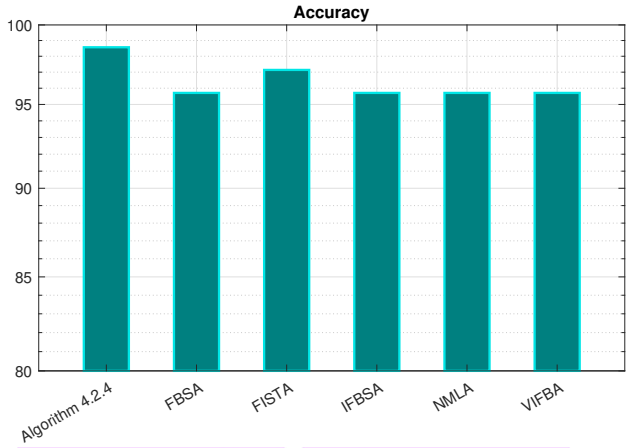
Table 16: The performance of each algorithm with the stopping criteria as training accuracy > 95 and testing accuracy > 95.

Algorithm	**Iteration No.**	**Training Time**	**Precision**	**Recall**	**Accuracy**
FBSA	644	0.2328	97.87	95.83	98.57
IFBSA	70	0.0326	93.88	87.50	95.71
FISTA	94	0.0236	97.87	95.83	98.57
VIFBA	68	0.0151	93.88	87.50	95.71
NMLA	60	0.3666	95.74	91.67	95.71
Algorithm 4.2.4	53	0.0434	96.00	97.83	98.57

Table 17: The performance of each algorithm at the 60 iteration.

Algorithm	**Iteration No.**	**Training Time**	**Precision**	**Recall**	**Accuracy**
FBSA	60	0.0224	93.88	87.50	95.71
IFBSA	60	0.0330	93.88	87.50	95.71
FISTA	60	0.0628	95.83	91.67	97.14
VIFBA	60	0.0527	93.88	87.50	95.71
NMLA	60	0.6227	93.88	87.50	95.71
Algorithm 4.2.4	60	0.1022	96.00	97.83	98.57

Table 17 shows that our algorithm obtained the highest performance in precision, recall, and accuracy with the highest probability of correctly classifying tumours in comparison to the investigated algorithms.



Figures 7-9: Comparison of models based on precision, recall and accuracy, respectively.

From Figures 7-9, it is seen that our inertial forward-backward splitting algorithm (4.2.4) performs decently in terms of precision, recall and accuracy, especially when compared with FBSA, IFBSA, FISTA, VIFBA, and NMLA.

Example 5.1.3 In this thesis, we use the cervical cancer behavior risk dataset from the UCI (University of California Irvine) Machine Learning Repository [55, 82]. The dataset consists of 72 samples related to cervical cancer, with 19 distinct attributes and one designated target column. The target column represents the class attribute, where a value of 1 indicates the presence of cervical cancer, while a value of 0 represents its absence. For a comprehensive understanding of the dataset's statistical properties, please refer to Table 18.

Table 18: Cervical cancer behavior risk data attribute information.

Attribute Name	**Max**	**Min**	**Mean**	**Median**	**Standard Deviation**
behavior_sexualRisk	10	2	9.6667	10	1.1868
behavior_eating	15	3	12.7917	13	2.3613
behavior_personalHygine	15	3	11.0833	11	3.0338
intention_aggregation	10	2	7.9028	10	2.7381
intention_commitment	15	6	13.3472	15	2.3745
attitude_consistency	10	2	7.1806	7	1.5228
attitude_spontaneity	10	4	8.6111	9	1.5157
norm_significantPerson	5	1	3.1250	3	1.8457
norm_fulfillment	15	3	8.4861	7	4.9076
perception_vulnerability	15	3	8.5139	8	4.2757
perception_severity	10	2	5.3889	4	3.4007
motivation_strength	15	3	12.6528	14	3.2072
motivation_willingness	15	3	9.6944	11	4.1304
socialSupport_emotionality	15	3	8.0972	9	4.2432
socialSupport_appreciation	10	2	6.1667	6.5	2.8973
socialSupport_instrumental	15	3	10.3750	12	4.3165
empowerment_knowledge	15	3	10.5417	12	4.3668
empowerment_abilities	15	3	9.3194	10	4.1819
empowerment_desires	15	3	10.2778	11	4.4823
ca_cervix	1	0	0.29167	0	0.4577

Next, we divide the dataset into 80% for training and 20% for testing. We set $\gamma_n = \frac{0.9999}{max(eigenvalue(\mathcal{H}^T \times \mathcal{H}))}$, $\eta_n = \frac{1}{100n+1}$, and $M = 270$. We compare the performance of FBSA, IFBSA, and our algorithm. All the parameters are chosen as seen in Table 19.

Table 19: Chosen parameters of each algorithm.

Algorithm	α_n	θ_n	λ	ρ
FBSA	-	$\frac{10^{10}}{\|x_n-x_{n-1}\|^5+n^5+10^{10}}$	0.5	-
IFBSA	$\frac{9n}{10n+1}$	$\frac{1}{\|x_n-x_{n-1}\|^5+n^5}$	1	-
Algorithm 4.3.1 (RLSP-L_1)	$\frac{1}{100n+1}$	$\frac{10^{10}}{\|x_n-x_{n-1}\|^5+n^5+10^{10}}$	0.5	-
Algorithm 4.3.1 (RLSP-L_2)	0.9	0.999	10^{-5}	-
Algorithm 4.3.1 (RLSP-CL_1)	$\frac{1}{100n+1}$	0.99	0.5	3
Algorithm 4.3.1 (RLSP-CL_2)	0.9	0.9999	10^{-5}	3
Algorithm 4.3.1 (CLSP-L_1)	0.9	0.999	-	5
Algorithm 4.3.1 (CLSP-L_2)	0.9	0.9999	-	3

Table 20: Comparison of the performance with each algorithm.

Algorithm	**Iteration No.**	**Training time**	**Precision**	**Recall**	**F1-score**	**Accuracy**
FBSA	45	0.0065	100.00	75.00	85.71	92.86
IFBSA	105	0.0066	100.00	50.00	66.67	85.71
Algorithm 4.3.1 (RLSP-L_1)	45	0.0094	100.00	75.00	85.71	92.86
Algorithm 4.3.1 (RLSP-L_2)	361	0.0158	100.00	100.00	100.00	100.00
Algorithm 4.3.1 (RLSP-CL_1)	48	0.0056	100.00	75.00	85.71	92.86
Algorithm 4.3.1 (RLSP-CL_2)	361	0.0203	100.00	100.00	100.00	100.00
Algorithm 4.3.1 (CLSP-L_1)	154	0.0092	100.00	75.00	85.71	92.86
Algorithm 4.3.1 (CLSP-L_2)	361	0.0171	100.00	100.00	100.00	100.00

Table 20 shows the results of the Algorithms compared in terms of accuracy, precision, recall, and F1-score. The results obtained Algorithm 4.3.1 (RLSP-L_2), Algorithm 4.3.1 (RLSP-CL_2), and Algorithm 4.3.1 (CLSP-L_2) have the highest performance metrics. The lowest on the accuracy, precision, recall, and F1-score are IFBSA.

We compare analysis, evaluating our algorithm against various machine learning models, such as logistic regression, Naive Bayes, k-nearest neighbors (kNN), weighted kNN (WkNN), artificial neural networks, decision trees, random forests, eXtreme Gradient Boosting (XGBoost), statistical implicative analysis (SIA), multilayer perceptron (MLP), and Sigmoid-support vector machines

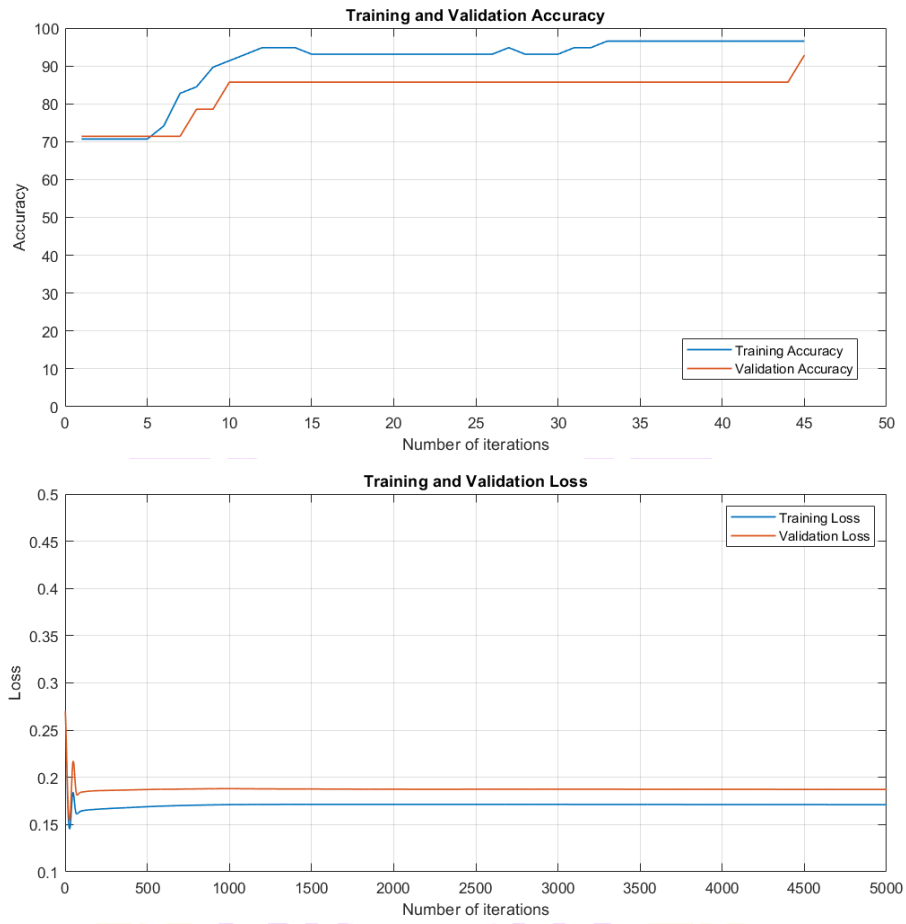
(Sigmoid-SVM). Extensive research consistently demonstrates the progressive success of these models over time. Machine learning finds applications in diverse domains, including medicine [27, 47], meteorology [7, 42], economy, and outlier detection [29, 26, 28].

The performance results of methods in previous literature using a shared dataset were analyzed. Accuracy, precision, recall, and F1-score were the evaluation metrics used in the studies referenced in [36, 2, 65] and the present study, with a division of data into training and testing sets. In contrast, the studies mentioned in [77, 38, 30] employed k-fold cross-validation splits the data into several parts, k number of parts, known as k-fold, where one part is validation data, and the rest is training data and then iterates through k-cycles, while the survey in [55] used the entire dataset. The outcomes from these different approaches are presented in Table 21.

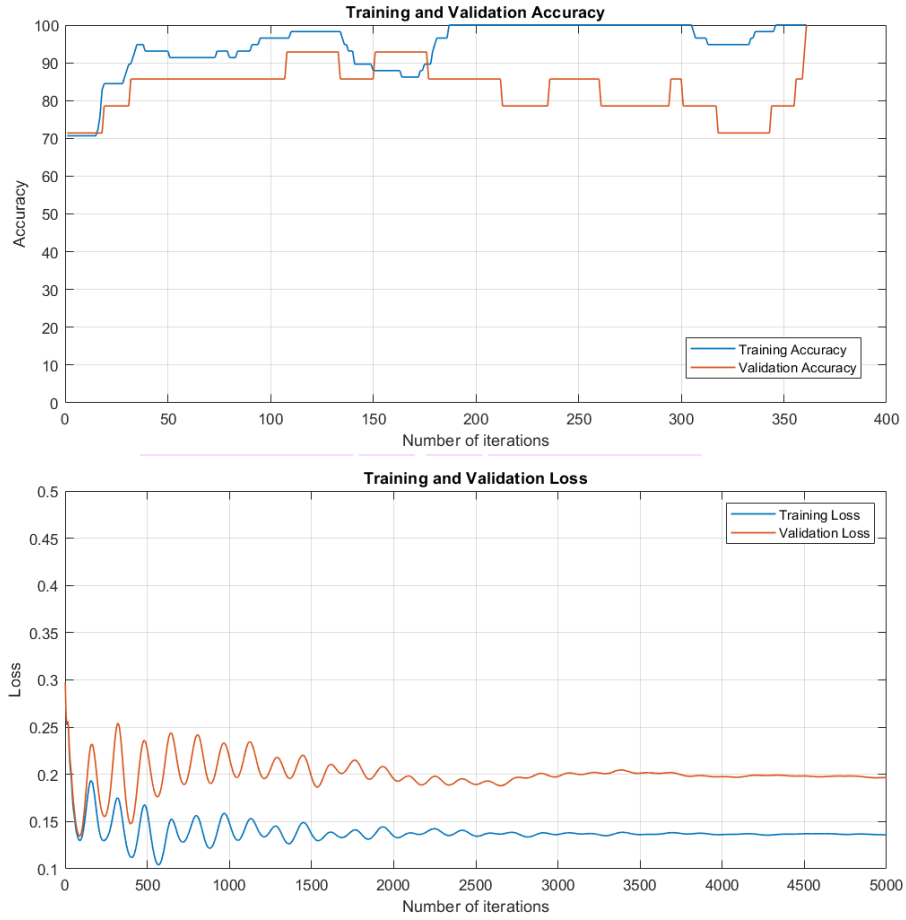
Table 21: Comparison of the performance with the literature.

Paper	Validation	Method	Precision	Recall	F1-score	Accuracy
Machmud and Wijaya [55]	All data	Logistic regression	80.00	76.19	78.05	87.50
		Naive Bayes	89.47	80.95	85.00	91.67
Tarakci and Ozkan [77]	k-fold cross validation, k = 10	kNN	82.25	100.00	90.26	84.70
		WkNN	86.44	100.00	92.72	88.90
Gamara et al. [36]	train (70%), validation (15%), and test (15%)	Artificial Neural Networks	100.00	80.00	88.89	90.91
Akter et al. [2]	train (80%), test (20%)	Decision Tree	100.00	90.91	95.24	93.33
		Random Forest	91.67	100.00	95.65	93.33
		XGBoost	93.33	100.00	96.55	93.33
Ghanem et al. [38]	k-fold cross validation, k=5	SIA	78.33	73.01	75.58	80.56
Ratul et al. [65]	train (80%), test (20%)	MLP	90.91	100.00	95.24	93.33
Degirmenci [30]	k-fold cross validation, k=10	Sigmoid-SVM	90.93	84.10	85.65	92.74
This study	train (80%), test (20%)	Algorithm 4.3.1 (RLSP-L_2)	100.00	100.00	100.00	100.00
		Algorithm 4.3.1 (RLSP-CL_2)	100.00	100.00	100.00	100.00
		Algorithm 4.3.1 (CLSP-L_2)	100.00	100.00	100.00	100.00

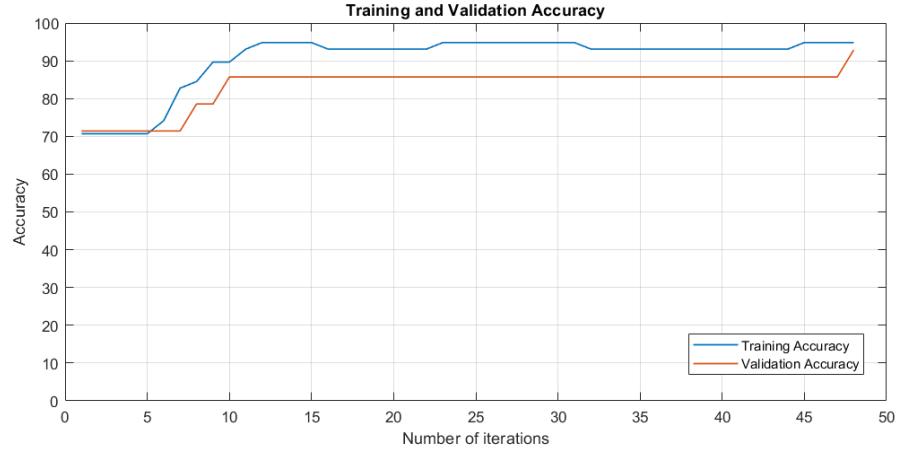
From Table 21, we see that the method studied is efficient in terms of accuracy, precision, recall, and F1-score the highest, which is the most accurate predictor of cervical cancer.



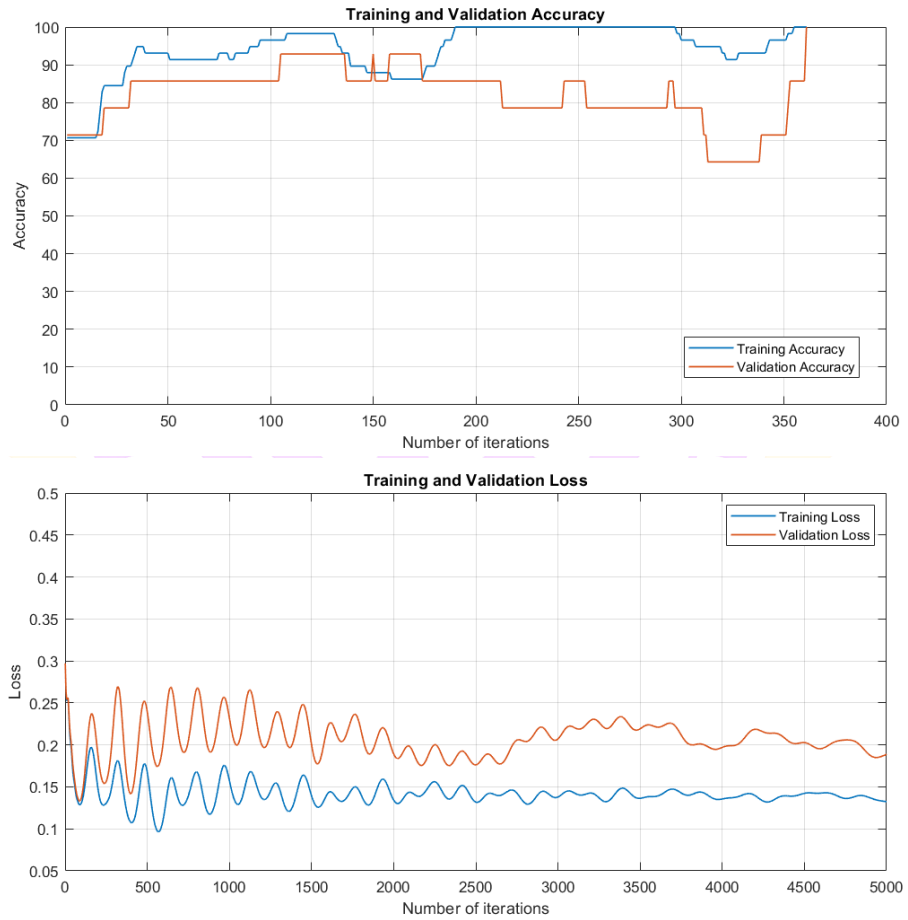
Figures 10-11: Accuracy and Loss plots of the iteration of RLSP-L_1.



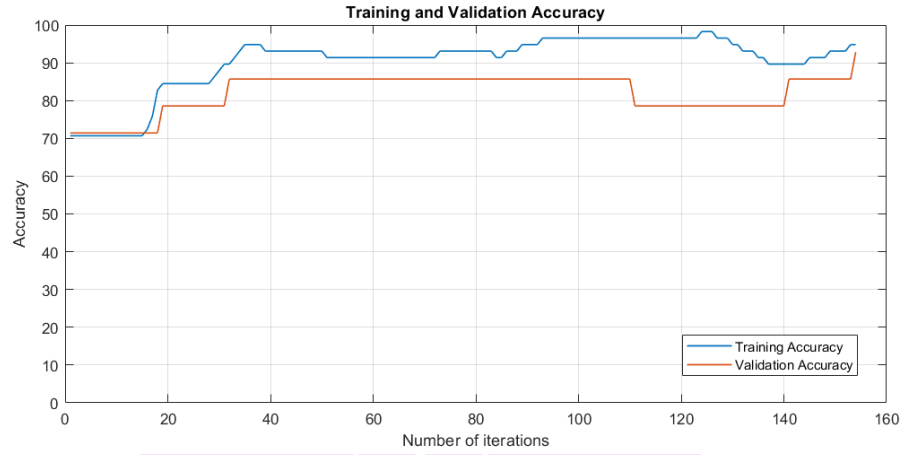
Figures 12-13: Accuracy and Loss plots of the iteration of RLSP-L_2.

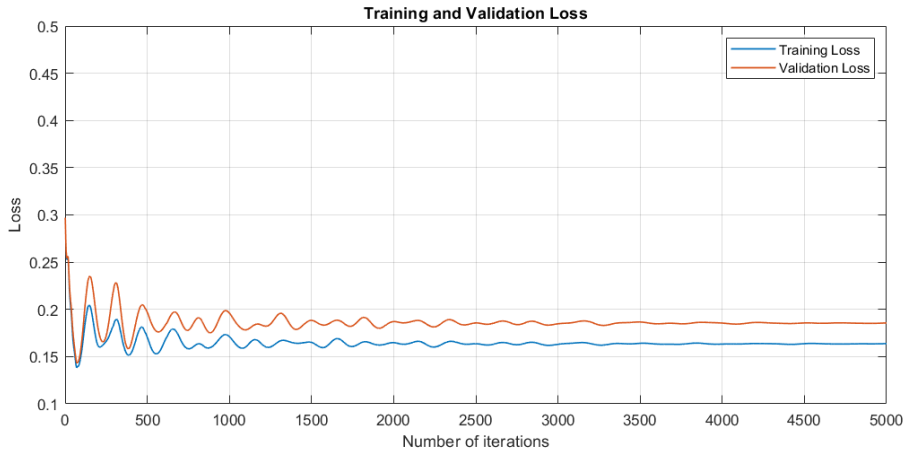


Figures 14-15: Accuracy and Loss plots of the iteration of RLSP-CL_1.

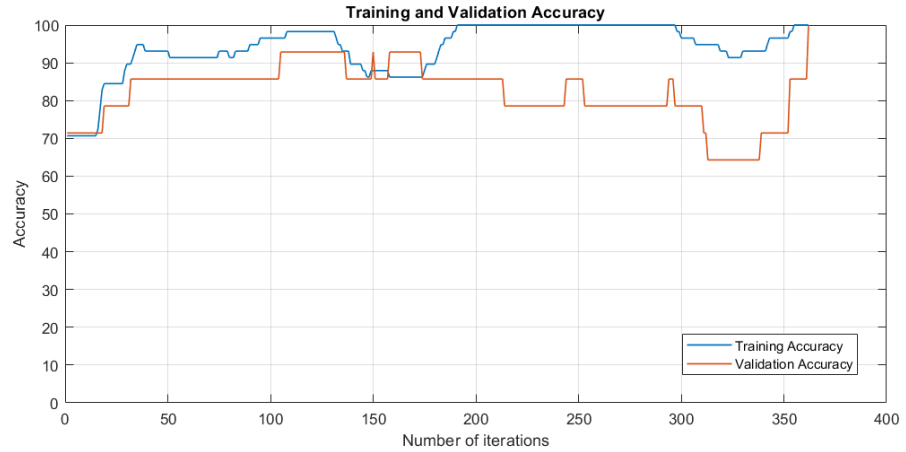


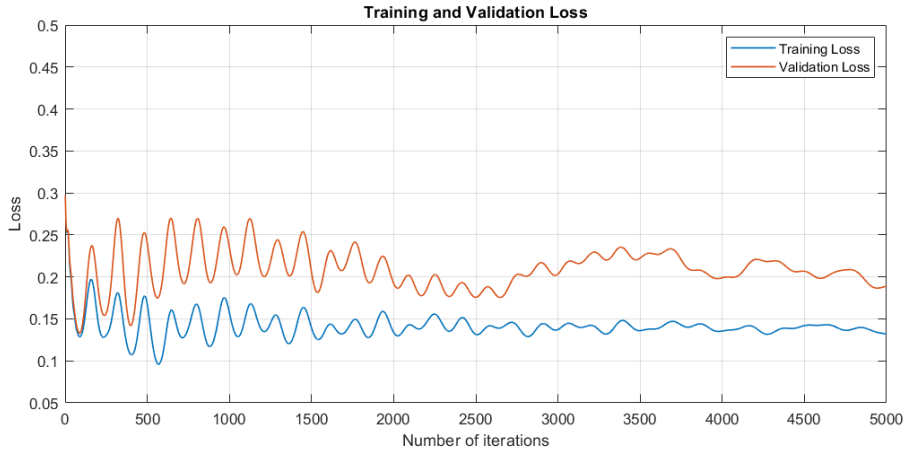
Figures 16-17: Accuracy and Loss plots of the iteration of RLSP-CL_2.





Figures 18-19: Accuracy and Loss plots of the iteration of CLSP-L_1.





Figures 20-21: Accuracy and Loss plots of the iteration of CLSP-L_2.

From Figures 20-21, we can see that the Accuracy tends to increase when there are more Train Models. The gap between Training Accuracy and Validation Accuracy is not much different, and the Loss will continue to decrease until it remains stable at some point. There is a slight gap between Training Loss and Validation Loss. This learning curve model shows that it is a model with good learning.

5.2 Application to Signal Recovery

This section is devoted to the presentation of numerical experiments which illustrate the application to signal recovery problem in compressed sensing that involves several blurring filters. A signal recovery problem can be modelled as the following underdetermined linear equation system:

$$b = Ax + \nu, \tag{5.2.1}$$

where $x \in \mathbb{R}^N$ is the original signal, $b \in \mathbb{R}^M$ is the observed signal with noise ν, and $A \in \mathbb{R}^{M \times N}$ $(M < N)$ is filter matrix. It is well known that the problem

(5.2.1) to equivalent to solving the following regularized least squares problem:

$$\min_{x\in\mathbb{R}^N} \frac{1}{2}\|Ax-b\|_2^2 + \lambda\|x\|_1, \tag{5.2.2}$$

where $\lambda > 0$ is a parameter. Problem (5.2.2) can be seen as problem (2.1.1) through the following settings: $F(x) = \nabla(\frac{1}{2}\|Ax-b\|_2^2)$ and $G(x) = \partial\|x\|_1$.

For the experiments in this section, we choose the signal size to be $N = 1024$ and $M = 512$ and the original signal x is generated by the uniform distribution in $[-2, 2]$ with m nonzero elements. We use the mean-squared error to measure the restoration accuracy defined as $MSE_n = \frac{1}{N}\|x_n - x\|_2^2 < 10^{-4}$. We choose all of parameters of Algorithm 4.1.1, Algorithm 4.1.3, FISTA and VIFBA as in Tables 7-9.

The original signal and the measurement by using A with $m = 100$ are shown in the Figure 16. Given that the initial points x_0, x_1 are generated by command rand (N, 1). The comparison are presented next.

Table 22: Numerical comparison of four algorithms.

Algorithm	**Iteration No.**	**Elapsed Time(s)**
FISTA	254	0.1516
VIFBA	163	0.0809
Algorithm 4.1.1	145	0.0759
Algorithm 4.1.3	122	0.4084

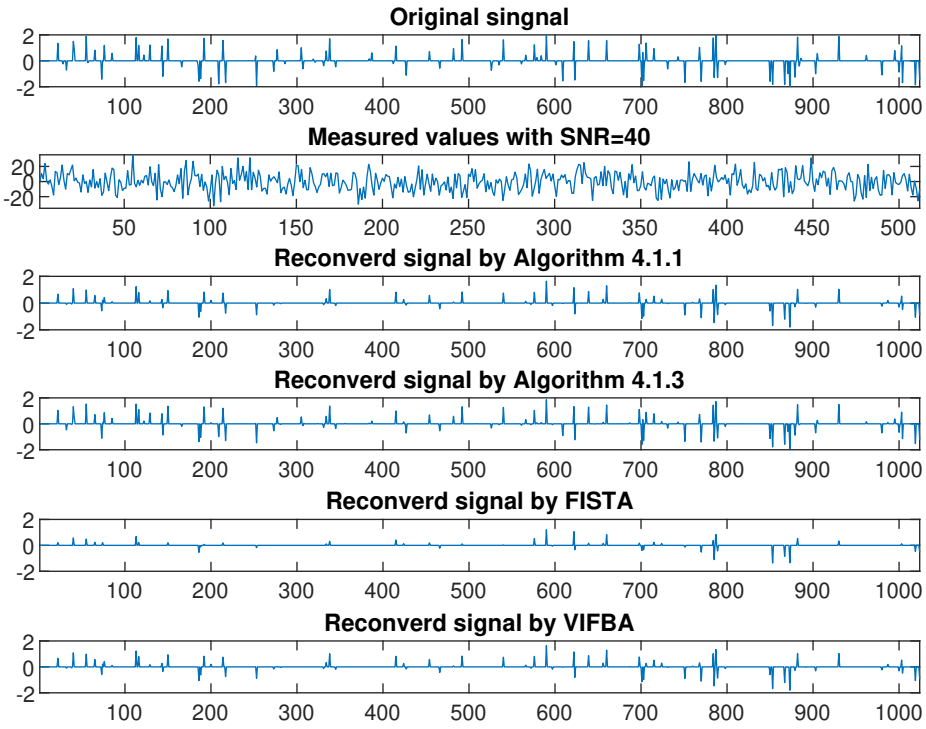


Figure 16: The original signal and the measurement and the reconstructed signals by using each input for m = 100.

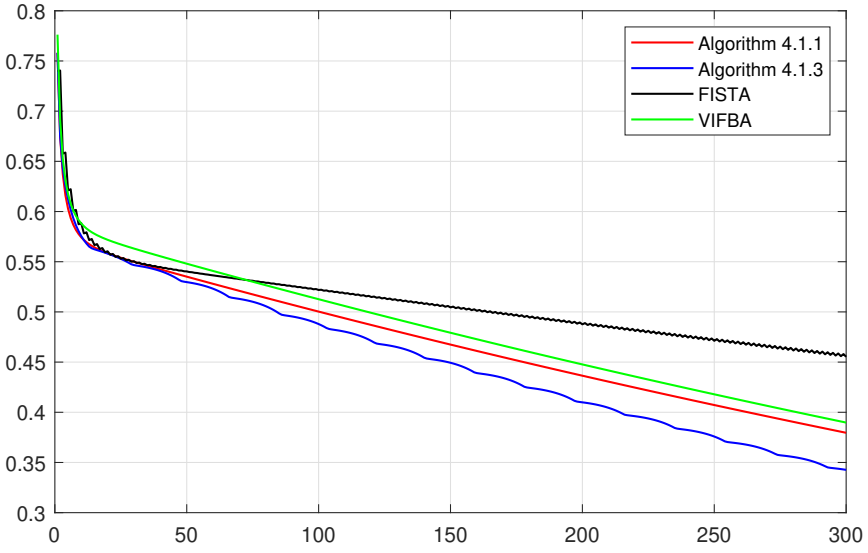


Figure 17: The mean-squared error versus number of iterations for m = 100.

From Figures 16-17, we see that the performance of our Algorithm 4.1.1 and Algorithm 4.1.3 is obviously better than FISTA and VIFBA with respect to the mean-squared error is rather stable at the end in Algorithm 4.1.1 and Algorithm 4.1.3 is better.

5.3 Application to Image Recovery

The image restoration problem is the recovering process of a degraded version which is a blurred and noisy image. This problem can be formulated in the linear equation system as follows:

$$b = Bx + \upsilon, \tag{5.3.1}$$

where $x \in \mathbb{R}^{n\times 1}$ is an original image, $b \in \mathbb{R}^{m\times 1}$ is the unknown image, υ is additive noise and $B \in \mathbb{R}^{m\times n}$ is the blurring operation. The main goal of image restoration problem (5.3.1) is to find the original image x. In some case, finding $x = B^{-1}(b - \upsilon)$ maybe a difficult task, thus finding the solution x by mean of convex minimization can overcome such difficulty, which is known as the following least squares (LS) problem

$$\min_x \frac{1}{2}\|b - Bx\|_2^2, \tag{5.3.2}$$

where $\|.\|$ is ℓ_2-norm defined by $\|x\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2}$. The solution of (5.3.2) can be estimated by many well known iteration methods [32, 40, 41, 84].

The main goal in digital image restoration is to find the unknown image that we don't know which one is the blurring matrix of this unknown image. This problem can be considered in the system of least squares problems:

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|B_1x - b_1\|_2^2, \min_{x\in\mathbb{R}^n} \frac{1}{2}\|B_2x - b_2\|_2^2, ..., \min_{x\in\mathbb{R}^n} \frac{1}{2}\|B_Nx - b_N\|_2^2 \tag{5.3.3}$$

where x is the original true image, B_i is the blurred matrix, b_i is the blurred image by the blurred matrix B_i for all $i = 1, 2, ..., N$. For solving (5.3.3), we can apply our main Algorithm 4.4.1 by setting $F_i x = B_i^T(B_i x - b_i)$ for all $x \in \mathbb{R}^n$ in Algorithm 4.4.1 since $B_i^T(B_i x - b_i)$ is Lipschitz continuous for each $i = 1, 2, ..., N$. This algorithm is generated as follows:

$$\begin{cases} t_n = x_n + \theta_n(x_n - x_{n-1}), \quad \forall n \geq 1, \\ y_n^i = P_C(t_n - \gamma_n^i B_i^T(B_i t_n - b_i)), \quad \forall n \geq 1 \text{ and } \forall i = 1, 2, ..., N, \\ (l_n^i \text{ is the smallest nonnegative integer such that} \\ \gamma_n^i \parallel B_i^T(B_i t_n - b_i) - B_i^T(B_i y_n^i - b_i) \parallel \leq \mu \parallel t_n - y_n^i \parallel), \\ z_n^i = P_{T_n^i}(t_n - \gamma_n^i B_i^T(B_i y_n^i - b_i)), \\ \bar{u}_n = \alpha_n^0(t_n) + \sum_{i=1}^{N} \alpha_n^i z_n^i, \ n \geq 1, \\ x_{n+1} = P_{C_{n+1}} x_1, \end{cases}$$

where $T_n^i = \{z \in H | \langle t_n - \gamma_n^i B_i t_n - y_n^i, z - y_n^i \rangle \leq 0\}$, $C_{n+1} = \{z \in C_n | \parallel \bar{u}_n - z \parallel \leq \parallel t_n - z \parallel\}$, $\rho, \mu, \alpha_n^i \in (0, 1)$ and $\{\theta_n\} \subseteq [0, \theta]$ for some $\theta \in [0, 1)$.

We will show the efficiency of our Algorithm 4.4.1 in image deblurring for the following three blur types:

Type 1: Gaussian blur of filter size 9×9 with standard deviation $\sigma = 4$ (blur matrix B_1).

Type 2: Out of focus blur (Disk) with radius $r = 6$ (blur matrix B_2).

Type 3: Motion blur specifying with motion length of 21 pixels (len = 21) and motion orientation 11° ($\theta = 11$) (blur matrix B_3).

The original Grey and RGB images are show in figures 18-19.

Figures 18-19: The original Grey and RGB image of sizes 276×490 and $280 \times 440 \times 3$, respectively.

The different types of blurred Grey and RGB images degraded by the blurring matrices B_1, B_2 and B_3 are shown in figures 20-25.

Gaussian Blurred Image

Gaussian Blurred Image

Out of Focus Blurred Image

Out of Focus Blurred Image

Motion Blurred Image

Motion Blurred Image

Figures 20-25: The degraded Grey and RGB images by blurred matrices B_1, B_2 and B_3, respectively.

We apply the PVSEGM amd our Algorithm 4.4.1 in getting the solution of deblurring problem with the three blurring matrices B_1, B_2, B_3. The results of the PVSEGM and our Algorithm 4.4.1 are considered in following seven cases:

Case I: Inputting B_1 on the PVSEGM and Algorithm 4.4.1.

Case II: Inputting B_2 on the PVSEGM and Algorithm 4.4.1.

Case III: Inputting B_3 on the PVSEGM and Algorithm 4.4.1.

Case IV: Inputting B_1 and B_2 on the PVSEGM and Algorithm 4.4.1.

Case V: Inputting B_1 and B_3 on the PVSEGM and Algorithm 4.4.1.

Case VI: Inputting B_2 and B_3 on the PVSEGM and Algorithm 4.4.1.

Case VII: Inputting B_1, B_2 and B_3 on the PVSEGM and Algorithm 4.4.1.

The following parameters are used for our algorithm:

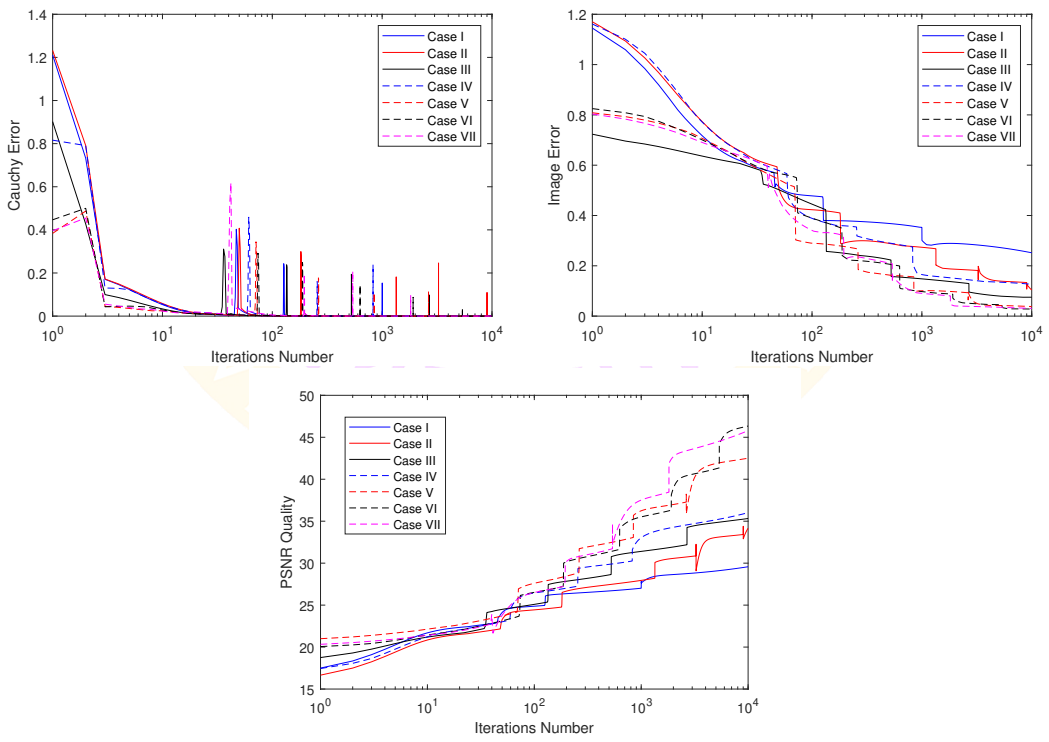
$$\theta_n = \begin{cases} 0.12 & \text{if } x_n \neq x_{n-1} \text{ and } n \leq 10,000 \\ \frac{1}{n^2\|x_n - x_{n-1}\|} & \text{if } x_n \neq x_{n-1} \text{ and } n > 10,000 \\ 0 & \text{Otherwise,} \end{cases}$$

$\rho = 0.5$ and $\mu = 0.35$. We choose $\mu = 0.95$, $\rho = 0.5$, $\alpha_n^0 = 1 - \frac{3n}{3n+1}$, $\alpha_n^1 = \frac{n}{3n+1}$, $\alpha_n^2 = \frac{n}{3n+1}$ and $\alpha_n^3 = 1 - \alpha_n^0 - \alpha_n^1 - \alpha_n^2$ for PVSEGM.

Table 23: Comparison of the number of iterations in Grey images.

Inputting	PSNR of 10000		Number of Iterations 33 PSNR	
	PVSEGM	Our Algorithm	PVSEGM	Our Algorithm
B_1	24.70720	29.57263	4921	50
B_2	26.47867	34.15647	2775	58
B_3	29.50780	35.32024	801	36
B_1, B_2	28.59585	36.01784	975	60
B_1, B_3	32.37244	42.50473	446	62
B_2, B_3	33.47745	46.33505	538	73
B_1, B_2, B_3	34.41830	45.79034	411	52

Moreover, the Cauchy error, the figure error and the peak signal-to-noise ratio (PSNR) for recovering processes of the degraded Grey images by using the proposed method within the first 10000 iterations are shown in figures 26-28.

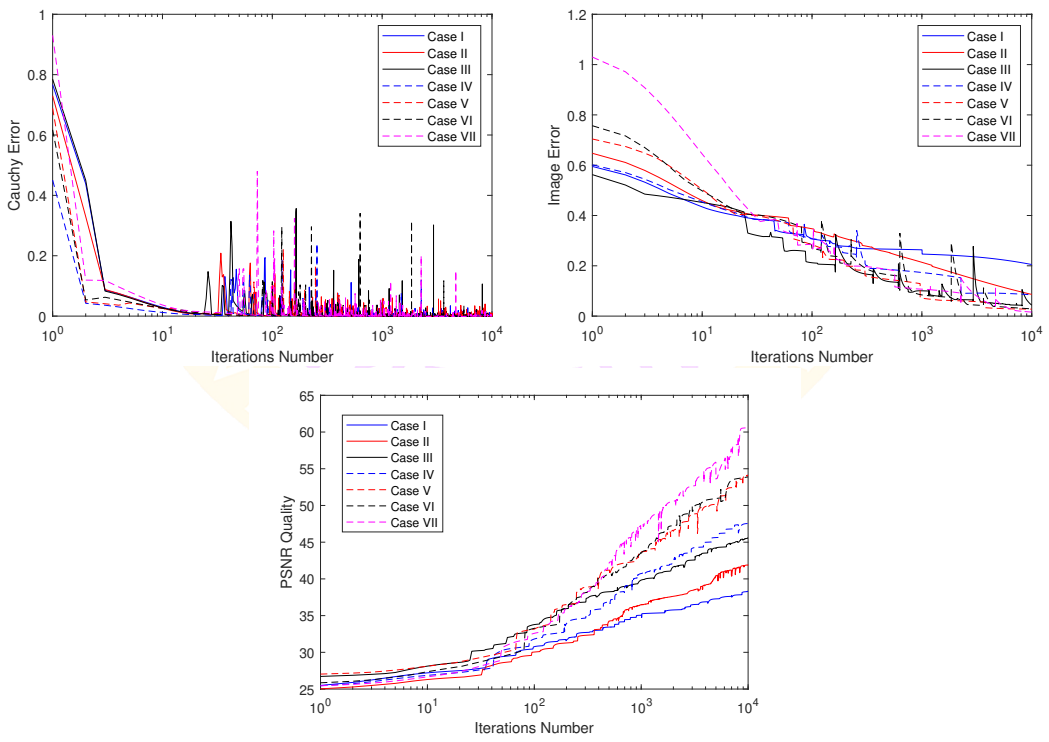


Figures 26-28: Cauchy error, Figure error and PSNR quality plots of the proposed iteration in all cases of Grey images.

Table 24: Comparison of the number of iterations in RGB images.

Inputting	PSNR of 10000		Number of Iterations 33 PSNR	
	PVSEGM	Our Algorithm	PVSEGM	Our Algorithm
B_1	33.47997	38.31203	6816	385
B_2	34.13544	41.83745	5800	364
B_3	37.89834	45.57931	1014	86
B_1, B_2	37.46071	47.54648	1253	190
B_1, B_3	41.57133	54.15965	509	86
B_2, B_3	41.77308	53.88841	634	87
B_1, B_2, B_3	43.52842	60.59668	474	122

Moreover, the Cauchy error, the figure error and the peak signal-to-noise ratio (PSNR) for recovering processes of the degraded RGB images by using the proposed method within the first 10000 iterations are shown in figures 29-31.



Figures 29-31: Cauchy error, Figure error and PSNR quality plots of the proposed iteration in all cases of RGB images.

The figures of deblurring when the 10000 iterations is the stopping criterion are shown in figures 33-46 that be composed of the restored image and its PSNR.

Figures 33-38: The reconstructed Grey and RGB images with their PSNR for Case I - Case III being used our Algorithm 4.4.1 presented in 10000 iterations respectively.

It can be seen from figures 39-44 that the quality of restored image by using our Algorithm 4.4.1 in solving the common solutions of deblurring problem (VIP) with ($N = 2$) has improved compared with the previous result on figures 33-38.

Figures 39-44: The reconstructed Grey and RGB images with their PSNR for Case IV - Case VI used our Algorithm 4.4.1 presented in 10000 iterations respectively.

Finally, the common solution of deblurring problem (VIP) with $(N = 3)$ by using the proposed algorithm is also tested (Inputting B_1, B_2 and B_3 on the proposed algorithm).

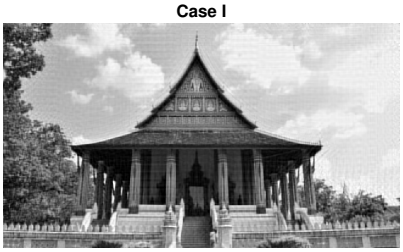
PSNR = 45.79034

PSNR = 60.59668

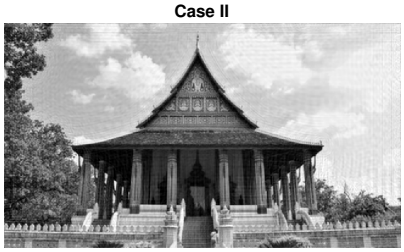
Figures 45-46: The reconstructed Grey and RGB images from the blurring operators B_1, B_2 and B_3 (Case VII) being used our Algorithm 4.4.1 presented in 10000 iterations, respectively.

Figures 45-46 show the reconstructed Grey and RGB images with thousand iteration. It has been found that the quality (PSNR) of the recovered Grey and RGB images obtained by this algorithm is highest compared to the previous two algorithm.

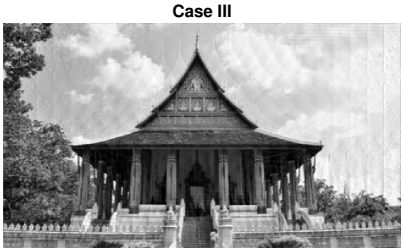
The figures of deblurring when the 33 PSNR is the stopping criterion are shown in figures 47-60 that be composed of the restored image and its number of iterations.



PSNR = 29 (10000 Iteration)

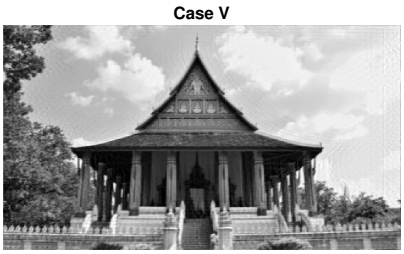


PSNR = 29 (1343 Iteration)

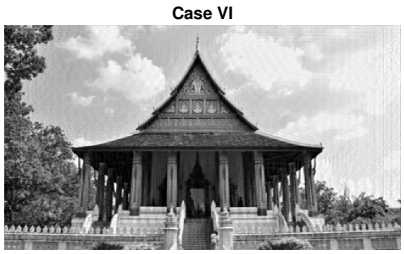


PSNR = 29 (524 Iteration)

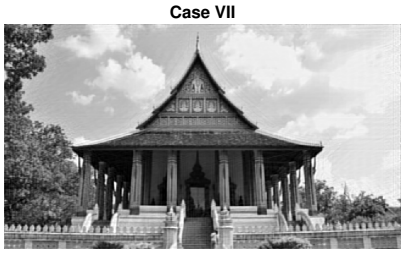
PSNR = 29 (256 Iteration)



PSNR = 29 (262 Iteration)



PSNR = 29 (188 Iteration)



PSNR = 29 (195 Iteration)

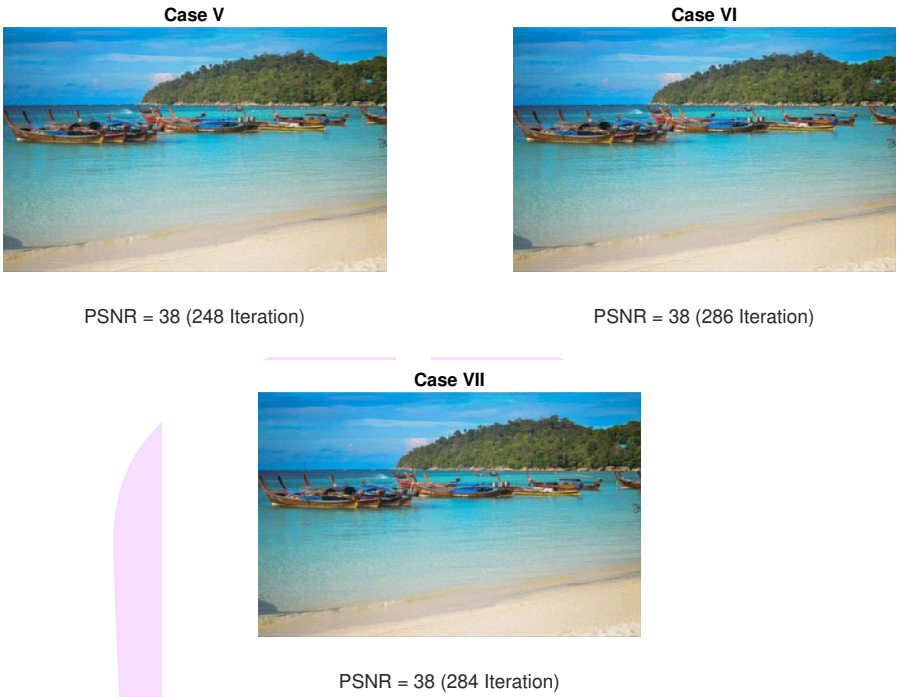
Figures 47-53: The reconstructed Grey images of all cases being used our Algorithm 4.4.1 with PSNR = 29.

PSNR = 38 (10000 Iteration)

PSNR = 38 (2693 Iteration)

PSNR = 38 (414 Iteration)

PSNR = 38 (604 Iteration)



Figures 54-60: The reconstructed RGB images of all cases being used our Algorithm 4.4.1 with PSNR = 38.

CHAPTER 6

CONCLUSIONS

The following results are all main theorems of this thesis:

Algorithm 6.1.1 Inertial Mann forward-backward splitting algorithm (IMFBSA)

Initialization: *Select* $x_0, x_1 \in H, \{\alpha_n\} \subset (0,1), \{\gamma_n\} \subset (0, 2\beta)$ *and* $\{\theta_n\} \subset [0, \infty)$ *satisfies the condition such that*

$$0 < \liminf_{n\to\infty} \gamma_n \leq \limsup_{n\to\infty} \gamma_n < 2\beta \text{ and } \sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty.$$

Iterative step: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

$$y_n = x_n + \theta_n(x_n - x_{n-1}),$$

and

$$z_n = y_n + \alpha_n(x_n - y_n).$$

Step 2. *Compute*

$$x_{n+1} = J^G_{\gamma_n}(I - \gamma_n F)z_n.$$

Replace n *by* $n+1$ *and then repeat* ***Step 1.***

Theorem 6.1.2 *The sequence* $\{x_n\}$ *generated by IMFBSA weakly converges to a solution of* $(F+G)^{-1}(0)$.

Algorithm 6.1.3 Initialization: *Select* $x_0, x_1 \in H, \{\alpha_n\} \subset (0,1), \mu_1, \mu_2, \mu_3 \in (0,2), \{\gamma_1\} \in (0, 2\beta)$ *and* $\{\theta_n\} \subset [0, \infty)$ *satisfies the condition such that*

$$\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty.$$

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

$$y_n = x_n + \theta_n(x_n - x_{n-1}),$$

and

$$z_n = y_n + \alpha_n(x_n - y_n).$$

Step 2. *Compute*

$$x_{n+1} = J^G_{\gamma_n}(I - \gamma_n F)z_n.$$

$$\gamma_{n+1} = \textbf{\textit{Stepsize UP}}\Big(\gamma_n, \frac{\|y_n - x_{n+1}\|}{\|F(y_n) - F(x_{n+1})\|}, \frac{\|z_n - x_{n+1}\|}{\|F(z_n) - F(x_{n+1})\|}, \frac{\|z_n - y_n\|}{\|F(z_n) - F(y_n)\|}, \mu_1, \mu_2, \mu_3\Big).$$

Replace n *with* $n+1$ *and then repeat* ***Step 1****.*

Lemma 6.1.4 *The sequence* $\{\gamma_n\}$ *generated by* ***Stepsize UP*** *in Algorithm 6.1.3 is a nonincreasing sequence and*

$$\lim_{n\to\infty} \gamma_n = \gamma \geq \min\{\gamma_1, \mu_1\beta, \mu_2\beta, \mu_3\beta\}.$$

Algorithm 6.1.5 Modified inertial forward-backward splitting algorithm (MIFBSA)

Initialization: *Select* $x_0, x_1 \in H$, $\{\gamma_n\} \subset (0, 2\beta)$, $\{\theta_n\} \subset [0, \theta]$ *for some*

$\theta \in [0,1)$, $\{\alpha_n\}$ *and* $\{\beta_n\}$ *are sequences in* $[0,1]$.

Iterative step: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Compute*

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

Step 2. *Compute*

$$z_n \quad = (1-\beta_n)x_n + \beta_n J^G_{\gamma_n}(I - \gamma_n F)y_n.$$

Step 3. *Compute*

$$x_{n+1} \quad = (1-\alpha_n)J^G_{\gamma_n}(I - \gamma_n F)x_n + \alpha_n J^G_{\gamma_n}(I - \gamma_n F)z_n.$$

where $J^G_{\gamma_n} = (I + \gamma_n F)^{-1}$. *Set* $n = n+1$ *and return to* **Step 1**.

Theorem 6.1.6 *The sequence* $\{x_n\}$ *generated by Algorithm 6.1.5 weakly converges to a solution of* $(F+G)^{-1}(0)$. *Assume that the following conditions hold:*

1. $\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty$;
2. $\liminf_{n\to\infty} \alpha_n > 0$ *and* $\liminf_{n\to\infty} \beta_n > 0$;
3. $0 < \liminf_{n\to\infty} \gamma_n \leq \limsup_{n\to\infty} \gamma_n < 2\beta$.

Algorithm 6.1.7 Initialization: *Select* $x_0, x_1 \in H$, $\gamma \in (0, 2\beta]$, $\delta \in (0,1)$, $\{\theta_n\} \subset [0,\theta]$ *for some* $\theta \in [0,1)$, $\{\alpha_n\}$ *and* $\{\beta_n\}$ *are sequences in* $[0,1]$ *and* N *is a stop number of iteration.*

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

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

Step 2. *Compute*

$$z_n = (1-\beta_n)x_n + \beta_n J^G_{\gamma_n}(I-\gamma_n F)y_n.$$

Step 3. *Compute*

$$x_{n+1} = (1-\alpha_n)J^G_{\gamma_n}(I-\gamma_n F)x_n + \alpha_n J^G_{\gamma_n}(I-\gamma_n F)z_n.$$

$$\gamma_{n+1} = \begin{cases} \textbf{\textit{Stepsize}}\ \boldsymbol{N}(y_n, x_n, z_n, \mu, \delta), & \text{if } 1 \le n < N, \\ \gamma_N, & \text{otherwise}, \end{cases}$$

where $J^G_{\gamma_n} = (I+\gamma^G_n)^{-1}$. *Replace* n *with* $n+1$ *and then repeat* ***Step 1***.

Theorem 6.1.8 *The sequence* $\{x_n\}$ *generated by Algorithm 6.1.7 weakly converges to* $z \in (F+G)^{-1}(0)$. *Assume that the following conditions hold:*

1. $\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty$;
2. $\liminf_{n\to\infty} \alpha_n > 0$ *and* $\liminf_{n\to\infty} \beta_n > 0$.

Algorithm 6.1.9 New projection algorithm (NPA)

Initialization: *Select* $x_0, x_1 \in H$, $\{\gamma_n\} \subset (0, 2\beta)$, $\{\theta_n\} \subset [0, \infty)$, $\{\eta_n\}$ *and* $\{\alpha_n\}$ *are sequences in* $(0, 1)$.

Iterative step: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Define*

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

Step 2. *Compute*

$$z_n = y_n + \eta_n(x_n - y_n).$$

Step 3. *Compute*

$$x_{n+1} = P_C(\alpha_n y_n + (1-\alpha_n)J^G_{\gamma_n}(I-\gamma_n F)z_n).$$

Replace n *by* $n+1$ *and then repeat* ***Step 1.***

Theorem 6.1.10 *Assume that the following conditions hold:*

1. $\sum_{n=1}^{\infty} \theta_n \|x_n - x_{n-1}\| < \infty$;
2. $0 < \liminf_{n\to\infty} \gamma_n \leq \limsup_{n\to\infty} \gamma_n < 2\beta$;
3. $\limsup_{n\to\infty} \alpha_n > 1$.

Then the sequence $\{x_n\}$ *weakly converges to* $z \in (F+G)^{-1}(0) \cap C$.

Algorithm 6.1.11 Hybrid inertial parallel subgradient extragradient-line algorithm (HIPSEA)

Initialization: *Take* $\rho \in (0,1)$, $\mu \in (0,1)$. *Select arbitrary points* $x_0, x_1 \in H$ *and* $\{\theta_n\} \subseteq [0,\theta]$ *for some* $\theta \in [0,1)$. *Set* $n := 1$.

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Compute*

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

Step 2. *Compute* y^i_n *for all* $i = 1, 2, ..., N$ *by*

$$y^i_n = P_C(t_n - \lambda^i_n F_i t_n),$$

where $\lambda^i_n = \rho^{k^i_n}$ *and* k^i_n *is the smallest nonnegative integer such that*

$$\lambda^i_n \parallel F_i t_n - F_i y^i_n \parallel \leq \mu \parallel t_n - y^i_n \parallel . \tag{6.1.4}$$

Step 3. *Compute*

$$z_n^i = P_{T_n^i}(t_n - \lambda_n^i F_i y_n^i),$$

$$\text{where } T_n^i := \{z \in H : \langle t_n - \lambda_n^i F_i t_n - y_n^i, z - y_n^i \rangle \leq 0\}.$$

Step 4. *Compute*

$$\bar{u}_n = \alpha_n^0(t_n) + \sum_{i=1}^{N} \alpha_n^i z_n^i,$$

$$\text{where } \alpha_n^i \in (0,1), \ \ \forall i = 1, 2, ..., N \text{ and } \sum_{i=0}^{N} \alpha_n^i = 1, \ \forall n \in N.$$

Step 5. *Compute*

$$x_{n+1} = P_{C_{n+1}} x_1,$$

$$\text{where } C_{n+1} := \{z \in C_n : \| \bar{u}_n - z \| \leq \| t_n - z \|\}.$$

Set $n + 1 \to n$ *and go to* **Step 1**.

Lemma 6.1.12 *There exists a nonnegative integer* k_n^i *satisfying (6.1.4).*

Theorem 6.1.13 *Assume that the conditions hold:*

1. $\sum_{n=1}^{\infty} \theta_n \| x_n - x_{n-1} \| < \infty$.
2. $\liminf_{n\to\infty} \alpha_n^i > 0$ *for all* $i = 1, 2, ..., N$.

Then the sequence $\{x_n\}$ *generated by Algorithm 6.1.11 converges strongly to* $z \in \Upsilon$.

Algorithm 6.1.14 Initialization: *Take* $\rho \in (0,1)$, $\mu \in (0,1)$. *Select arbitrary points* $x_0, x_1 \in H$ *and* $\{\theta_n\} \subseteq [0, \theta]$ *for some* $\theta \in [0,1)$. *Set* $n := 1$.

Iterative Steps: *Construct* $\{x_n\}$ *by using the following steps:*

Step 1. *Compute*

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

Step 2. *Compute* y_n *by*

$$y_n = P_C(t_n - \lambda_n F t_n),$$

where $\lambda_n = \rho^{k_n}$ *and* k_n *is the smallest nonnegative integer such that*

$$\lambda_n \parallel Ft_n - Fy_n \parallel \leq \mu \parallel t_n - y_n \parallel . \tag{6.1.5}$$

Step 3. *Compute*

$$z_n = P_{T_n}(t_n - \lambda_n F y_n),$$

where $T_n := \{z \in H : \langle t_n - \lambda_n F t_n - y_n, z - y_n \rangle \leq 0\}$.

Step 4. *Compute*

$$\bar{u}_n = \alpha_n t_n + (1 - \alpha_n) z_n,$$

where $\alpha_n \in (0, 1)$.

Step 5. *Compute*

$$x_{n+1} = P_{C_{n+1}} x_1,$$

where $C_{n+1} := \{z \in C_n : \parallel \bar{u}_n - z \parallel \leq \parallel t_n - z \parallel\}$.

Set $n + 1 \to n$ *and go to* **Step 1**.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Agarwal, R. P., O'Regan, D., and Sahu, D. R. (2009). Fixed point theory for Lipschitzian-type mappings with applications. **New York: Springer**.

[2] Akter, L., Islam, M. M., Al-Rakhami, M. S., and Haque, M. R. (2021). Prediction of cervical cancer from behavior risk using machine learning techniques. **SN Computer Science**, 2, 1-10.

[3] Alvarez, F., and Attouch, H. (2001). An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping. **Set-Valued Analysis**, 9, 3-11.

[4] Alvarez, F. (2004). Weak convergence of a relaxed and inertial hybrid projection-proximal point algorithm for maximal monotone operators in Hilbert space. **SIAM Journal on Optimization**, 14(3), 773-782.

[5] Anh, P. K., and Van Hieu, D. (2016). Parallel hybrid iterative methods for variational inequalities, equilibrium problems, and common fixed point problems. **Vietnam Journal of Mathematics**, 44, 351-374.

[6] Antipin, A. S. (1978). Method of convex programming using a symmetric modification of Lagrange function. **Matekon**, 14(2), 23-38.

[7] Apaydin, M., Yumuş, M., Değirmenci, A., and Karal, Ö. (2022). Evaluation of air temperature with machine learning regression methods using Seoul City meteorological data. **Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi**, 28(5), 737-747.

[8] Auslender, A., Teboulle, M., and Ben-Tiba, S. (1999). A logarithmic-quadratic proximal method for variational inequalities. **Computational Optimization: A Tribute to Olvi Mangasarian**, 31-40.

[9] Bartle, R. G., and Sherbert, D. R. (2000). Introduction to real analysis (Vol. 2). **New York: Wiley**.

[10] Bauschke, H. H., and Borwein, J. M. (1996). On projection algorithms for solving convex feasibility problems. **SIAM review**, 38(3), 367-426.

[11] Bauschke, H. H. and Combettes, P. L. (2011). Convex analysis and monotone operator theory in Hilbert spaces. (Vol. 408). **New York: Springer**.

[12] Beck, A., and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. **SIAM journal on imaging sciences**, 2(1), 183-202.

[13] Bello Cruz, J. Y., and Nghia, T. T. (2016). On the convergence of the forward?backward splitting method with linesearches. **Optimization Methods and Software**, 31(6), 1209-1238.

[14] Breast Cancer Wisconsin (Original) Data Set, [Online]. Available: https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-cancer-wisconsin.data, accessed on: Aug. 25, 2017.

[15] Browder, F. E. (1965). Nonexpansive nonlinear operators in a Banach space. **Proceedings of the National Academy of Sciences**, 54(4), 1041-1044.

[16] Bruck, R. E., and Reich, S. (1977). Nonexpansive projections and resolvents of accretive operators in Banach spaces. **Houston J. Math**, 3(4), 459-470.

[17] Cegielski, A. (2012). Iterative methods for fixed point problems in Hilbert spaces (Vol. 2057). **Springer**.

[18] Ceng, L. C., Hadjisavvas, N., and Wong, N. C. (2010). Strong convergence theorem by a hybrid extragradient-like approximation method for variational inequalities and fixed point problems. **Journal of Global Optimization**, 46(4), 635-646.

[19] Censor, Y., Gibali, A., and Reich, S. (2011). Strong convergence of subgradient extragradient methods for the variational inequality problem in Hilbert space. **Optimization Methods and Software**, 26(4-5), 827-845.

[20] Censor, Y., Gibali, A., and Reich, S. (2011). The subgradient extragradient method for solving variational inequalities in Hilbert space. **Journal of Optimization Theory and Applications**, 148(2), 318-335.

[21] Chen, P., Huang, J., and Zhang, X. (2013). A primal-dual fixed point algorithm for convex separable minimization with applications to image restoration. **Inverse Problems**, 29(2), 025011.

[22] Chidume, C. E., and Ezeora, J. N. (2014). Krasnoselskii-type algorithm for family of multi-valued strictly pseudo-contractive mappings. **Fixed Point Theory and Applications**, 2014(1), 1-7.

[23] Cholamjiak, W., Cholamjiak, P., and Suantai, S. (2018). An inertial forward?backward splitting method for solving inclusion problems in Hilbert spaces. **Journal of Fixed Point Theory and Applications**, 20, 1-17.

[24] Combettes, P. L., and Wajs, V. R. (2005). Signal recovery by proximal forward-backward splitting. **Multiscale modeling and simulation**, 4(4), 1168-1200.

[25] Dang, Y., Sun, J., and Xu, H. (2017). Inertial accelerated algorithms for solving a split feasibility problem. **Journal of Industrial and Man-**

agement Optimization, 13(3), 1383-1394.

[26] Degirmenci, A., and Karal, O. (2022). Efficient density and cluster based incremental outlier detection in data streams. **Information Sciences**, 607, 901-920.

[27] Degirmenci, A., and Karal, O. (2018). Evaluation of kernel effects on svm classification in the success of wart treatment methods. **Am. J. Eng. Res**, 7, 238-244.

[28] Degirmenci, A., and Karal, O. (2022). iMCOD: Incremental multi-class outlier detection model in data streams. **Knowledge-Based Systems**, 258, 109950.

[29] Degirmenci, A., and Karal, O. (2021). Robust incremental outlier detection approach based on a new metric in data streams. **IEEE Access**, 9, 160347-160360.

[30] Degirmenci, A. Performance Comparison of kNN, Random Forest and SVM in the Prediction of Cervical Cancer from Behavioral Risk. vol, 7, 71-79.

[31] Dong, Q. L., Yuan, H. B., Cho, Y. J., and Rassias, T. M. (2018). Modified inertial Mann algorithm and inertial CQ-algorithm for nonexpansive mappings. **Optimization Letters**, 12, 87-102.

[32] Engl, H. W., Hanke, M., and Neubauer, A. (1996). Regularization of inverse problems (Vol. 375). **Springer Science and Business Media**.

[33] Fang, Y. P., and Huang, N. J. (2003). H-monotone operator and resolvent operator technique for variational inclusions. **Applied mathematics and Computation**, 145(2-3), 795-803.

[34] Fang, C., and Chen, S. (2015). Some extragradient algorithms for variational inequalities. **Advances in variational and hemivariational inequalities: theory, numerical analysis, and applications**, 145-171.

[35] Frerking, J., and Westphal, U. (1989). On a property of metric projections onto closed subsets of Hilbert spaces. **Proceedings of the American Mathematical Society**, 105(3), 644-651.

[36] Gamara, R. P. C., Neyra, R. Q., and Recto, K. H. A. (2021, November). Behavior-Based Early Cervical Cancer Risk Detection Using Artificial Neural Networks. **In 2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)** (pp. 1-6). IEEE.

[37] Gibali, A., and Thong, D. V. (2018). Tseng type methods for solving inclusion problems and its applications. **Calcolo**, 55, 1-22.

[38] Ghanem, S., Couturier, R., and Gregori, P. (2021). An Accurate and Easy to Interpret Binary Classifier Based on Association Rules Using Implication Intensity and Majority Vote. **Mathematics**, 9(12), 1315.

[39] Goebel, K., and Kirk, W. A. (1990). Topics in metric fixed point theory (No. 28). **Cambridge university press**.

[40] Hansen, P. C. (1998). Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion. **Society for Industrial and Applied Mathematics**.

[41] Hansen, P. C. (2010). Discrete inverse problems: insight and algorithms. **Society for Industrial and Applied Mathematics**.

[42] Hatipoğlu, Ş ., Belgrat, M. A., Degirmenci, A., and Karal, Ö. (2021, October). Prediction of Unemployment Rates in Turkey by k-Nearest Neighbor Regression Analysis. **In 2021 Innovations in Intelligent Systems and Applications Conference (ASYU)** (pp. 1-5). IEEE.

[43] Ho, Y., and Wookey, S. (2019). The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling. **IEEE access**, 8, 4806-4813.

[44] Huang, G. B., Zhu, Q. Y., and Siew, C. K. (2006). Extreme learning machine: theory and applications. **Neurocomputing**, 70(1-3), 489-501.

[45] Inthakon, W., Suantai, S., Sarnmeta, P., and Chumpungam, D. (2020). A new machine learning algorithm based on optimization method for regression and classification problems. **Mathematics**, 8(6), 1007.

[46] Jitpeera, T., and Kumam, P. (2011). A new hybrid algorithm for a system of mixed equilibrium problems, fixed point problems for nonexpansive semigroup, and variational inclusion problem. **Fixed Point Theory and Applications**, 2011, 1-27.

[47] Karal, Ö. (2018). EKG verilerinin destek vektör regresyon yöntemiyle sıkıştırılması. **Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi**, 33(2), 743-756.

[48] Konnov, I. (2001). Combined relaxation methods for variational inequalities (Vol. 495). **Springer Science and Business Media**.

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

[50] Kreyszig, E. (1991). Introductory functional analysis with applications (Vol. 17). **John Wiley and Sons**.

[51] Liang, J., Luo, T., and Schonlieb, C. B. (2022). Improving "fast iterative shrinkage-thresholding algorithm": faster, smarter, and greedier. **SIAM Journal on Scientific Computing**, 44(3), A1069-A1091.

[52] Lions, P. L., and Mercier, B. (1979). Splitting algorithms for the sum of two nonlinear operators. **SIAM Journal on Numerical Analysis**, 16(6), 964-979.

[53] López, G., Martín-Márquez, V., Wang, F., and Xu, H. K. (2012, January). Forward-backward splitting methods for accretive operators in Banach spaces. **In Abstract and Applied Analysis**, (Vol. 2012). Hindawi.

[54] Lorenz, D. A., and Pock, T. (2015). An inertial forward-backward algorithm for monotone inclusions. **Journal of Mathematical Imaging and Vision**, 51, 311-325.

[55] Machmud, R., and Wijaya, A. (2016). Behavior determinant based cervical cancer early detection with machine learning algorithm. **Advanced Science Letters**, 22(10), 3120-3123.

[56] Martinet, B. (1970). Regularisation d'inequations variationelles par approximations successives. **Revue Francaise d'informatique et de Recherche operationelle**, 4, 154-159.

[57] Martinez-Yanes, C., and Xu, H. K. (2006). Strong convergence of the CQ method for fixed point iteration processes. **Nonlinear Analysis: Theory, Methods and Applications**, 64(11), 2400-2411.

[58] Moudafi, A., and Oliny, M. (2003). Convergence of a splitting inertial proximal method for monotone operators. **Journal of Computational and Applied Mathematics**, 155(2), 447-454.

[59] Nesterov, Y. (1983). A method for unconstrained convex minimization problem with the rate of convergence $O(1/k2)$. **In Dokl. Akad. Nauk. SSSR** (Vol. 269, No. 3, p. 543).

[60] Parikh, N., and Boyd, S. (2013). Proximal Algorithms, Found. **Trends Optim**. 3.

[61] Passty, G. B. (1979). Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. **Journal of Mathematical Analysis and Applications**, 72(2), 383-390.

[62] Pławiak, P., Abdar, M., and Acharya, U. R. (2019). Application of new deep genetic cascade ensemble of SVM classifiers to predict the Australian credit scoring. **Applied Soft Computing**, 84, 105740.

[63] Pławiak, P., Abdar, M., Pławiak, J., Makarenkov, V., and Acharya, U. R. (2020). DGHNL: A new deep genetic hierarchical network of learners for prediction of credit scoring. **Information Sciences**, 516, 401-418.

[64] Polyak, B. T. (1964). Some methods of speeding up the convergence of iteration methods. **Ussr computational mathematics and mathematical physics**, 4(5), 1-17.

[65] Ratul, I. J., Al-Monsur, A., Tabassum, B., Ar-Rafi, A. M., Nishat, M. M., and Faisal, F. (2022, May). Early risk prediction of cervical cancer: A machine learning approach. **In 2022 19th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)** (pp. 1-4). IEEE.

[66] Rockafellar, R. (1970). On the maximal monotonicity of subdifferential mappings. **Pacific Journal of Mathematics**, 33(1), 209-216.

[67] Rockafellar, R. T. (1976). Monotone operators and the proximal point algorithm. **SIAM journal on control and optimization**, 14(5), 877-898.

[68] Sahu, D. R., Cho, Y. J., Dong, Q. L., Kashyap, M. R., and Li, X. H. (2021). Inertial relaxed CQ algorithms for solving a split feasibility problem in Hilbert spaces. **Numerical Algorithms**, 87, 1075-1095.

[69] Stark, H. (Ed.). (2013). Image recovery: theory and application. **Elsevier**.

[70] Suantai, S. (2005). Weak and strong convergence criteria of Noor iterations for asymptotically nonexpansive mappings. **Journal of Mathematical Analysis and Applications**, 311(2), 506-517.

[71] Suantai, S., Peeyada, P., Yambangwai, D., and Cholamjiak, W. (2020). A parallel-viscosity-type subgradient extragradient-line method for finding the common solution of variational inequality problems applied to image restoration problems. **Mathematics**, 8(2), 248.

[72] Szaleniec, J., Wiatr, M., Szaleniec, M., Składzień, J., Tomik, J., Oleś, K., and Tadeusiewicz, R. (2013). Artificial neural network modelling of the results of tympanoplasty in chronic suppurative otitis media patients. **Computers in biology and medicine**, 43(1), 16-22.

[73] Szaleniec, M., Tadeusiewicz, R., and Witko, M. (2008). How to select an optimal neural model of chemical reactivity?. **Neurocomputing**, 72(1-3), 241-256.

[74] Takahashi, W. (2009). Introduction to nonlinear and convex analysis. **Yokohama Publishers**.

[75] Takahashi, W. (2000). Nonlinear functional analysis. **Fixed Point Theory and its Applications**.

[76] Tan, B., Zhou, Z., and Qin, X. (2020). Accelerated projection-based forward-backward splitting algorithms for monotone inclusion problems. arXiv preprint arXiv:2004.04326.

[77] Tarakci, F., and Ozkan, I. A. (2021). Comparison of classification performance of kNN and WKNN algorithms. **Selcuk University Journal of Engineering Sciences**, 20(2), 32-37.

[78] Thomas, T., Pradhan, N., and Dhaka, V. S. (2020, February). Comparative analysis to predict breast cancer using machine learning algorithms: a survey. **In 2020 International conference on inventive computation technologies (ICICT)** (pp. 192-196). IEEE.

[79] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. **Journal of the Royal Statistical Society Series B: Statistical Methodology**, 58(1), 267-288.

[80] Trmolieres, R., Lions, J. L., and Glowinski, R. (2011). Numerical analysis of variational inequalities. **Elsevier**.

[81] Tseng, P. (2000). A modified forward-backward splitting method for maximal monotone mappings. **SIAM Journal on Control and Optimization**, 38(2), 431-446.

[82] UCI Machine Learning Repository: Cervical cancer behavior risk data set. https://archive.ics.uci.edu/ml/datasets/Cervical+Cancer+Behavior+Risk

[83] Verma, M., Sahu, D. R., and Shukla, K. K. (2018). VAGA: a novel viscosity-based accelerated gradient algorithm: Convergence analysis and ap-

plications. **Applied Intelligence**, 48, 2613-2627.

[84] Vogel, C. R. (2002). Computational methods for inverse problems. **Society for Industrial and Applied Mathematics**.

[85] Wang, F., and Cui, H. (2012). On the contraction-proximal point algorithms with multi-parameters. **Journal of Global Optimization**, 54, 485-491.

[86] Wardhani, N. W. S., Rochayani, M. Y., Iriany, A., Sulistyono, A. D., and Lestantyo, P. (2019, October). Cross-validation metrics for evaluating classification performance on imbalanced data. **In 2019 international conference on computer, control, informatics and its applications (IC3INA)** (pp. 14-18). IEEE.

[87] Xiu, N., Wang, C., and Kong, L. (2007). A note on the gradient projection method with exact stepsize rule. **Journal of Computational Mathematics**, 221-230.

[88] Yao, Y., Kang, S. M., Jigang, W., and Yang, P. X. (2012). A regularized gradient projection method for the minimization problem. **Journal of Applied Mathematics**, 2012.

[89] Zhang, S. S., Lee, J. H., and Chan, C. K. (2008). Algorithms of common solutions to quasi variational inclusion and fixed point problems. **Applied Mathematics and Mechanics**, 29(5), 571-581.

BIOGRAPHY

BIOGRAPHY

Name Surname	Pronpat Peeyada
Date of Birth	March 27, 1996
Place of Birth	Chiang Rai Province, Thailand
Address	51 Moo 4, Pa Hung Sub - district, Phan District, Chiang Rai Province, Thailand 57120

Education Background

2023	Ph.D. (Mathematics), University of Phayao, Phayao, Thailand
2018	B.Sc. (Mathematics), University of Phayao,

Publications

Articles

1. **Peeyada, P.**, and Cholamjiak, W. (2023). Prediction of cervical cancer using a new projection algorithm for solving variational inclusion problems. Journal of Computational and Applied Mathematics.(Submitted)

2. **Peeyada, P.**, Dutta, H., Shiangjen, K., and Cholamjiak, W. (2023). A modified forward-backward splitting methods for the sum of two monotone operators with applications to breast cancer prediction. Mathematical Methods in the Applied Sciences, 46(1), 1251-1265.

3. **Peeyada, P.**, Suparatulatorn, R., and Cholamjiak, W. (2022). An inertial Mann forward-backward splitting algorithm of variational inclusion problems and its applications. Chaos, Solitons and Fractals, 158, 112048.

4. **Peeyada, P.**, Cholamjiak, W., and Yambangwai, D. (2022). A hybrid inertial parallel subgradient extragradient-line algorithm for variational inequalities with an application to image recovery, J. Nonlinear Funct. Anal., 2022 (2022), 9.

Others

1. Kesornprom, S., **Peeyada, P.**, Cholamjiak, W., Ngamkhum, T., and Jun-on, N. (2023). New Iterative Method with Inertial Technique for Split Variational Inclusion Problem to Classify TPACK Level of Pre-Service Mathematics Teachers. Thai Journal of Mathematics, 21(2), 351-365.

2. Suantai, S., Yajai, W., **Peeyada, P.**, Cholamjiak, W., and Chachvarat, P. (2023). A modified inertial viscosity extragradient type method for equilibrium problems application to classification of diabetes mellitus: Machine learning methods. AIMS Mathematics, 8(1), 1102-1126.

3. Suantai, S., **Peeyada, P.**, Fulga, A., and Cholamjiak, W. (2023). Heart disease detection using inertial Mann relaxed CQ algorithms for split feasibility problems. AIMS Mathematics, 8(8), 18898-18918.

4. Suantai, S., **Peeyada, P.**, Cholamjiak, W., and Dutta, H. (2022). Image deblurring using a projective inertial parallel subgradient extragradient-line algorithm of variational inequality problems. Filomat, 36(2), 423-437.

5. Kitisak, P., **Peeyada, P.**, Tewalok, N., Choiban, Y., Chaiwong, S., and Cholamjiak, W. (2021). Parallel Hybrid Extragradient-Proximal Methods for Split Equilibrium Problems in Hilbert

Spaces. Thai Journal of Mathematics, 19(3), 942-956.

6. Suantai, S., **Peeyada, P.**, Yambangwai, D., and Cholamjiak, W. (2020). A parallel-viscosity-type subgradient extragradient-line method for finding the common solution of variational inequality problems applied to image restoration problems. Mathematics, 8(2), 248.

Conference presentations

1. **Peeyada, P.** (August 2-5, 2023). An inertial Mann forward-backward splitting algorithm for variational inclusion problems with applications. The 11th Asian Conference on Fixed Point Theory and Optimization (ACFPTO).

2. **Peeyada, P.** (July 11-14, 2023). Solving variational inclusion problems using inertial Mann forward-backward splitting algorithm and its applications. The 14th International Conference on Fixed Point Theory and its Applications (ICFPTA), (Online).

3. **Peeyada, P.** (October 12-15, 2022). A modified forward-backward splitting methods for the sum of two monotone operators with applications. International Workshop on Nonlinear Analysis and its Applications (IWNAA), (Online).

4. **Peeyada, P.** (September 8-9, 2022). An image recovery using an inertial parallel subgradient extragradient-line algorithm for variational inequality problems. In International conference on Digital Image Processing and Machine Learning (ICDIPML), University of Phayao, Phayao, Thailand.

5. **Peeyada, P.** (August 29-September 1, 2022). A hybrid inertial parallel subgradient extragradient-line algorithm of variational inequality problems with applications. 11th International Eurasian Conference on Mathematical Sciences and Applications (IECMSA), Istanbul, Turkey.

6. **Peeyada, P.** (October 13-16, 2021). Solving variational inequalities problems using inertial parallel subgradient extragradient-line algorithm for application to image recovery. International Workshop on Nonlinear Analysis and its Applications (IWNAA), (Online).

7. **Peeyada, P.** (August 20-22, 2021). A hybrid inertial parallel subgradient extragradient-line algorithm for solving variational inequality problems with applications to image recovery. International Workshop on Digital Image Processing and Machine Learning (IWDM), (Online).

8. **Peeyada, P.** (April 28-30, 2021). Modified inertial parallel subgradient extragradient-line method for variational inequality problems with application to image deblurring. The 24^{th} International Annual Symposium on Computational Science and Engineering, (Online).

9. **Peeyada, P.** (December 22-23, 2020). An image deblurring using a projective inertial parallel subgradient extragradient-line algorithm of variational inequality problems. International Conference on Applied Nonlinear Analysis and Soft Computing, (Online).