FUPRE Journal

of

Scientific and Industrial Research

ISSN: 2579-1184(Print)
ISSN: 2578-1129 (Online)

http://fupre.edu.ng/journal

Inertial Residual Projection Method (IRPM) for Approximating Solutions of Variational Inequality Problems

EKUMA-OKEREKE, E.[1,*], ***OKUDU, G.***[2]

[1]*Department of Mathematics, Federal University of Petroleum Resources, P.M.B 1221, Effurun, Delta State, Nigeria*

ARTICLE INFO

Received: 21/05/2025
Accepted: 08/09/2025

Keywords

Halpern update, Variational Inequality Problem, strong convergence, projection method

ABSTRACT

This study proposes a new inertial residual projection method (IRPM) with or without Halpern update for solving monotone variational inequality problems (VIPs) in real Hilbert spaces. Existing explicit projection methods, including those introduced by Noor *et al.*, 2000a, 2000b, are limited by weak convergence guarantees, multiple projection steps per iteration, and fixed step-size dependence—factors that hinder their efficiency, robustness, and scalability. To address these limitations, the proposed IRPM-H method integrates an inertial extrapolation step for acceleration, Halpern-type anchoring for strong convergence, and a residual-based adaptive step-size strategy that eliminates the need for prior knowledge of Lipschitz constants. The algorithm is designed to solve VIPs involving monotone operators such as linear mappings with positive semidefinite matrices and gradients of convex functions. Under standard monotonicity and continuity assumptions, we prove that the sequence generated by the IRPM-H method converges strongly to a solution of the variational inequality, which also satisfies the fixed-point formulation. Numerical illustrations were given to justify the theretical assertons and to demonstrate the effectiveness of the proposed models. The results shows that our model compete favourably with other existing models cited in the literature.

1. INTRODUCTION

Variational inequality problems (VIPs) represent a unifying framework for a wide class of mathematical models arising in diverse fields such as optimization, equilibrium theory, network flows, economics, engineering, and machine learning (Stampacchia, 1964, kinderlehrer and Stampacchia, 1980), (Al-Mezel *et al.* 2014). Originating in the work of Fichera and formalized extensively by Kinderlehrer and Stampacchia (kinderlehrer and Stampacchia, 1980), the variational inequality problem provides an elegant formulation for systems where equilibrium is subject to constraints and governed by nonlinear or monotone dynamics.
Formally, given a real Hilbert space $\mathcal{H}$, a nonempty closed convex set $C \subseteq \mathcal{H}$, and a

*Corresponding author, e-mail: ekuma.okereke@fupre.edu.ng, garen.omonigho26@gmail.com
DIO

mapping $F: C \to \mathcal{H}$, the variational inequality problem seeks to find a point $x^* \in C$ such that:

$$\langle F(x^*), x - x^* \rangle \geq 0, \quad \forall x \in C. \qquad (1)$$

When F is monotone and Lipschitz continuous, the problem admits both rich theoretical structure and efficient solution strategies. Recent developments highlight its operator-theoretic structure and wide applicability across optimization and equilibrium problems in Hilbert spaces (Izuchukwu and Shehu, 2022).

Monotone mappings most especially those arising as gradients of convex functions, linear transformations with symmetric positive semidefinite matrices, and projection operators are central to the analysis of variational inequalities in Hilbert space (Ram and Iqbal, 2022). In practical and applied applications such as traffic network equilibrium, Nash equilibrium in games, and market clearing models, monotonicity often encodes rational behavior or conservation laws and princples, making Variational inequality problems both relevant and indispensable for modeling equilibrium phenomena (Arezadeh and Nedi¢, 2022).

A major advantage of the Variational inequality framework is its reformulation as a fixed-point problem, which enables the application of projection based iterative methods for numerical solution (see Alakoya, 2024, Alakoya and Mewomo, 2022, Blum and Oettli, 1994, Bokodisa, 2021). However, despite the elegance and theoretical soundness of projection methods such as the extragradient algorithm, forward-backward splitting, projection and contraction methods, several challenges persist. The major one among these is the limitation of weak convergence particularly in infinite-dimensional or ill-conditioned settings, sensitivity to step-size parameters, and computational costs due to multiple projection steps per iteration (Korpelevich, 1976, Bauschke and Combettes, 2017).

To address these limitations, researchers have introduced several enhancements (see Ceng *et al.*, 2021, Cholamjiak *et al.*, 2019, Cholamjiak *et al.*, 2018, Jolaoso *et al.*, 2020, Zegeye *et al.*, 2022). Inertial techniques inspired by Polyak's heavy ball method introduce a memory term that accelerates convergence (Polyak1964), while Halpern-type schemes provide and guarantees strong convergence by anchoring iterates toward a fixed reference point (Kraikaew and Saejung, 2013, 2015). Meanwhile, adaptive step size strategies that relay on adjusting step sizes based on local residuals have proven effective in improving robustness and removing reliance on unknown Lipschitz constants (Bux *et al.*, 2022).

Building upon these development, in this study, we propose a new explicit projection scheme, the Inertial Residual Projection Method with Halpern modification (IRPM-H) which integrates three key innovations: a Halpern anchoring term for strong convergence, an inertial step for acceleration, and a residual based adaptive step size rule that allows for practical implementation without prior knowledge of operator constants e.g. Lipschitz constant.

The design and analysis of IRPM-H address gaps in existing methods such as those identified in (Noor *et al.*, 2020a, 2020b) explicit projection methods, which, though elegant, often suffer from slow convergence, require fixed step sizes, and involve multiple projection steps that limit scalability in high dimensional applications.

The aim of this paper is to develop a dual mode explicit projection based method for solving monotone variational inequality problems in Hilbert spaces that achieves the same (or better) solution accuracy, strong convergence and improved computational performance.

2. METHODOLOGY

2.1: Preliminaries

Definition 2.1 (Inequality Problem) *Let H be a real Hilbert space with inner product $\langle \cdot,\cdot \rangle$ and induced norm $\|\cdot\|$. Let $C \subseteq H$ be a nonempty, closed, and convex subset and Let $T: C \to H$ be a given operator.*

The **Variational Inequality Problem (VIP)** is to find a point $x^* \in C$ such that:

$$\langle T(x^*), x - x^* \rangle \geq 0 \quad \text{for all} x \in C. \tag{2}$$

This definition is fundamental in monotone operator theory and was first formally framed in the context of Hilbert spaces by **cite author here**. The set of all such solutions is denoted as:

$$\Omega := \{x^* \in C | \langle T(x^*), x - x^* \rangle \geq 0, x \in C\}. \tag{3}$$

The goal in this study is to approximate the point $P_\Omega(u)$, the **metric projection** of a chosen anchor point $u \in H$ onto the solution set Ω, leveraging the Halpern fixed-point framework.

Definition 2.2 (Mapping) *Let $T: C \to H$ be an operator.*

• T is **monotone** if:

$$\langle T(x) - T(y), x - y \rangle \geq 0 \quad \forall x, y \in C. \tag{4}$$

• T is **strongly monotone** if there exists. $\mu > 0$ such that:

$$\langle T(x) - T(y), x - y \rangle \geq \mu \| x - y \|^2 \quad \forall x, y \in C. \tag{5}$$

• T is **pseudo-monotone** if:

$$\langle T(x), y - x \rangle \geq 0 \Rightarrow \langle T(y), y - x \rangle \geq 0 \quad \forall x, y \in C. \tag{6}$$

Definition 2.3 (Continuity) *A mapping $T: C \to H$ is Lipschitz continuous with constant $L > 0$ if:*

$$\| T(x) - T(y) \| \leq L \| x - y \| \quad \forall x, y \in C. \tag{7}$$

Definition 2.4 (onto a Convex Set) *The projection of $x \in H$ onto the convex set C is defined as:*

$$P_C(x) := \operatorname*{argmin}_{y \in C} \| x - y \|. \tag{8}$$

The projection operator P_C satisfies:

• **Nonexpansiveness**:

$$\| P_C(x) - P_C(y) \| \leq \| x - y \| \quad \forall x, y \in H. \tag{9}$$

• **Firm Nonexpansiveness**:

$$\| P_C(x) - P_C(y) \|^2 + \| (I - P_C)(x) - (I - P_C)(y) \|^2 \leq \| x - y \|^2. \tag{10}$$

$$x^* = P_C\big(x^* - \rho T(x^*)\big) \quad \text{for} \quad \rho > 0. \tag{11}$$

Where P_C is the projection onto C, and $\rho > 0$ is a step size.

Definition 2.5 (Monotonicity) *A sequence $\{x_n\}$ is Fejér monotone with respect to the solution set S if:*

$$\| x_{n+1} - x^* \| \leq \| x_n - x^* \|, \forall x^* \forall n \geq 0. \tag{12}$$

Definition 2.6 *A mapping T is said to be demiclosed at 0 if, whenever $x_n \rightharpoonup x$ and $T(x_n) \to 0$, it follows that $T(x) = 0$. This tool is especially useful when combined with nonexpansive operators and projection steps*

Definition 2.7 (convergence and Weak Convergence) *Strong convergence is convergence of sequence in norm while weak convergence is convergence in inner product.*

2.2: IRPM Algorithm and Parameters

We begin by formally defining the IRPM algorithm, with a complete description of the algorithm and its update rule, followed by interpretation of the iteration structure and parameter roles.

Inputs:

• Monotone operator $T: H \to H$
• A closed convex feasible set $C \subseteq H$. P_C the metric projector.
• Initial points $x_0, x_1 \in C$
• Anchor Point $u \in H$ (e.g., $u = x^0$)

Parameters:

• Halpern Sequence $\{\beta_k\}$
• Inertial Weight $\{\alpha_k\} \in [0, \alpha_{max})$ α_{max} limits momentum acceleration
• Adaptive numerator or Residual Scale $\delta > 0$ Controls step size sensitivity
• Step Size bound $\rho_{\max} > 0$
• Safeguard to prevents division by zero $\varepsilon > 0$
• Tolerance and Termination threshold tol, max iteration K_{max}

Iteration Steps for $\boldsymbol{k = 1, 2, 3, \ldots, K_{max}}$
Algorithm IRPM (without Halpern)

1. Inertial Step:
$$y_k = x_k + \alpha_k(x_k - x_{k-1}),$$
$$\alpha_k = \min\left(\alpha_{\text{cap}}, \frac{k-1}{k+2}\right)$$
2. Residual Calculation:
$$r_k = \| T(y_k) - T(x_k) \|$$
3. Adaptive Step Size:
$$\rho_k = \min\left(\frac{\delta}{r_k + \varepsilon},\ \rho_{\max}\right)$$
4. Projection Step:
$$x_{k+1} = P_C(y_k - \rho_k T(y_k))$$
5. Stopping Rule: Terminate the algorithm and return x_{k+1} as the solution.
Iterationresidual: $\| x_{k+1} - x_k \| < \text{tol}$
OR
Projectionresidual: $\| x_{k+1} - P_C(x_{k+1} - \rho_k T(x_{k+1})) \| < \text{tol}$
6. Update Iterates and go for the next iteration:

Output: Approximate solution $x^* \approx x_{k+1} \in C$ upon meeting the convergence criterion.

Algorithm IRPM (With Halpern)

1. Inertial Step:
$$y_k = x_k + \alpha_k(x_k - x_{k-1}),$$
$$\alpha_k = \min\left(\alpha_{\text{cap}}, \frac{k-1}{k+2}\right)$$
2. Residual Calculation:
$$r_k = \| T(y_k) - T(x_k) \|$$
3. Adaptive Step Size:
$$\rho_k = \min\left(\frac{\delta}{r_k + \varepsilon},\ \rho_{\max}\right)$$
4. Projection Step:
$$z_k = P_C(y_k - \rho_k T(y_k))$$
5. Halpern Update:
$$\beta_k = \frac{1}{100k + 100}$$
$$x_{k+1} = \beta_k u + (1 - \beta_k) z_k$$
6. Stopping Rule: Terminate the algorithm and return x_{k+1} as the solution.
Iterationresidual: $\| x_{k+1} - x_k \| < \text{tol}$
OR
Projectionresidual: $\| x_{k+1} - P_C(x_{k+1} - \rho_k T(x_{k+1})) \| < \text{tol}$
7. Update Iterates and go for the next iteration

Output: Approximate solution $x^* \approx x^{k+1} \in C$ upon meeting the convergence criterion.

2.3: Algorithmic Interpretation

We now interpret each component of the algorithm in terms of existing literature and its contribution to convergence behavior and performance.

1. Inertial Step: This extrapolates the current search direction, accelerating convergence with α_k parameter controlling

the momentum. It Escapes flat regions and speeds up convergence without sacrificing precision. For stability, the sequence $\{\alpha_k\} \in [0, \alpha_{max})$, where $\alpha_{\max} < 1$.

2. Residual Calculation: This measures the change in the operator's value, which is used to adapt the step size.

3. Adaptive Step Size: With small $\delta > 0$, $\varepsilon > 0$, the step size ρ_k is inversely proportional to the residual r_k, meaning the algorithm takes smaller steps when the operator is changing rapidly and larger steps up to rho_{max} when it is sTable. It uses only evaluations of T at the current iterate and the inertial point; no global. When the operator changes rapidly between x_k and y_k

4. Projection Step: Ensures the step stays within the feasible set C by correcting the iterate direction using a projected residual from the inertial point y_k.

5. Halpern Update: This gently pulls the sequence towards the anchor point u to prevent it from oscillating or diverging, especially in complex infinite-dimensional spaces. This guarantees it will eventually hit the true solution which ensures the entire sequence converges strongly to a solution.

6. Stopping rule terminate the algorithm and return x_{k+1} as the solution.

7. The iteration residual checks for stability of the sequence. The projection residual is a direct measure of how well the current point satisfies the fixed point condition $x^* = P_C(x^* - \rho T(x^*))$, which is equivalent to the VIP

To ensure faster convergence, we choose $\beta_k = \frac{1}{100k+10}$ or more generally $\beta_k \to 0$ with $\sum \beta_k = \infty$.

2.4 Boundedness of Iterates

Before proving convergence, we must first demonstrate that all iterates generated by the IRPM algorithm with or without Halpern remain uniformly bounded. Boundedness ensures the feasibility of the algorithm and is a key prerequisite for invoking deeper convergence tools such as the demiclosedness principle and Fejér monotonicity.

We proceed by proving several lemmas and theorems under the following standard assumptions.

Assumption 1 *Let $T: C \to H$ be monotone and L-Lipschitz continuous on a nonempty, closed, and convex set $C \subset H$. Assume:*

- $\alpha_k \in [0, \alpha_{\max}]$ with $0 \leq \alpha_{\max} < 1$
- $\beta_k \in (0,1)$ with $\beta_k \to 0$, $\sum_{k=1}^{\infty} \beta_k = \infty$
- $\rho_k \in [\rho_{\min}, \rho_{\max}]$ for some $0 < \rho_{\min} < \rho_{\max} < \frac{2}{L}$

We denote the solution set of the VIP by:

$$\Omega := \{x^* \in C | \langle T(x^*), x - x^* \rangle \geq 0, \forall x \in C\}. \quad (13)$$

Lemma 2.1 (Fejér Monotonicity of $\{x_k\}$) *Let $x^* \in \Omega$. Then the sequence $\{x_k\}$ generated by IRPM-H satisfies:*

$$\| x_{k+1} - x^* \|^2 \leq \| x_k - x^* \|^2 - (1 - \alpha_k)^2 \| x_k - x_{k-1} \|^2 + \text{errorterms}. \quad (14)$$

Proof. Recall the IRPM with Halpern update rule:

$$y_k = x_k + \alpha_k(x_k - x_{k-1}),$$
$$z_k = P_C(x_k - \rho_k T(y_k)),$$
$$x_{k+1} = \beta_k u + (1 - \beta_k) z_k.$$

Let $x^* \in \Omega$, and define:

$$\delta_k = \| x_k - x^* \|^2,$$
$$\epsilon_k = \| x_k - x_{k-1} \|^2.$$

We analyze $\| x_{k+1} - x^* \|^2$. From the update of x_{k+1}:

$$x_{k+1} = \beta_k u + (1 - \beta_k) z_k,$$

by the convexity of the squared norm:

$$\| x_{k+1} - x^* \|^2 = \| \beta_k u + (1 - \beta_k) z_k - x^* \|^2.$$

Using the convexity identity:

$\| ax + (1-a)y \|^2 = a \| x \|^2 + (1-a) \| y \|^2 - a(1-a) \| x - y \|^2,$

we get:

$$\| x_{k+1} - x^* \|^2 = \beta_k \| u - x^* \|^2 + (1-\beta_k) \| z_k - x^* \|^2 - \beta_k(1-\beta_k) \| z_k - u \|^2. \quad (15)$$

Now bound $\| z_k - x^* \|^2$. Recall:

$$z_k = P_C(x_k - \rho_k T(y_k)).$$

Invoke the firm nonexpansiveness of projection P_C:

$$\| P_C(a) - P_C(b) \|^2 \leq \langle P_C(a) - P_C(b), a - b \rangle,$$

which leads to:

$$\| z_k - x^* \|^2 \leq \| x_k - \rho_k T(y_k) - x^* \|^2 - \| x_k - \rho_k T(y_k) - z_k \|^2. \quad (16)$$

Expand:

$$\| x_k - \rho_k T(y_k) - x^* \|^2 = \| x_k - x^* \|^2 - 2\rho_k \langle T(y_k), x_k - x^* \rangle + \rho_k^2 \| T(y_k) \|^2. \quad (17)$$

Because $x^* \in \Omega$ and T is monotone: $\langle T(y_k), y_k - x^* \rangle \geq 0$.

Since $y_k = x_k + \alpha_k(x_k - x_{k-1})$:

$$\langle T(y_k), x_k - x^* \rangle = \langle T(y_k), y_k - x^* \rangle - \alpha_k \langle T(y_k), x_k - x_{k-1} \rangle.$$

Hence:

$$\langle T(y_k), x_k - x^* \rangle \geq -\alpha_k \langle T(y_k), x_k - x_{k-1} \rangle. \quad (18)$$

Substitute (17) and (18) into (16):

$$\| z_k - x^* \|^2 \leq \| x_k - x^* \|^2 + 2\rho_k \alpha_k \langle T(y_k), x_k - x_{k-1} \rangle + \rho_k^2 \| T(y_k) \|^2 - \| x_k - \rho_k T(y_k) - z_k \|^2. \quad (19)$$

Now plug (19) into (15):

$$\| x_{k+1} - x^* \|^2 \leq \beta_k \| u - x^* \|^2 + (1-\beta_k)[\| x_k - x^* \|^2 + 2\rho_k \alpha_k \langle T(y_k), x_k - x_{k-1} \rangle + \rho_k^2 \| T(y_k) \|^2 - \| x_k - \rho_k T(y_k) - z_k \|^2] - \beta_k(1-\beta_k) \| z_k - u \|^2.$$

Grouping terms and defining:

$$E_k := 2\rho_k \alpha_k \langle T(y_k), x_k - x_{k-1} \rangle + \rho_k^2 \| T(y_k) \|^2,$$

$$e_k := \| x_k - \rho_k T(y_k) - z_k \|^2,$$

and noting that $\beta_k \to 0$ and the rest decay under suiTable assumptions, we arrive at:

$$\| x_{k+1} - x^* \|^2 \leq \| x_k - x^* \|^2 - (1-\alpha_k)^2 \| x_k - x_{k-1} \|^2 + E_k - e_k + \text{boundbias}. \quad (20)$$

This completes the proof.

Theorem 2.1 (Boundedness of Iterates) *Suppose Assumption A holds. Then the sequences $\{x_k\}$, $\{y_k\}$, $\{z_k\}$, and $\{T(y_k)\}$ generated by the IRPM-H algorithm are bounded in H.*

Proof. Let $x^* \in \Omega$, the solution set of the variational inequality.

Step 1: Boundedness of $\{x_k\}$ From Lemma 4.1 (Fejér Monotonicity), we have:

$$\| x_{k+1} - x^* \|^2 \leq \| x_k - x^* \|^2 + \text{smallerrorterms} - (1-\alpha_k)^2 \| x_k - x_{k-1} \|^2. \quad (21)$$

This implies that the sequence $\{\| x_k - x^* \|\}$ is non-increasing up to a summable perturbation. Hence, $\{x_k\}$ is bounded in H.

Step 2: Boundedness of $\{y_k\}$ Recall the update rule:

$$y_k = x_k + \alpha_k(x_k - x_{k-1}). \quad (22)$$

Since $\{x_k\}$ is bounded and α_k is bounded ($\alpha_k \leq \alpha_{\max} < 1$), the difference $\{x_k - x_{k-1}\}$ is also bounded. Therefore, $\{y_k\}$ is bounded in H.

Step 3: Boundedness of $\{T(y_k)\}$ Since T is assumed to be Lipschitz continuous:

$$\| T(y_k) \| \leq \| T(y_k) - T(x_0) \| + \| T(x_0) \| \leq L \| y_k - x_0 \| + \| T(x_0) \|, \quad (23)$$

which implies that $\{T(y_k)\}$ is bounded.

Step 4: Boundedness of $\{z_k\}$ Recall:

$$z_k = P_C(x_k - \rho_k T(y_k)). \quad (24)$$

Since $\{x_k\}$ and $\{T(y_k)\}$ are bounded, and ρ_k is bounded, it follows that $\{x_k - \rho_k T(y_k)\}$ is bounded. The projection operator P_C is nonexpansive, so $\{z_k\}$ is bounded.

Thus, all sequences are bounded in the Hilbert space H.

Corollary 2.1 (Boundedness of $\{x_k\}$) *Since $x_{k+1} = \beta_k u + (1-\beta_k) z_k$, and both u and $\{z_k\}$ are bounded, it follows that:*

$$\| x_{k+1} \| \leq \beta_k \| u \| + (1-\beta_k) \sup_k \| z_k \| \leq C < \infty. \quad (25)$$

Hence $\{x_{k+1}\}$ is uniformly bounded.

Proof. From the IRPM update rule:

$$x_{k+1} = \beta_k u + (1-\beta_k) z_k, \quad \text{with } \beta_k \in (0,1).$$

This is a convex combination of two vectors u and z_k. From Theorem 2.1, $\{z_k\}$ is bounded in H, i.e., there exists $B > 0$ such that $\| z_k \| \leq B$ for all k. Thus:

$$\| x_{k+1} \| = \| \beta_k u + (1-\beta_k) z_k \| \leq \beta_k \| u \| + (1-\beta_k) \| z_k \| \leq \max\{ \| u \|, B\} < \infty.$$

Theorem 2.2 (Strong Convergence of IRPM with Halpern): *Let $\boldsymbol{H}$ be a real Hilbert space, $\boldsymbol{C} \subset \boldsymbol{H}$ a nonempty closed convex set, and let $\boldsymbol{T}: \boldsymbol{C} \to \boldsymbol{H}$ be a monotone and Lipschitz continuous operator. Let $\boldsymbol{\Omega}$ denote the solution set of the variational inequality. Let the sequences $\{\boldsymbol{\alpha_k}\}, \{\boldsymbol{\beta_k}\}, \{\boldsymbol{\rho_k}\}$ satisfy:*

- $\alpha_k \in [0, \alpha_{\max})$, $\alpha_{\max} < 1$
- $\beta_k \to 0$, $\sum_{k=0}^{\infty} \beta_k = \infty$, $\sum_{k=0}^{\infty} |\beta_{k+1} - \beta_k| < \infty$
- $\rho_k \in [\rho_{\min}, \rho_{\max}] \subset (0, \frac{2}{L})$

Then the sequence $\{x_k\}$ generated by IRPM-H converges strongly to the unique Halpern solution $x^* = P_\Omega(u)$, i.e.,

$$\lim_{k\to\infty} x_k = P_\Omega(u). \quad (26)$$

Proof. Denote $x^* := P_\Omega(u)$. Our goal is to show $\lim_{k\to\infty} \| x_k - x^* \| = 0$.

Step 1: Boundedness From Theorem 2.2, all sequences are bounded. Hence:

$$\exists M > 0 \text{ such that } \| x_k \|, \| y_k \|, \| z_k \|, \| T(y_k) \| \leq M \quad \forall k.$$

Step 2: Fixed-Point Reformulation The VIP is equivalent to the fixed-point problem:

$$x^* = P_C(x^* - \rho_k T(x^*)) \quad \forall \rho_k \in (0, 2/L).$$

Thus, $x^* \in \Omega$ is a fixed point of the nonexpansive mapping:

$$T_\rho(x) := P_C(x - \rho_k T(x)).$$

Step 3: Recursive Inequality From Lemma 4.1, $\{\| x_k - x^* \|^2\}$ is a quasi-Fejér monotone sequence, convergent up to a summable perturbation, implying:

$$\lim_{k\to\infty} \| x_k - x^* \| = \eta \quad \text{for some } \eta \geq 0.$$

Step 4: Weak Convergence via Demiclosedness Principle From monotonicity and Lipschitz continuity of T, and nonexpansiveness of P_C, the mapping $x \mapsto P_C(x - \rho T(x))$ is demiclosed at zero. The bounded sequence $\{x_k\}$ admits weak accumulation points, and each belongs to Ω.

Step 5: Halpern Anchoring and Strong Convergence The anchor-based iteration:

$$x_{k+1} = \beta_k u + (1-\beta_k) z_k,$$

is a Halpern-type iteration. Under the conditions:

- $\beta_k \to 0$,
- $\sum \beta_k = \infty$,
- $\sum |\beta_{k+1} - \beta_k| < \infty$,

strong convergence to the projection of anchor u onto Ω is guaranteed.

Hence, $\lim_{k\to\infty} x_k = P_\Omega(u) = x^*$.

The key distinction of the IRPM with Halpern scheme is that strong convergence is guaranteed without requiring T to be strongly monotone. The Halpern term

$\beta_k u$ compensates for the lack of strict contractivity by enforcing convergence to a minimal-norm solution in Ω.

2.5: Parameter Selection and Practical Tuning

To ensure effective implementation of the IRPM algorithm, this section provides a detailed guide on the selection of key algorithmic parameters: the inertial weight α_k, the Halpern anchor decay β_k, and the adaptive step size ρ_k.

1. **Inertial Weight $\boldsymbol{\alpha_k}$**

The inertial term introduces acceleration but must be controlled to avoid instability.

Recommended Setting:

$$\alpha_k = \frac{k-1}{k+2} \quad (27)$$

This choice ensures:

- $\alpha_k \in [0,1)$
- $\alpha_k \to 1$ slowly
- Summability of $\sum_{k=0}^{\infty} \| \alpha_k(x_k - x_{k-1}) \|^2 < \infty$

2. **Halpern Decay Sequence $\boldsymbol{\beta_k}$**

The sequence $\{\beta_k\}$ must satisfy:

- $\beta_k \to 0$
- $\sum \beta_k = \infty$
- $\sum |\beta_{k+1} - \beta_k| < \infty$

Recommended Setting:

$$\beta_k = \frac{1}{1000k+10} \quad (28)$$

This choice decays enough to maintain anchoring while satisfying convergence conditions.

3. RESULTS

To validate the correctness of the IRPM algorithm, we now consider some examples to show the implementation and efficiency of the proposed method. it is important to begin with test problems whose solutions are known explicitly. Such problems allow for direct comparison between the iterates generated by the algorithm and the true solution, thereby providing a clear benchmark for numerical accuracy.

Problem A1: Linear Variational Inequality

We consider the problem of finding $x^* \in C$ such that

$$\langle Mx^* + q, \quad x - x^* \rangle \geq 0, \quad \forall x \in C,$$

where the mapping is affine $A(x) = Mx + q$. Following (Bokodisa *et al.*, 2021) the problem is defined with:

$$M = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}, \quad q = \begin{bmatrix} -2 \\ -6 \end{bmatrix}, \quad C = \{x \in \mathbb{R}^2 : -2 \leq x_1, x_2 \leq 5\}.$$

The unconstrained solution is obtained by solving $Mx + q = 04$, yielding

$$x^* = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Since (1,2) lies within the feasible set C, this vector is also the solution of the variational inequality. In this test, we take the following parameter values for implementation of the IRPM algorithm: $\delta = 0.5$, $\varepsilon = 10^{-8}$, $\alpha_{\text{cap}} = 0.5$, and stop when ever iterate residual $\| x_{k+1} - x_k \| < 10^{-7}$ OR projection residual: $\| x_{k+1} - P_C(x_{k+1} - \rho_k T(x_{k+1})) \| < 10^{-7}$

Table 1: Convergence results for IRPM algorithm on Problem A1

No. iteration	Iteration solution	Iteration Residual	Projection Residual
1	[1.00000000, 1.96000000]	0.96000000	0.03840000
2	[1.00000000, 2.00800000]	0.04800000	0.00768000
3	[1.00000000, 2.00108800]	0.00691200	0.00104448
4	[1.00000000, 1.99990528]	0.00118272	0.00009093
5	[1.00000000, 1.99997256]	0.00006728	0.00002635
6	[1.00000000, 2.00000025]	0.00002769	0.00000024
7	[1.00000000, 2.00000056]	0.00000032	0.00000054
8	[1.00000000, 2.00000003]	0.00000053	0.00000003

Observation: This Table 1 demonstrates the convergence behavior of the IRPM with or without Halpern algorithm applied to Problem A1. The algorithm shows rapid convergence to the known solution, with both the iteration residual and projected residual decreasing monotonically across iterations.

Table 2: Performance comparison of algorithms on Problem A1

Algorithm	No. Iteration	No. Projection	CPU Time (s)	Operator Eval
IRPM	08	1(8)	0.04	25
Alg 3.1	16	1(16)	0.07	16
Alg 3.4	09	2(18)	0.05	18

Observation: This Table 2 compares the IRPM algorithm with algorithms 3.1 and 3.4 in (Noor *et al.*, 2020a) we use the step size rule $\rho = (0, \frac{2}{L})$ for both algorithm 3.1 and 3.4 for Problem A1. It shows that our suggested algorithm performs better in terms of number of iterations, CPU Time.

Problem A2: 3D Linear Variational Inequality

Let $C = \{x \in \mathbb{R}^3 : -1 \leq x_i \leq 1\}$ and define the operator $A: \mathbb{R}^3 \to \mathbb{R}^3$ as

$$A(x) = \begin{bmatrix} 4x_1 + x_2 + x_3 - 1 \\ x_1 + 3x_2 + 2 \\ x_1 + 2x_3 + 0.5 \end{bmatrix}$$

Find $x^* \in C$ such that

$$\langle A(x^*), x - x^* \rangle \geq 0, \quad \forall x \in C.$$

with projection onto C clipping component-wise:

$$[P_C(x)]_i = \min(\max(x_i, -1), 1)$$

and initial point $x_0 = (0,0,0)^\top$, the exact solution is:

$$x^* = \begin{pmatrix} \frac{23}{38} & -\frac{33}{38} & -\frac{21}{38} \end{pmatrix}$$

In this test, we take the following parameter values for the implementation of the IRPM-H (IRPM): $\delta = 0.5,\ \rho_{\max} = 0.25,\ \text{tol} = 10^{-7},\ \epsilon = 10^{-8},\ \alpha_{\text{cap}} = 0.5$

Table 3: IRPM & IPPM-H with different inertial values for Problem A2

Algorithm	Inertial value	No iteration	CPU time(s)	Remark
IRPM-	0	35	0.27	Converges
	$\frac{k-1}{k+2}$	43	0.33	√
	$\min\left(\alpha_{\text{cap}}, \frac{k-1}{k+2}\right)$	23	0.12	√
IRPM-H	0	27	0.34	Converges
	$\frac{k-1}{k+2}$	41	0.27	√
	$\min(\alpha)$	23	0.27	√

Observation: This Table 3 demonstrates the convergence behavior of the IRPM with or without Halpern algorithm applied to Problem A2. The algorithm shows rapid convergence to the known solution, with both the iteration residual and projected residual decreasing monotonically across iterations.

Table 4: Algorithm Comparison for Problem A2

Algorithm	No. Iteration	No. Proj	CPU Time (s)	Operator Eval
IRPM	23	1(23)	0.12	69
Alg 3.1	76	1(76)	0.69	76
Alg 3.4	24	2(48)	0.38	48

Observation: This Table 4 compares the IRPM algorithm with algorithms 3.1 and 3.4 in (Noor *et al.*, 2020b) we use the step size rule $\rho = (0, \frac{2}{L})$ for both algorithm 3.1 and 3.4 for Problem A2. It shows that our suggested algorithm performs better in terms of number of iterations, CPU Time.

Problem A3: Classic Structured Test with Tridiagonal Positive Semidefinite Matrix

Let $T(x) = Mx + q$ where

$$M = \begin{bmatrix} 4 & -1 & 0 & \cdots & 0 \\ -1 & 4 & -1 & \cdots & 0 \\ 0 & -1 & 4 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 4 \end{bmatrix}, \quad q = \begin{bmatrix} -1 \\ -1 \\ -1 \\ \vdots \\ -1 \end{bmatrix}$$

with the feasible set

$$C = \{x \in \mathbb{R}^n : x_i \leq 1, i = 1,2, \dots, n\}$$

we can easily see that T is monotone as M is symmetric and tridiagonal, positive definite with eigenvalues > 0. We are to find $x^* \in C$ such that

$$\langle T(x^*), x - x^* \rangle \geq 0, \quad \forall x \in C$$

For dimensions $n = (10,30,50,100)$ with initial points $x_0 = (0, \ldots, 0)^T$, the unique solution is $x^* = (1, \ldots, 1)$. This problem was tested in (Noor *et al.*, 2020b) new Example 3.

We take the following parameters for implementation: $\delta = 0.5$, $\rho_{\max} = 0.2$, $\alpha_{\text{cap}} = 0.5$, $\text{tol} = 10^{-7}$, $\rho \in (0, \frac{2}{L})$ for algorithm 3.1 and 3.4

Table 5: Algorithm comparison for Problem A3 for Cases n = 10, 50, 100

Case n = 10

Algorithm	No. Iter	No. Proj	CPU Time (s)	Operator Eval
IRPM	21	1(21)	0.03	42
Alg 3.1	38	1(38)	0.52	77
Alg 3.4	19	2(38)	0.45	58

Case n = 30

Algorithm	No. Iter	No. Proj	CPU Time (s)	Operator Eval
IRPM	21	1(21)	0.03	42
Alg 3.1	41	1(41)	0.64	83
Alg 3.4	21	2(42)	0.23	64

Case n = 50

Algorithm	No. Iter	No. Proj	CPU Time (s)	Operator Eval
IRPM	21	1(21)	0.03	42
Alg 3.1	42	1(42)	0.65	85
Alg 3.4	21	2(42)	0.29	64

Case n = 100

Algorithm	No. Iter	No. Proj	CPU Time (s)	Operator Eval
IRPM	21	1(21)	0.06	42
Alg 3.1	43	1(43)	0.73	87
Alg 3.4	22	2(44)	0.35	67

Observations: Observation: This Table 5 compares the IRPM algorithm with algorithms 3.1 and 3.4 in (Noor *et al.*, 2020b) we use the step size rule $\rho = (0, \frac{2}{L})$ for both algorithm 3.1 and 3.4 for Problem A3. It shows that our suggested algorithm performs better in terms of number of iterations, CPU Time and operator evaluation for three cases of n.

Problem B1: Linear Complementarity Problem

Let $n \in \mathbb{N}$. The the operator $T: \mathbb{R}^n \rightarrow \mathbb{R}^n$ is defined by

$$T(x) = Mx + q$$

where

$$M = \text{diag}(\frac{1}{n}, \frac{2}{n}, \dots, \frac{n}{n}) \in \mathbb{R}^{n\times n},$$

$$q = (-1, -1, \dots, -1)^T \in \mathbb{R}^n$$

with the feasible set the non-negative orthant intersected with box constraints:

$$C = \{x \in \mathbb{R}^n : 0 \leq x_i \leq 1, i = 1,2, \dots, n\}$$

We are to find $x^* \in C$ such that

$$\langle T(x^*), x - x^* \rangle \geq 0, \quad \forall x \in C$$

For dimensions $n = (10,30,50,100)$ with initial points $x_0 = (0, \dots, 0)^T$ and unique solution $x^* = (1, \dots, 1)^T$. This test problem was treated in Noor *et al.*, 2000a Example 4. We take the following parameters for implementation: $\delta = 0.5$, $\rho_{\max} = 0.3$, $\alpha_{\text{cap}} = 0.5$, $\text{tol} = 10^{-7}$, $\rho \in (0, \frac{2}{L})$.

Table 6: Algorithm comparison for Problem B1 for cases n = 10, 30, 50,100

Case n = 10

Algorithm	**No. Iter**	**No. Proj**	**CPU Time (s)**	**Operator Eval**
IRPM	7	1(7)	0.05	14
Alg 3.1	14	1(14)	0.01	29
Alg 3.4	7	2(14)	0.09	23

Case n = 30

Algorithm	**No. Iter**	**No. Proj**	**CPU Time (s)**	**Operator Eval**
IRPM	7	1(7)	0.05	14
Alg 3.1	14	1(14)	0.02	29
Alg 3.4	7	2(14)	0.05	23

Case n = 50

Algorithm	**No. Iter**	**No. Proj**	**CPU Time (s)**	**Operator Eval**
IRPM	7	1(7)	0.05	14
Alg 3.1	14	1(14)	0.02	29
Alg 3.4	7	2(14)	0.05	23

Case n = 100

Algorithm	**No. Iter**	**No. Proj**	**CPU Time (s)**	**Operator Eval**
IRPM	7	1(7)	0.05	14
Alg 3.1	14	1(14)	0.01	29
Alg 3.4	7	2(14)	0.05	23

Observations: This Table 6 compares the IRPM algorithm with algorithms 3.1 and 3.4 in (Noor *et al.*, 2020b) we use the step size rule $\rho = (0, \frac{2}{L})$ for both algorithm 3.1 and 3.4 for Problem B1. It shows that our suggested algorithm performs better in terms of number of iterations, CPU Time and operator evaluation for three cases of n.

Problem B2: The HP-Hard Variational Inequality Problem

This problem, inspired by the construction of Harker, is specifically designed to present a significant challenge for algorithms designed to solve Variational Inequality Problems (VIP). The goal is to find a vector $\mathbf{x}^* \in C$ such that:

$$\langle T(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^* \rangle \geq 0 \quad \forall \mathbf{x} \in C,$$

where the function $F: \mathbb{R}^n \to \mathbb{R}^n$ is given by:

$$T(\mathbf{x}) = M\mathbf{x} + \mathbf{q},$$

with $M \in \mathbb{R}^{n\times n}$ a positive definite matrix and $\mathbf{q} \in \mathbb{R}^n$ a vector. The matrix M is constructed to be positive definite through the summation of three randomly generated matrices:

$$M = A^\top A + B + D.$$

Matrix $\boldsymbol{A}$: is a dense matrix where each entry a_{ij} is independently and uniformly generated from the interval $(-5,5)$. The product $A^\top A$ ensures the resulting matrix is positive semi-definite. **Matrix $\boldsymbol{B}$:** A skew-symmetric matrix ($B = -B^\top$) with each entry b_{ij} for $i < j$ is independently and uniformly generated from $(-5,5)$, with $b_{ji} = -b_{ij}$ and $b_{ii} = 0$. This component introduces antisymmetry into the system without affecting the positive definiteness of M, as $\mathbf{x}^\top B\mathbf{x} = 0$ for all $\mathbf{x}$.
Matrix $\boldsymbol{D}$: A diagonal matrix where each diagonal entry d_{ii} is independently and uniformly generated from the interval $(0,0.3)$. This matrix ensures M is positive definite and full rank. The vector $\mathbf{q}$ is generated such that each component q_i is independently and uniformly distributed from the interval $(-500,0)$. This significant negative offset is a primary source of the problem's difficulty, as it forces the solution towards the boundary of the simplex, testing the algorithm's ability to handle active constraints.

The feasible set C is defined as the **standard simplex** in $\mathbb{R}^n$:

$$C = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{x} \geq \mathbf{0},\ \mathbf{e}^\top\mathbf{x} = n\},$$

where $\mathbf{e}$ denotes the vector of ones in $\mathbb{R}^n$. This constraint requires the solution to be a non-negative vector whose components sum to n with initial point set to: $\mathbf{x}^0 = (1,1,\ldots,1)$. A similar type of problem was tested in (Noor *et al*., 2000a, 2000b).

We take the following parameters: $\delta = 0.5$, $\rho_{\max} = 0.005$, $\alpha_{\text{cap}} = 0.5$, $\text{tol} = 10^{-7}$, $\rho \in (0, \frac{2}{L})$.

Table 7: Algorithm comparison for Problem B2 for Cases n = 10, 30, 50, 100

Case n = 10

Algorithm	No. Iter	No. Proj	CPU Time (s)	Operator Eval
IRPMM	42	1(42)	0.041	84
Alg 3.1	33	1(33)	0.039	33
Alg 3.4	46	2(92)	0.043	92

Case n = 30

Algorithm	No. Iter	No. Proj	CPU Time (s)	Operator Eval
IRPM	171	1(171)	0.038	342
Alg 3.1	749	1(749)	0.104	749
Alg 3.4	180	2(360)	0.061	360

Case n = 50

Algorithm	No. Iter	No. Proj	CPU Time (s)	Operator Eval
IRPM	360	1(360)	0.104	720
Alg 3.1	1091	1(1091)	0.140	1091

Alg 3.4	365	2(730)	0.077	730
Case n = 100				
Algorithm	**No. Iter**	**No. Proj**	**CPU Time (s)**	**Operator Eval**
IRPM	311	1(311)	0.1073	622
Alg 3.1	799	1(799)	0.107	799
Alg 3.4	337	2(674)	0.083	674

Observations: This Table 7 compares the IRPM algorithm with algorithms 3.1 and 3.4 in (Noor *et al.*, 2020b) we use the step size rule $\rho = (0, \frac{2}{L})$ for both algorithm 3.1 and 3.4 for Problem B2. It shows that our suggested algorithm performs better in terms of number of iterations, CPU Time and operator evaluation for three cases of n.

The Braess Network Problem

This problem illustrates the **Braess Paradox**, a phenomenon where adding capacity to a network (e.g., a new road) can lead to increased overall travel time and congestion for all users. This problem is formulated as a Variational Inequality (VI) considered in Marcotte and possesses a known, unique solution, making it an ideal test case for validating algorithmic correctness and observing fundamental performance characteristics.

The network is defined by a directed graph $G(N, A)$ with:

- **Node Set:** $N = \{1,2,3,4\}$
- **Arc Set:** $A = \{a_1, a_2, a_3, a_4, a_5\} = \{(1,2), (1,3), (2,3), (2,4), (3,4)\}$

A total travel demand of 6 units flows from origin node 1 to destination node 4. The vector of arc flows is denoted as $\mathbf{x} = (x_{12}, x_{13}, x_{23}, x_{24}, x_{34})^T \in \mathbb{R}_+^5$. The cost (e.g., travel time) on each arc is a linear, separable function of its own flow. The cost vector $\mathbf{T}(\mathbf{x})$ is given by:

$$\mathbf{T}(\mathbf{x}) = M\mathbf{x} + \mathbf{q}$$

where:

- $M = \text{diag}(10,1,1,1,10)$ is a diagonal matrix of congestion sensitivity parameters.
- $\mathbf{q} = (0,50,10,50,0)^T$ is a vector of free-flow travel times.

This yields the following explicit cost functions:

$$\begin{aligned} T_{12}(x_{12}) &= 10x_{12} \\ T_{13}(x_{13}) &= x_{13} + 50 \\ T_{23}(x_{23}) &= x_{23} + 10 \\ T_{24}(x_{24}) &= x_{24} + 50 \\ T_{34}(x_{34}) &= 10x_{34} \end{aligned}$$

The feasible set C consists of all non negative flow vectors $\mathbf{x}$ that satisfy flow conservation at every node and is given as:

$$C = \{\mathbf{x} \in \mathbb{R}_+^5 | B\mathbf{x} = \mathbf{b}\}$$

where B is the node-arc incidence matrix:

$$B = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{bmatrix} \quad (\text{rowscorrespondtonodes1to4})$$

and $\mathbf{b} = (6,0,0,-6)^T$ is the supply/demand vector, encoding 6 units of flow entering at node 1 and exiting at node 4. For computational purposes, due to the linear dependence in the rows of B (rank = 3), the first row is removed to form a full row rank matrix $\hat{B}$ and a corresponding reduced vector $\hat{\mathbf{b}} = (0,0,-6)^T$.

The traffic equilibrium is characterized by the solution $\mathbf{x}^* \in C$ such that:

$$\langle \mathbf{T}(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^* \rangle \geq 0 \quad \forall \mathbf{x} \in C$$

where $\langle \cdot,\cdot \rangle$ denotes the inner product. This condition ensures that no user can unilaterally change their route to reduce their travel cost. The unique solution to this VI is:

$$\mathbf{x}^* = (4,2,2,2,4)^T$$

The paradox is observed if arc (2,3) is removed; the resulting equilibrium yields a lower total system travel time, demonstrating that the presence of the "shortcut" is collectively detrimental. To test the algorithms from a non equilibrium state, the following initial flow vector is used:

$$\mathbf{x^0} = (6,0,6,0,6)^T$$

This initial point represents a state where all flow is forced along the path $1 \to 2 \to 3 \to 4$, providing a significant deviation from the true equilibrium for algorithms to overcome. We take the following parameters: $\delta = 0.5$, $\rho_{\max} = 0.3$, $\alpha_{\text{cap}} = 0.5$, $\text{tol} = 10^{-7}$, $\rho \in (0, \frac{2}{L})$.

Table 8: Convergence of Algorithms to solutions to Braess Network Problem

No. Iter	**Iter solution**	**Iter Residual**	**Proj Residual**
1	[4.049375, 1.950625, 2.098750, 1.950625, 4.049375]	5.44648998	0.13616225
2	[3.669576, 2.330424, 1.339151, 2.330424, 3.669576]	1.07423479	0.21979868
3	[3.877126, 2.122874, 1.754252, 2.122874, 3.877126]	0.58704045	0.25900687
4	[3.999523, 2.000477, 1.999045, 2.000477, 3.999523]	0.34619024	0.00131675
5	[4.001518, 1.998482, 2.003036, 1.998482, 4.001518]	0.00564413	0.00418627
6	[4.000063, 1.999937, 2.000126, 1.999937, 4.000063]	0.00411572	0.00017344
7	[3.999983, 2.000017, 1.999967, 2.000017, 3.999983]	0.00022489	0.00004582
8	[3.999999, 2.000001, 1.999997, 2.000001, 3.999999]	0.00004301	0.00000389
9	[4.000000, 2.000000, 2.000000, 2.000000, 4.000000]	0.00000442	0.00000043
10	[4.000000, 2.000000, 2.000000, 2.000000, 4.000000]	0.00000054	0.00000005

Observation: This Table 8 demonstrates the convergence behavior of the IRPM with or without Halpern algorithm applied to Brass Network Problem . The algorithm shows rapid convergence to the known solution, with both the iteration residual and projected residual decreasing monotonically across iterations.

4. CONCLUSION

We introduced two algorithms called IRPM with or without Halpern update applied to solving different variational inequality problems. The IRPM without Halpern is known to converge weakly to the known solution while we proved strong convergence for IRPM with Halpern update. Using a stopping criterion or tolerance value, the two contructed algorithms shows rapid convergence to the known solution, with both the iteration residuals and projected residuals decreasing monotonically across iterations as displayed through Tables 1-8. The algorithms so constructed and studied improved computational performance and is good fit for solving variational inequality problems in Hilbert without computing the projection rule twice which is comptutationally constly as in the case of extragradient method.

Conflicts of interest

The authors declared that there is no conflict of interest.

Acknowledgments

The author(s) acknowledge(s) the Editor in chief and the Reviewers for their selfless services to improve this research work.

REFERENCES

Alakoya, T. O. (Ed.). (2024). “Advances in fixed point theory and its applications”. https://doi.org/10.3390/books978-3-7258-1876-1

Alakoya, T., and Mewomo, O. (2022). “Modified inertial hybrid subgradient extragradient method for solving variational inequalities and fixed point problems for an infinite family of multivalued relatively nonexpansive mappings in Banach spaces with applications”, *Matematicki Vesnik*.

Al-Mezel, S. A. R., Al-Solamy, F. R. M., and Ansari, Q. H. (2014). “Fixed point theory, variational analysis, and optimization”, *CRC press*.

Arezadeh, S., & Nedi¢, A. (2024). “Non monotone variational inequalities”, *Proceedings of the 2024 60th Annual Allerton Conference on Communication, Control and Computing*, 1-7. https://doi.org/10.1109/Allerton63246.2024. 10735327

Bauschke, H. H., and Combettes, P. L. (2017). “Convex analysis and monotone operator theory in Hilbert spaces”, *Springer*. https://doi.org/10.1007/978-3-319-48311-5

Blum, E., and Oettli, W. (1994). “From optimization and variational inequalities to equilibrium problems”, *The Mathematics student*, 63, 123-145.

Bokodisa, A. T., Jolaoso, L. O., and Aphane, M. (2021). “Halpern-subgradient extragradient method for solving equilibrium and common fixed point problems in reflexive banach spaces”, *Mathematics*, 9(7), 743. https://doi.org/10. 3390/math9070743

Bux, M., Ullah, S., Arif, M. S., and Abodayeh, K. (2022). “A self-adaptive technique for solving variational inequalities: A new approach to the problem”, *Journal of Function Spaces*, 2022, 7078707. https://doi.org/10.1155/2022/7078707

Ceng, L.-C., Yao, J.-C., and Shehu, Y. (2021). “On Mann type subgradient-like extragradient method with linear search process for hierarchical variational inequalities for asymptotically nonexpansive mappings", *Mathematics*, 9(24), 3322. https://doi.org/10.3390/math9243322

Cholamjiak, P., Thong, D. V., and Cho, Y. J. (2019). "A novel inertial projection and contraction method for solving pseudomonotone variational inequality problems", *Acta Applicandae Mathematicae*, 169(1), 217-245. https://doi. org/10.1007/s10440-019-00297-7

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), 42. https://doi.org/10. 1007/s11784-018-0526-5

Izuchukwu, C., and Shehu, Y. (2022). "New inertial projection methods for solving multivalued variational inequality problems and fixed point problems in hilbert spaces", *Journal of Optimization Theory and Applications*, 193(1), 297-323. https://doi.org/10.1007/s10957-021-01913-z

Jolaoso, L. O., Taiwo, A., Alakoya, T. O., and Mewomo, O. T. (2020). "A strong convergence theorem for solving pseudo-monotone variational inequalities using projection methods", *Journal of Optimization Theory and Applications*, 185(3), 744 -766. https://doi.org/10.1007/s10957-020-01672-3

Kinderlehrer, D., and Stampacchia, G. (1980). "An introduction to variational inequalities and their applications", Academic Press. https://doi.org/10.1137/ 1.9780898719451

Korpelevich, G. M. (1976). "The extragradient method for finding saddle points and other problems", *Ekonomika i Matematicheskie Metody*, 12, 747-756.

Kraikaew, R., and Saejung, S. (2013). "Strong convergence of the Halpern subgradient extragradient method for solving variational inequalities in hilbert spaces", *Journal of Optimization Theory and Applications*, 163(2), 399-412. https: //doi.org/10.1007/s10957-013-0494-2

Kraikaew, R., and Saejung, S. (2015). "Strong convergence of the Halpern subgradient extragradient method in banach spaces", *Journal of Optimization Theory and Applications*, 163(2), 399-412. https://doi.org/10.1007/s10957-013-0494-2

Noor, M. A., Noor, K. I., and Bnouhachem, A. (2020a). "Some new iterative methods for solving variational inequalities", *Canadian Journal of Applied Mathematics*, 2(2), 1-7

Noor, M. A., Noor, K. I., and Rassias, M. T. (2020b). "New trends in general variational inequalities", *Acta Applicandae Mathematicae*, 170(1), 981-1064.

Polyak, B. T. (1964). "Some methods of speeding up the convergence of iteration methods", *USSR Computational Mathematics and Mathematical Physics*, 4(5), 1-17. https://doi.org/10.1016/0041-5553(64)90137-5

Ram, T., and Iqbal, M. (2022). "Generalized monotone mappings and applications", *Communications in Mathematics and Applications*, 13(2), 477-491. https: //doi.org/10.26713/cma.v13i2.1723

Stampacchia, G. (1964). “Formes bilinéaires coercitives sur les ensembles convexes”, Comptes Rendus de l'Académie des Sciences, 258, 4413-4416.

Zegeye, S. B., Zegeye, H., Sangago, M. G., and Boikanyo, O. A. (2022). “A convergence theorem for a common solution of fixed point, variational inequality and generalized mixed equilibrium problems in banach spaces”, *International Journal of Nonlinear Analysis and Applications*, 13(2), 1069-1087, https://doi.org/10.22075/ijnaa.2022.25363.299