

ORIGINAL RESEARCH ARTICLE

Two-step Inertial Self-Adaptive Gradient Methods for the Split Feasibility Problem with Application

¹*L.B. Mohammed, ²B.I. Babura, and ³A. Ibrahim

¹Department of Mathematics, Faculty of Physical Sciences, Federal University Dutse, PMB 7156, Dutse, Jigawa State, Nigeria

ABSTRACT

This paper introduces novel two-step inertial self-adaptive gradient algorithms (TISGA) for solving the Split Feasibility Problem (SFP) in Hilbert spaces. The proposed method integrates dual inertial parameters $\gamma_1 \in (0, \frac{1}{2}]$ and $\gamma_2 \leq 0$ satisfying $\gamma_1 + \gamma_2 \geq 0$, together with a self-adaptive step-size $\lambda_n = \frac{\beta \rho h(w_n)}{\|\nabla h(w_n)\|^2}$ where $\rho \in (0, 4)$ and $\beta > 0$ is chosen to satisfy specific bounding conditions. This approach eliminates the requirement for prior knowledge of the bounded linear operator's norm—a value often challenging to determine in practice. Under mild assumptions on the underlying operators, we establish weak convergence of the algorithm to a solution of the SFP. Numerical examples and an application are given to justify the theoretical results presented. The self-adaptive mechanism ensures practical implementability without norm estimation, while the two-step inertial acceleration potentially enhances convergence speed. This work extends and generalizes several established results in the existing literature, including one-step inertial methods and Euclidean-space two-step methods.

ARTICLE HISTORY

Received December 26, 2025

Accepted May 24, 2026

Published June 08, 2026

KEYWORDS

Fixed Point Problem; Iterative algorithm; Nonlinear Mappings; Weak and Strong Convergent.



© The Author(s). This is an Open Access article distributed under the terms of the Creative Commons Attribution 4.0 License [creativecommons.org](https://creativecommons.org/licenses/by-nc/4.0/)

INTRODUCTION

Consider Hilbert spaces \mathcal{H}_1 and \mathcal{H}_2 , with nonempty closed convex subsets $\mathcal{C} \subseteq \mathcal{H}_1$ and $\mathcal{Q} \subseteq \mathcal{H}_2$, and a bounded linear operator $\mathcal{B}: \mathcal{H}_1 \rightarrow \mathcal{H}_2$. The SFP is then to determine a point

$$x \in \mathcal{C} \text{ such that } \mathcal{B}x \in \mathcal{Q}. \quad (1)$$

This problem has found extensive applications in diverse fields, including signal processing, medical imaging, intensity-modulated radiation therapy, and control theory, making it a cornerstone of modern optimization research (see Byrne, 2003; Ansari & Rehan, 2014; Censor et al., 2006; and references therein).

Traditional methods for solving the SFP, such as the CQ algorithm and its variants, rely on projection-based techniques (see Kiliçman & Mohammed, 2016, 2018; Byrne, 2002; Mohammed et al., 2025; Anh et al., 2018; Kesornprom, 2020; Ma & Liu, 2022). One limitation of the CQ algorithm (see Byrne, 2003) is the requirement to compute or estimate the norm of the bounded linear operator at each iteration. This task can be challenging in many scenarios, posing a significant obstacle to the algorithm's practical implementation. To address this issue, Qu and Xiu (2005) proposed a modified version of the CQ algorithm by incorporating the Armijo line-search procedure to determine the step size. However, their approach involves computing the value of $h(x)$ multiple

times during the line-search step to find the smallest nonnegative integer. This can become computationally expensive, particularly when the set \mathcal{Q} has a complex structure, making the evaluation of $h(x)$ and the line-search procedure costly, as reported by Vinh et al. (2023).

To overcome these constraints, gradient-based approaches are commonly used for their computational effectiveness. However, a key limitation of such methods lies in their reliance on an appropriate step-size selection, which often demands prior estimation of the operator norm—a value that can be impractical to determine.

The CQ algorithm operates as a gradient-projection technique applied to the optimization problem

$$\min_{x \in \mathcal{C}} h(x),$$

where $h(x) := \frac{1}{2} \|\mathcal{B}x - P_{\mathcal{Q}}(\mathcal{B}x)\|^2$ for all $x \in \mathcal{H}_1$. This objective function h is both convex and differentiable, possessing a Lipschitz-continuous gradient of the form:

$$\nabla h(x) = \mathcal{B}^*(I - P_{\mathcal{Q}})\mathcal{B}x.$$

To circumvent the need for operator norm estimation, López et al. (2012) proposed an alternative iterative scheme. Beginning from an arbitrary $x_0 \in \mathcal{H}_1$, the algorithm generates subsequent iterates via:

Correspondence: L. B. Mohammed. Department of Mathematics, Faculty of Physical Sciences, Federal University Dutse, PMB 7156, Dutse, Jigawa State, Nigeria. ✉ lawanbulama@gmail.com.

How to cite: Mohammed, L. B., Babura, B. I., & Arzuka, I. (2026). Two-step Inertial Self-Adaptive Gradient Methods for the Split Feasibility Problem with Application. *UMYU Scientifica*, 5(2), 131 – 145. <https://doi.org/10.56919/usci.2652.013>

$$x_{n+1} = P_{\mathcal{C}}(I - \tau_n \mathcal{B}^*(I - P_Q)\mathcal{B})x_n, \quad \forall n \in \mathbb{N},$$

where each step size τ_n is adaptively computed according to Polyak's (1987) rule:

$$\tau_n := \rho \frac{h(x_n)}{\|\nabla h(x_n)\|^2}, \quad 0 < \rho < 4.$$

This approach avoids the need for repeated evaluations of $h(x)$ during the line search, thereby reducing computational overhead, especially in cases where Q has a complicated structure.

Other step sizes have been considered by numerous scholars. Notable examples include:

Dynamic Step Size by López et al. (2012)

$$\tau_n := \frac{\rho_n \|(I - P_Q)\mathcal{B}x_n\|^2}{\|\mathcal{B}^*(I - P_Q)\mathcal{B}x_n\|^2}, \quad \rho_n \in (0, 4). \quad (2)$$

Adaptive Step Size by Anh et al. (2018)

$$\tau_n := \frac{\rho_n}{\max\{1, \|\mathcal{B}^*(I - P_Q)\mathcal{B}x_n\|\}}, \quad (3)$$

where the sequence $\{\rho_n\}$ satisfies $\lim_{n \rightarrow \infty} \rho_n = 0$ and $\sum_{n \geq 0} \rho_n = \infty$.

The inertial method, first established by Attouch and Alvarez (2001) for analyzing maximal monotone operators, has gained widespread recognition for solving the SFP, inspiring the development of numerous inertial-type algorithms in the literature (see Suantai et al., 2018; Sahu et al., 2021; Shehu & Gibali, 2021; Vinh et al., 2023; Alvarez & Attouch, 2001; Mohammed et al., 2026). In recent years, these techniques have proven to be a highly effective approach for enhancing the convergence rate of iterative procedures.

This motivated Vinh et al. (2023) to propose an inertial approach combined with Polyak's step size to design an algorithm specifically tailored for solving the SFP. This method incorporates the projection of the inertial term onto the feasible set. Under standard assumptions concerning the operators and parameters involved, they demonstrated the weak convergence of the introduced iterative scheme, which is defined as follows:

$$\begin{cases} w_n = P_{\mathcal{C}}(x_n + \tau_n(x_n - x_{n-1})), \\ x_{n+1} = w_n - \lambda_n \nabla h(w_n), \quad \forall n \geq 1, \end{cases} \quad (4)$$

where the step size is given by

$$\tau_n = \rho_n \frac{h(w_n)}{\|\nabla h(w_n)\|^2}.$$

The research presented in [22] demonstrated that the conventional one-step inertial technique

$$w_n = x_n + \tau(x_n - x_{n-1}), \quad \tau \in [0, 1),$$

can, in fact, fail to achieve accelerated convergence, a point also discussed in Okeke and Adamu (2023). As noted in Liang (2016), incorporating inertial terms that consider more than just the two most recent iterates can

be more effective for acceleration. For instance, a two-step inertial method of the form

$$y_n = x_n + \alpha(x_n - x_{n-1}) + \delta(x_{n-1} - x_{n-2}),$$

with parameters $\alpha > 0$ and $\delta < 0$, has been shown to successfully enhance convergence speed. The limitations of the one-step inertial approach are further elaborated upon in Poon and Liang (2019).

In light of the constraints associated with one-step inertial acceleration, Polyak (1987) proposed that employing multi-step inertial approaches could improve the rate of convergence in optimization processes, though no convergence results were provided. Recent studies, such as those in Mohammed et al. (2026), Okeke and Adamu (2023), Poon and Liang (2019), and Adamu et al. (2025), have explored multi-step inertial algorithms. These multi-step approaches have shown significant advantages over one-step inertial algorithms, highlighting the potential of multi-step techniques to overcome the limitations of one-step methods, offering faster and more efficient solutions for optimization problems.

Recently, Adamu et al. (2025) considered the following two-step inertial CQ method for the SFP:

$$\begin{cases} w_n = x_n + \tau(x_n - x_{n-1}) + \gamma(x_{n-1} - x_{n-2}), \\ x_{n+1} = P_{\mathcal{C}}(w_n - \eta_n \nabla h(w_n)), \quad \forall n \geq 1, \end{cases} \quad (5)$$

$$\text{where } \eta_n := \begin{cases} \frac{\sigma h(w_n)}{\|\nabla h(w_n)\|^2}, & \|\nabla h(w_n)\| \neq 0, \\ 0, & \text{otherwise.} \end{cases}$$

Under specific conditions applied to the parameters and operators involved, the authors successfully demonstrated the weak convergence of their algorithm. We note, however, that this result was confined to Euclidean space. Building on these results, our study seeks to:

Develop a new two-step inertial algorithm featuring a self-adaptive step size to address the SFP. The key innovation involves the projection of the inertial term onto the constraint set \mathcal{C} , representing a strategic departure from existing methodologies that is expected to enhance convergence of an algorithm.

Establish a theoretical framework for the proposed algorithm under standard assumptions.

Demonstrate how the integrated approach of inertial acceleration and self-adaptivity eliminates the requirement for prior knowledge of the operator norm of the bounded linear mapping.

The organization of this manuscript is outlined below. Section 2 introduces the necessary preliminary concepts. Section 3 presents the proposed algorithm and its convergence analysis. Section 4 provides numerical experiments. Section 5 offers concluding remarks.

PRELIMINARIES

This section introduces foundational definitions and lemmas that will be used throughout the paper.

Let u be an element of the Hilbert space \mathcal{H}_1 . The metric projection onto \mathcal{Q} , denoted by $P_{\mathcal{Q}}u$, is the unique point in \mathcal{Q} that minimizes the distance from u to the set. This is characterized by the following condition:

$$\|u - P_{\mathcal{Q}}u\| = \min_{v \in \mathcal{Q}} \|u - v\|.$$

Remark 1. The metric projection operator $P_{\mathcal{Q}}: \mathcal{H}_1 \rightarrow \mathcal{Q}$ possesses the following key properties for all $u \in \mathcal{H}_1$ and all $v \in \mathcal{Q}$:

$$\langle u - P_{\mathcal{Q}}u, v - P_{\mathcal{Q}}u \rangle \leq 0 \text{ (variational inequality);}$$

$$\|u - P_{\mathcal{Q}}u\|^2 \leq \langle u - P_{\mathcal{Q}}u, u - v \rangle;$$

$$\|P_{\mathcal{Q}}u - v\|^2 \leq \|u - v\|^2 - \|u - P_{\mathcal{Q}}u\|^2.$$

Definition 2. Let $f: \mathcal{H}_1 \rightarrow \mathbb{R}$ be a convex function on a Hilbert space \mathcal{H}_1 . The function f is subdifferentiable at $u \in \mathcal{H}_1$ if its subdifferential at u , $\partial f(u) = \{w \in \mathcal{H}_1 \mid f(v) \geq f(u) + \langle w, v - u \rangle \text{ for all } v \in \mathcal{H}_1\}$, is nonempty. Any element $w \in \partial f(u)$ is called a subgradient, and the defining inequality is known as the subgradient inequality.

Remark 3. If a convex function f is Fréchet differentiable at u , its subdifferential consists solely of its gradient; that is, $\partial f(u) = \{\nabla f(u)\}$.

Definition 4. A function $f: \mathcal{H}_1 \rightarrow \mathbb{R}$ is lower semicontinuous at a point u if for every sequence $\{u_n\}$ in \mathcal{H}_1 converging to u , $f(u) \leq \liminf_{n \rightarrow \infty} f(u_n)$. The function is said to be lower semicontinuous on \mathcal{H}_1 if this property holds at every point in the space.

Lemma 5 (Okeke and Adamu (2023)). Let $u, v, w \in \mathcal{H}_1$ and let $\alpha, \gamma \in \mathbb{R}$. Then the following identities hold:

$$2\langle u, v \rangle = \|u\|^2 + \|v\|^2 - \|u - v\|^2 = \|u + v\|^2 - \|u\|^2 - \|v\|^2;$$

$$\begin{aligned} \|(1 + \alpha)u - (\alpha - \gamma)v - \gamma w\|^2 &= (1 + \alpha)\|u\|^2 - (\alpha - \gamma)\|v\|^2 - \gamma\|w\|^2 \\ &\quad + (1 + \alpha)(\alpha - \gamma)\|u - v\|^2 \\ &\quad + \gamma(1 + \alpha)\|u - w\|^2 - \gamma(\alpha - \gamma)\|v - w\|^2. \end{aligned}$$

Lemma 6 Aubin (2013). Let $F: \mathcal{H}_1 \rightarrow \mathbb{R}$ be a function defined by $F(u) := \frac{1}{2} \|Bu - P_{\mathcal{Q}}(Bu)\|^2, \quad \forall u \in \mathcal{H}_1$. Then the following properties hold:

F is convex and differentiable;

F is weakly lower semicontinuous (w-lsc) on \mathcal{H}_1 ;

$$\nabla F(u) = \mathcal{B}^*(I - P_{\mathcal{Q}})Bu \text{ for all } u \in \mathcal{H}_1;$$

∇F is $\frac{1}{\|\mathcal{B}\|^2}$ -inverse strongly monotone, that is,

$$\langle \nabla F(u) - \nabla F(v), u - v \rangle \geq \frac{1}{\|\mathcal{B}\|^2} \|\nabla F(u) - \nabla F(v)\|^2, \quad \forall u, v \in \mathcal{H}_1.$$

Lemma 7 (Opial (1967)). Let \mathcal{S} be a nonempty subset of a Hilbert space \mathcal{H}_1 , and let $\{u_n\}_{n \in \mathbb{N}}$ be a sequence in \mathcal{H}_1 satisfying:

For each $z \in \mathcal{S}$, $\lim_{n \rightarrow \infty} \|u_n - z\|$ exists;

$\Omega_w(u_n) \subseteq \mathcal{S}$, where $\Omega_w(u_n)$ denotes the weak ω -limit set of $\{u_n\}$.

Then $\{u_n\}$ converges weakly to some $u^* \in \mathcal{S}$.

MAIN RESULTS

This section outlines the principal contributions of this paper. Throughout, we denote by \mathcal{S} the solution set of the split feasibility problem stated in (1). That is,

$$\mathcal{S} = \{x \in \mathcal{C} \text{ such that } \mathcal{B}x \in \mathcal{Q}\}. \tag{6}$$

Let $P_C: \mathcal{H}_1 \rightarrow \mathcal{C}$ and $P_Q: \mathcal{H}_2 \rightarrow \mathcal{Q}$ denote the metric projection operators onto \mathcal{C} and \mathcal{Q} , respectively. Furthermore, let $\mathcal{B}: \mathcal{H}_1 \rightarrow \mathcal{H}_2$ be a bounded linear operator with adjoint \mathcal{B}^* .

Consider the sequence $\{x_n\}$ generated by the following iterative scheme:

Algorithm 1: Two-step Inertial Self-Adaptive Gradient Algorithm (TISGA)

Initialization: Choose arbitrary initial points $x_0, x_1, x_2 \in \mathcal{H}_1$. Let:

$$\rho \in (0, 4),$$

$$\gamma_1 \in (0, \frac{1}{2}], \gamma_2 \leq 0, \text{ such that } \gamma_1 + \gamma_2 \geq 0,$$

$$\beta > 0 \text{ such that } \beta \leq \min \left\{ \frac{(4-\rho)(1-\gamma_1+\gamma_2)}{2\gamma_1\rho}, \frac{(4-\rho)A}{\rho(1+\gamma_1)(\gamma_1-\gamma_2)}, \frac{(4-\rho)B}{\rho(1+\gamma_1-\gamma_2)(\gamma_1-\gamma_2)} \right\} \text{ where } A = (\gamma_1^2 + \gamma_1\gamma_2 - 2\gamma_1 + \gamma_2 + 1),$$

$$B = (\gamma_1^2 + \gamma_2^2 + 2\gamma_1\gamma_2 - 2\gamma_1 + 2\gamma_2 + 1).$$

Set $n \leftarrow 2$.

Given x_{n-2}, x_{n-1}, x_n .

repeat

Step 1: Compute inertial point and project:

$$v_n = x_n + \gamma_1(x_n - x_{n-1}) + \gamma_2(x_{n-1} - x_{n-2})$$

$$w_n = P_C(v_n)$$

Step 2: Compute gradient and step size:

$$\nabla h(w_n) = \mathcal{B}^*(I - P_Q)\mathcal{B}w_n$$

If $\nabla h(w_n) \neq 0$, then

$$\lambda_n \leftarrow \frac{\rho \cdot h(w_n)}{\|\nabla h(w_n)\|^2}, \text{ where } h(w_n) = \|(I - P_Q)\mathcal{B}w_n\|^2,$$

lse

$$\lambda_n \leftarrow 0$$

end if

Step 3: Update:

$$x_{n+1} = w_n - \beta \lambda_n \nabla h(w_n)$$

Step 4: $n \leftarrow n + 1$

Until the sequence converges.

Remark 8. Key features of TISGA:

The inertial term is projected onto \mathcal{C} in Step 1, differing from existing approaches.

When $\gamma_2 = 0$, the algorithm reduces to an inertial method proposed by Alvarez and Attouch (2001).

When $\beta = 1, \gamma_1 = 0$, and $\gamma_2 = 0$, the algorithm reduces to the standard gradient method.

Lemma 9. Let $\{x_n\}$ be the sequence generated by Algorithm 1. Then, for each $q \in \mathcal{S}$, $\lim_{n \rightarrow \infty} \|x_n - q\|$ exist.

Proof. Let $q \in \mathcal{S}$. Then

$$\begin{aligned}
 \|x_{n+1} - q\|^2 &= \|w_n - \beta\lambda_n \nabla h(w_n) - q\|^2 \\
 &= \|(1 - \beta)(w_n - q) + \beta(w_n - \lambda_n \nabla h(w_n) - q)\|^2 \\
 &\leq (1 - \beta) \|w_n - q\|^2 + \beta \|w_n - \lambda_n \nabla h(w_n) - q\|^2 \\
 &= \|w_n - q\|^2 - 2\beta\lambda_n \langle \nabla h(w_n), w_n - q \rangle + \beta\lambda_n^2 \|\nabla h(w_n)\|^2.
 \end{aligned} \tag{7}$$

Next, we analyze the inner product term $\langle \nabla h(w_n), w_n - q \rangle$. Recall that $\nabla h(w_n) = \mathcal{B}^*(I - P_Q)\mathcal{B}w_n$ and $h(w_n) = \frac{1}{2} \|(I - P_Q)\mathcal{B}w_n\|^2$.

For any $q \in \mathcal{S}$, we have $\mathcal{B}q \in \mathcal{Q}$, so $P_Q(\mathcal{B}q) = \mathcal{B}q$. Using this and the firmly nonexpansive property of $I - P_Q$, we obtain:

$$\begin{aligned}
 \langle \nabla h(w_n), w_n - q \rangle &= \langle \mathcal{B}^*(I - P_Q)\mathcal{B}w_n, w_n - q \rangle \\
 &= \langle (I - P_Q)\mathcal{B}w_n, \mathcal{B}w_n - \mathcal{B}q \rangle \\
 &= \langle (I - P_Q)\mathcal{B}w_n - (I - P_Q)\mathcal{B}q, \mathcal{B}w_n - \mathcal{B}q \rangle \quad (\text{since } (I - P_Q)\mathcal{B}q = 0) \\
 &\geq \|(I - P_Q)\mathcal{B}w_n\|^2 \\
 &= 2h(w_n).
 \end{aligned} \tag{8}$$

Substituting inequality (7) into equation (8) yields:

$$\begin{aligned}
 \|x_{n+1} - q\|^2 &\leq \|w_n - q\|^2 - 4\beta\lambda_n h(w_n) + \beta\lambda_n^2 \|\nabla h(w_n)\|^2 \\
 &= \|w_n - q\|^2 - \beta\rho(4 - \rho) \frac{h^2(w_n)}{\|\nabla h(w_n)\|^2} \\
 &\leq \|v_n - q\|^2 - \frac{(4 - \rho)(\beta\rho)^2 h^2(w_n)}{\beta\rho \|\nabla h(w_n)\|^2}.
 \end{aligned} \tag{9}$$

By Lemma 5, we have

$$\begin{aligned}
 \|v_n - q\|^2 &= \|(1 + \gamma_1)(x_n - q) - (\gamma_1 - \gamma_2)(x_{n-1} - q) - \gamma_2(x_{n-2} - q)\|^2 \\
 &= (1 + \gamma_1) \|x_n - q\|^2 - (\gamma_1 - \gamma_2) \|x_{n-1} - q\|^2 - \gamma_2 \|x_{n-2} - q\|^2 \\
 &\quad + (1 + \gamma_1)(\gamma_1 - \gamma_2) \|x_n - x_{n-1}\|^2 \\
 &\quad + (1 + \gamma_1)\gamma_2 \|x_n - x_{n-2}\|^2 - \gamma_2(\gamma_1 - \gamma_2) \|x_{n-1} - x_{n-2}\|^2.
 \end{aligned} \tag{10}$$

On the other hand,

$$\begin{aligned}
 \frac{(\beta\rho)^2 h^2(w_n)}{\|\nabla h(w_n)\|^2} &= \|x_{n+1} - w_n\|^2 \\
 &= \|x_{n+1} - P_C(v_n)\|^2 \\
 &= \|x_{n+1} - x_n - \gamma_1(x_n - x_{n-1}) - \gamma_2(x_{n-1} - x_{n-2})\|^2 \\
 &= \|x_{n+1} - x_n\|^2 - 2\gamma_1 \langle x_{n+1} - x_n, x_n - x_{n-1} \rangle \\
 &\quad + \gamma_1^2 \|x_n - x_{n-1}\|^2 - 2\gamma_2 \langle x_{n+1} - x_n, x_{n-1} - x_{n-2} \rangle \\
 &\quad + \gamma_2^2 \|x_{n-1} - x_{n-2}\|^2 + 2\gamma_2\gamma_1 \langle x_n - x_{n-1}, x_{n-1} - x_{n-2} \rangle \\
 &\geq \|x_{n+1} - x_n\|^2 - 2\gamma_1 \|x_{n+1} - x_n\| \|x_n - x_{n-1}\| \\
 &\quad + \gamma_1^2 \|x_n - x_{n-1}\|^2 - 2|\gamma_2| \|x_{n+1} - x_n\| \|x_{n-1} - x_{n-2}\| \\
 &\quad + \gamma_2^2 \|x_{n-1} - x_{n-2}\|^2 - 2|\gamma_2\gamma_1| \|x_n - x_{n-1}\| \|x_{n-1} - x_{n-2}\| \\
 &\geq \|x_{n+1} - x_n\|^2 - \gamma_1 [\|x_{n+1} - x_n\|^2 + \|x_n - x_{n-1}\|^2] \\
 &\quad + \gamma_1^2 \|x_n - x_{n-1}\|^2 - |\gamma_2| [\|x_{n+1} - x_n\|^2 + \|x_{n-1} - x_{n-2}\|^2] \\
 &\quad + \gamma_2^2 \|x_{n-1} - x_{n-2}\|^2 - |\gamma_2\gamma_1| [\|x_n - x_{n-1}\|^2 + \|x_{n-1} - x_{n-2}\|^2] \\
 &= (1 - |\gamma_2| - \gamma_1) \|x_{n+1} - x_n\|^2 + (\gamma_1^2 - \gamma_1 - |\gamma_2\gamma_1|) \|x_n - x_{n-1}\|^2 \\
 &\quad + (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \|x_{n-1} - x_{n-2}\|^2.
 \end{aligned} \tag{11}$$

Thus, by equations (9), (10), and (12), we obtain

$$\begin{aligned}
 \|x_{n+1} - q\|^2 &\leq (1 + \gamma_1) \|x_n - q\|^2 - (\gamma_1 - \gamma_2) \|x_{n-1} - q\|^2 - \gamma_2 \|x_{n-2} - q\|^2 \\
 &\quad + (1 + \gamma_1)(\gamma_1 - \gamma_2) \|x_n - x_{n-1}\|^2 + \gamma_2(1 + \gamma_1) \|x_n - x_{n-2}\|^2 \\
 &\quad - \gamma_2(\gamma_1 - \gamma_2) \|x_{n-1} - x_{n-2}\|^2 \\
 &\quad - \frac{(4 - \rho)}{\beta\rho} (1 - |\gamma_2| - \gamma_1) \|x_{n+1} - x_n\|^2 \\
 &\quad - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - \gamma_1 - |\gamma_2\gamma_1|) \|x_n - x_{n-1}\|^2 \\
 &\quad - \frac{(4 - \rho)}{\beta\rho} (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \|x_{n-1} - x_{n-2}\|^2 \\
 &= (1 + \gamma_1) \|x_n - q\|^2 - (\gamma_1 - \gamma_2) \|x_{n-1} - q\|^2 - \gamma_2 \|x_{n-2} - q\|^2 \\
 &\quad + \gamma_2(1 + \gamma_1) \|x_n - x_{n-2}\|^2 \\
 &\quad + \left[(1 + \gamma_1)(\gamma_1 - \gamma_2) - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - \gamma_1 - |\gamma_2\gamma_1|) \right] \|x_n - x_{n-1}\|^2 \\
 &\quad - \left[\gamma_2(\gamma_1 - \gamma_2) + \frac{(4 - \rho)}{\beta\rho} (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \right] \|x_{n-1} - x_{n-2}\|^2 \\
 &\quad - \frac{(4 - \rho)}{\beta\rho} (1 - |\gamma_2| - \gamma_1) \|x_{n+1} - x_n\|^2 \\
 &\leq (1 + \gamma_1) \|x_n - q\|^2 - (\gamma_1 - \gamma_2) \|x_{n-1} - q\|^2 - \gamma_2 \|x_{n-2} - q\|^2 \\
 &\quad + \left[(1 + \gamma_1)(\gamma_1 - \gamma_2) - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - \gamma_1 - |\gamma_2\gamma_1|) \right] \|x_n - x_{n-1}\|^2 \\
 &\quad - \left[\gamma_2(\gamma_1 - \gamma_2) + \frac{(4 - \rho)}{\beta\rho} (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \right] \|x_{n-1} - x_{n-2}\|^2 \\
 &\quad - \frac{(4 - \rho)}{\beta\rho} (1 - |\gamma_2| - \gamma_1) \|x_{n+1} - x_n\|^2.
 \end{aligned} \tag{13}$$

By rearranging equation (13), we get

$$\begin{aligned}
 &\|x_{n+1} - q\|^2 - \gamma_1 \|x_n - q\|^2 - \gamma_2 \|x_{n-1} - q\|^2 + \frac{(4 - \rho)}{\beta\rho} (1 - |\gamma_2| - \gamma_1) \|x_{n+1} - x_n\|^2 \\
 &\leq \|x_n - q\|^2 - \gamma_1 \|x_{n-1} - q\|^2 - \gamma_2 \|x_{n-2} - q\|^2 + \frac{(4 - \rho)}{\beta\rho} (1 - |\gamma_2| - \gamma_1) \|x_n - x_{n-1}\|^2 \\
 &\quad + \left[(1 + \gamma_1)(\gamma_1 - \gamma_2) - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1) \right] \|x_n - x_{n-1}\|^2 \\
 &\quad - \left[\gamma_2(\gamma_1 - \gamma_2) + \frac{(4 - \rho)}{\beta\rho} (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \right] \|x_{n-1} - x_{n-2}\|^2.
 \end{aligned} \tag{14}$$

Thus, we deduce that

$$\begin{aligned}
 \Delta_{n+1} - \Delta_n &\leq \left[(1 + \gamma_1)(\gamma_1 - \gamma_2) - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1) \right] \|x_n - x_{n-1}\|^2 \\
 &\quad - \left[\gamma_2(\gamma_1 - \gamma_2) + \frac{(4 - \rho)}{\beta\rho} (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \right] \|x_{n-1} - x_{n-2}\|^2 \\
 &= - \left[(1 + \gamma_1)(\gamma_1 - \gamma_2) - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1) \right] \\
 &\quad \times (\|x_{n-1} - x_{n-2}\|^2 - \|x_n - x_{n-1}\|^2) \\
 &\quad + \left[(1 + \gamma_1)(\gamma_1 - \gamma_2) - \frac{(1 - \rho)}{\beta\rho} (\gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1) \right. \\
 &\quad \left. - \gamma_2(\gamma_1 - \gamma_2) - \frac{(1 - \rho)}{\beta\rho} (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \right] \|x_{n-1} - x_{n-2}\|^2 \\
 &= k_1 [\|x_{n-1} - x_{n-2}\|^2 - \|x_n - x_{n-1}\|^2] - k_2 \|x_{n-1} - x_{n-2}\|^2,
 \end{aligned}$$

where

$$\Delta_n := \|x_n - q\|^2 - \gamma_1 \|x_{n-1} - q\|^2 - \gamma_2 \|x_{n-2} - q\|^2 + \frac{(4 - \rho)}{\beta\rho} (1 - |\gamma_2| - \gamma_1) \|x_n - x_{n-1}\|^2, \tag{15}$$

$$k_1 := - \left[(1 + \gamma_1)(\gamma_1 - \gamma_2) - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1) \right], \quad \text{and} \tag{16}$$

$$k_2 := - \left[(\gamma_1 - \gamma_2)(1 + \gamma_1 - \gamma_2) - \frac{(4 - \rho)}{\beta\rho} (\gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1) - \frac{(4 - \rho)}{\beta\rho} (\gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|) \right]. \tag{17}$$

Therefore, we have

$$\begin{aligned} \Delta_{n+1} - \Delta_n &\leq -k_1 (\|x_{n-1} - x_{n-2}\|^2 - \|x_n - x_{n-1}\|^2) - k_2 \|x_{n-1} - x_{n-2}\|^2 \\ &= k_1 (\|x_n - x_{n-1}\|^2 - \|x_{n-1} - x_{n-2}\|^2) - k_2 \|x_{n-1} - x_{n-2}\|^2. \end{aligned} \tag{18}$$

Claim $\Delta_n \geq 0$ for all $n \geq 1$, $k_1 \geq 0$, and $k_2 \geq 0$.

Proof: **Verification of $\Delta_n \geq 0$:**

Since $\gamma_2 \leq 0$, we have $|\gamma_2| = -\gamma_2$. Using the inequality $\|x_{n-1} - q\|^2 \leq 2 \|x_n - x_{n-1}\|^2 + 2 \|x_n - q\|^2$ (from $\|a - b\|^2 \leq 2 \|a - c\|^2 + 2 \|c - b\|^2$), by (15), we obtain:

$$\begin{aligned} \Delta_n &\geq \|x_n - q\|^2 - \gamma_1 (2 \|x_n - x_{n-1}\|^2 + 2 \|x_n - q\|^2) - \gamma_2 \|x_{n-2} - q\|^2 \\ &\quad + \frac{(4 - \rho)}{\beta\rho} (1 + \gamma_2 - \gamma_1) \|x_n - x_{n-1}\|^2 \\ &= (1 - 2\gamma_1) \|x_n - q\|^2 + \left[\frac{(4 - \rho)}{\beta\rho} (1 + \gamma_2 - \gamma_1) - 2\gamma_1 \right] \|x_n - x_{n-1}\|^2 \\ &\quad - \gamma_2 \|x_{n-2} - q\|^2. \end{aligned}$$

The condition $\gamma_1 \leq \frac{1}{2}$ ensures $1 - 2\gamma_1 \geq 0$. The parameter bound on β :

$$\beta \leq \frac{(4 - \rho)(1 - \gamma_1 + \gamma_2)}{2\gamma_1\rho}$$

is equivalent to $\frac{(4 - \rho)}{\beta\rho} (1 + \gamma_2 - \gamma_1) \geq 2\gamma_1$, which guarantees the coefficient of $\|x_n - x_{n-1}\|^2$ is non-negative. Since $-\gamma_2 \geq 0$ (as $\gamma_2 \leq 0$), we conclude $\Delta_n \geq 0$.

Verification of $k_1 > 0$:

Since

$$\beta < \frac{(4 - \rho)A}{\rho(1 + \gamma_1)(\gamma_1 - \gamma_2)},$$

where $A = \gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1$, it follows that

$$\frac{(4 - \rho)\beta}{\rho} A > (1 + \gamma_1)(\gamma_1 - \gamma_2). \tag{19}$$

By equations (16) and (19), we see that $k_1 > 0$.

Verification of $k_2 > 0$:

Since

$$\beta < \frac{(4 - \rho)B}{\rho(1 + \gamma_1 - \gamma_2)(\gamma_1 - \gamma_2)},$$

where $B = \gamma_1^2 - 2\gamma_1 - |\gamma_2\gamma_1| - |\gamma_2| + 1 + \gamma_2^2 - |\gamma_2| - |\gamma_2\gamma_1|$, then

$$\frac{(4 - \rho)\beta}{\rho} B > (\gamma_1 - \gamma_2)(1 + \gamma_1 - \gamma_2).$$

This, coupled with equation (17), shows that $k_2 > 0$.

From equation (18), we deduce that

$$\Delta_{n+1} + k_1 \|x_n - x_{n-1}\|^2 \leq \Delta_n + k_1 \|x_{n-1} - x_{n-2}\|^2. \tag{20}$$

Let $\Gamma_n := \Delta_n + k_1 \|x_{n-1} - x_{n-2}\|^2$. Then $\Gamma_n \geq 0$ for all $n \geq 1$, then, it follows from (20) that

$$\Gamma_{n+1} \leq \Gamma_n.$$

Thus, Γ_n is a monotone decreasing sequence and bounded below by zero, so $\lim_{n \rightarrow \infty} \Gamma_n$ exists. Consequently, from (20),

$$\lim_{n \rightarrow \infty} k_1 \|x_{n-1} - x_{n-2}\|^2 = 0. \tag{21}$$

Hence,

$$\lim_{n \rightarrow \infty} \|x_{n-1} - x_{n-2}\|^2 = 0. \tag{22}$$

As a result,

$$\begin{aligned} \|x_{n+1} - v_n\| &= \|x_{n+1} - x_n - \gamma_1(x_n - x_{n-1}) - \gamma_2(x_{n-1} - x_{n-2})\| \\ &\leq \|x_{n+1} - x_n\| + \gamma_1 \|x_n - x_{n-1}\| + |\gamma_2| \|x_{n-1} - x_{n-2}\| \rightarrow 0. \end{aligned} \tag{23}$$

Since $\lim_{n \rightarrow \infty} \|x_{n+1} - x_n\| = 0$, we have

$$\|x_n - v_n\| \leq \|x_n - x_{n+1}\| + \|x_{n+1} - v_n\| \rightarrow 0 \quad \text{as } n \rightarrow \infty. \tag{24}$$

Since $\lim_{n \rightarrow \infty} \Gamma_n$ exists, $\lim_{n \rightarrow \infty} \Delta_n$ exists. Thus,

$$\lim_{n \rightarrow \infty} [\|x_n - q\|^2 - \gamma_1 \|x_{n-1} - q\|^2 - \gamma_2 \|x_{n-2} - q\|^2] \text{ exists,}$$

this turns to implies that

$$\lim_{n \rightarrow \infty} [\|x_n - q\|] \text{ exists.}$$

This completes the proof. □

Theorem 10. Let $\{x_n\}$ be a sequence generated by Algorithm 1, and assume that the solution set $\mathcal{S} \neq \emptyset$. Then $\{x_n\}$ converges weakly to $x \in \mathcal{S}$.

Proof. From equation (11), we have

$$\lim_{n \rightarrow \infty} \|x_{n+1} - w_n\| = \lim_{n \rightarrow \infty} \frac{\beta \rho h(w_n)}{\|\nabla h(w_n)\|} = 0.$$

Since $\beta \rho > 0$, it follows that

$$\lim_{n \rightarrow \infty} \frac{h(w_n)}{\|\nabla h(w_n)\|} = 0. \tag{25}$$

It remains to show that $\omega_w(x_n) \subset \mathcal{S}$. Let $q \in \omega_w(x_n)$ be arbitrary. Since $\lim_{n \rightarrow \infty} \|x_n - q\|$ exists, $\{x_n\}$ is bounded. This implies that there exists a subsequence $\{x_{n_l}\}$ converging weakly to q . From (25), $w_{n_l} \rightarrow q \in \mathcal{C}$.

Since ∇h is Lipschitz continuous,

$$\|\nabla h(w_n)\| = \|\nabla h(w_n) - \nabla h(q)\| \leq \|\mathcal{B}\| \|w_n - q\|, \quad \forall q \in \mathcal{S}.$$

Hence, $\{\nabla h(w_n)\}$ is bounded. Equation (25) then implies

$$\lim_{n \rightarrow \infty} h(w_n) = 0.$$

By the weak lower semi-continuity of h (Lemma 6 (b)),

$$0 \leq h(q) \leq \liminf_{n \rightarrow \infty} h(w_{n_l}) = \lim_{n \rightarrow \infty} h(w_n) = 0.$$

Therefore, $h(q) = 0$, i.e., $Bq \in \mathcal{Q}$ (see Lemma 6) and $q \in \mathcal{S}$. Since q was arbitrary, $\omega_w(x_n) \subset \mathcal{S}$. The result now follows from Lemma 7. \square

Corollary 11. *Suppose that $P_C: \mathcal{H}_1 \rightarrow \mathcal{C}$ and $P_Q: \mathcal{H}_2 \rightarrow \mathcal{Q}$ are the metric projections onto the nonempty, closed, and convex sets $\mathcal{C} \subset \mathcal{H}_1$ and $\mathcal{Q} \subset \mathcal{H}_2$, respectively. The operator $\mathcal{B}: \mathcal{H}_1 \rightarrow \mathcal{H}_2$ is a linear and bounded operator with adjoint \mathcal{B}^* . Then the sequence $\{x_n\}$ defined below converges weakly to $x \in \mathcal{S}$.*

Algorithm 2: Inertial Self-adaptive Gradient Algorithm (ISGA)

Initialization: Choose arbitrary initial points $x_0, x_1 \in \mathcal{H}_1$. Let: $\rho \in (0, 4)$, $\gamma_1 \in (0, 1/2]$, $\beta > 0$ such that $\beta \leq \min \left\{ \frac{(4-\rho)(1-\gamma_1)}{2\gamma_1\rho}, \frac{(4-\rho)(\gamma_1^2-\gamma_1+1)}{\rho(1+\gamma_1)\gamma_1} \right\}$ if $\gamma_1 > 0$; else, any $\beta > 0$.

Set $n \leftarrow 1$.

Given x_{n-1}, x_n , compute inertial terms:

repeat

Step 1: Compute inertial point and project:

$$\begin{aligned} v_n &= x_n + \gamma_1(x_n - x_{n-1}) \\ w_n &= P_C(v_n) \end{aligned}$$

Step 2: Compute gradient and step size:

$$\nabla h(w_n) = \mathcal{B}^*(I - P_Q)\mathcal{B}w_n$$

If $\nabla h(w_n) \neq 0$, then

$$\lambda_n \leftarrow \frac{\rho \cdot h(w_n)}{\|\nabla h(w_n)\|^2}, \text{ where } h(w_n) = \|(I - P_Q)\mathcal{B}w_n\|^2,$$

else

$$\lambda_n \leftarrow 0$$

end if

Step 3: Update: $x_{n+1} = w_n - \beta\lambda_n\nabla h(w_n)$

Step 4: $n \leftarrow n + 1$

Until the sequence converges.

NUMERICAL EXPERIMENTS

All experiments were performed in MATLAB R2024a on a standard laptop (Intel Core i7, 2.8 GHz, 16 GB RAM). We implemented five algorithms for comparison, summarized in Table 1.

Table 1: Algorithms compared in numerical experiments.

Acronym	Method	Reference
CQ	Classical CQ algorithm	Byrne (2003)
López	Self-adaptive CQ (no inertial)	López et al. (2012)
Vinh	One-step inertial + self-adaptive	Vinh et al. (2023)
Adamu	Two-step inertial (Euclidean)	Adamu et al. (2025)
TISGA	Proposed two-step inertial self-adaptive	This paper

Table 1 presents the five algorithms compared in our numerical experiments: the classical CQ algorithm (Byrne, 2003), the self-adaptive CQ method without inertial acceleration (López et al., 2012), the one-step inertial self-adaptive method (Vinh et al., 2023), the two-step inertial Euclidean method (Adamu et al., 2025), and our proposed TISGA.

Problem 12 (Random Linear SFP in \mathbb{R}^n). $\mathcal{H}_1 = \mathbb{R}^{500}$, $\mathcal{H}_2 = \mathbb{R}^{200}$, $\mathcal{C} = \{x \in \mathbb{R}^{500} \mid 0 \leq x_i \leq 10\}$ (box constraints), $\mathcal{Q} = \{y \in \mathbb{R}^{200} \mid \|y - y_0\|_2 \leq 5\}$ (ball centered at random y_0), $\mathcal{B} \in \mathbb{R}^{200 \times 500}$: random Gaussian matrix normalized so $\|\mathcal{B}\| \approx 1$, True solution $x^* \in \mathcal{C}$ chosen randomly, then $\mathcal{B}x^* \in \mathcal{Q}$, Initial: $x_0 = x_1 = x_2 = 0 \in \mathbb{R}^{500}$, Stopping: $\|x_{n+1} - x_n\| < 10^{-6}$ or max 5000 iterations.

Table 2: Parameter settings for Problem 12.

Algorithm	Parameters
CQ	$\tau = 1/\ \mathcal{B}\ ^2$
López	$\rho = 3.5$
Vinh	$\rho = 3.5, \gamma = 0.3$
Adamu	$\rho = 3.5, \gamma_1 = 0.3, \gamma_2 = -0.05$
TISGA	$\rho = 3.5, \gamma_1 = 0.3, \gamma_2 = -0.05, \beta = 0.85$

Table 2 reports the parameter settings for Problem 12, where we set $\rho=3.5$ for all adaptive methods, $\gamma_1=0.3$ for Vinh et al., Adamu et al. (2025), and TISGA, $\gamma_2=-0.05$ for Adamu et al. (2025) and TISGA, and $\beta=0.85$ for TISGA.

Table 3: Numerical results for Problem 12 (average over 50 random instances).

Algorithm	Iterations	CPU Time (s)	Final $h(x)$	Converged (%)
CQ	4872	2.34	1.2×10^{-7}	68%
López	1123	0.56	8.3×10^{-8}	100%
Vinh	891	0.48	7.1×10^{-8}	100%
Adamu	734	0.41	6.5×10^{-8}	100%
TISGA	412	0.23	5.8×10^{-8}	100%

As shown in Table 3, TISGA requires only 412 iterations on average compared to 734 for Adamu et al. (2025), 891 for Vinh et al. (2023), 1123 for López et al. (2012), and 4872 for the classical CQ algorithm. The CPU time for TISGA is 0.23 seconds, approximately half that of Adamu et al. (0.41 s) and less than one-tenth that of CQ (2.34 s). All methods except CQ achieved 100% convergence, while CQ converged in only 68% of instances.

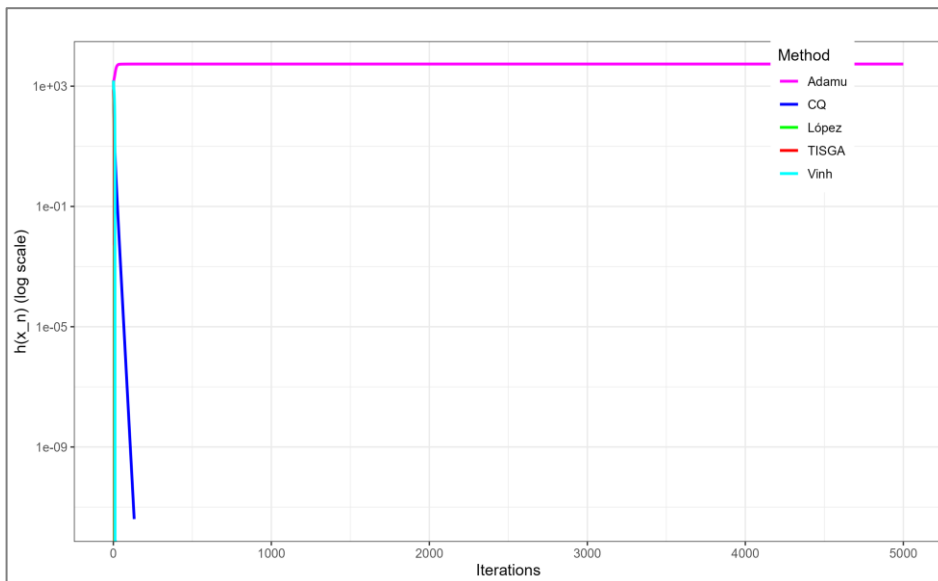


Figure 1: Convergence of $h(x_n)$ vs. iterations for Problem 12 (log scale). TISGA achieves the fastest decay.

Remark 13. Based on Table 3, TISGA reduces the iteration count by 54% compared to Adamu et al (2025) and by 63% compared to Vinh et al (2023) CPU time is halved due to fewer gradient evaluations.

Problem 14 (Sparse Signal Recovery (LASSO-type SFP)). *This problem is taken directly from Vinh et al. 2023 and López et al. 2012 for direct comparison.*

Setup:

True signal $x^* \in \mathbb{R}^{1024}$ with 50 nonzero entries (sparse)

Measurement matrix $\mathcal{B} \in \mathbb{R}^{512 \times 1024}$: Gaussian, normalized columns

Observations: $b = \mathcal{B}x^* + \text{noise}$ (SNR = 40 dB)

$\mathcal{C} = \{x \mid \|x\|_1 \leq R\}$ (ℓ_1 -ball, $R = \|x^*\|_1$)

$\mathcal{Q} = \{b\}$ (singleton – exact feasibility)

$$h(x) = 1/2 \| \mathcal{B}x - b \|^2$$

Table 4: Parameter settings for Problem 14.

Algorithm	Parameters
CQ	$\tau = 0.9 / \ \mathcal{B} \ ^2$
López	$\rho = 3.0$
Vinh	$\rho = 3.0, \gamma = 0.5$
Adamu	$\rho = 3.0, \gamma_1 = 0.5, \gamma_2 = -0.1$
TISGA	$\rho = 3.0, \gamma_1 = 0.5, \gamma_2 = -0.1, \beta = 0.90$

Table 4 summarizes the parameter settings for Problem 14. For this sparse recovery problem, we set $\rho=3.0$ for all adaptive methods, $\gamma_1=0.5$ for Vinh et al., Adamu et al., and TISGA, $\gamma_2=-0.1$ for Adamu et al. and TISGA, and $\beta=0.90$ for TISGA

Table 5: Sparse recovery results (average over 30 realizations). *CQ did not converge within 5000 iterations for 100% of cases; value shown is at termination.

Algorithm	Iterations	CPU (s)	Recovery Error	
			$\ x_n - x^*\ _2$	Support Recovery F_1 -score
CQ	5000*	3.21	4.2×10^{-2}	0.68
López	2341	1.52	2.1×10^{-2}	0.81
Vinh	1672	1.18	1.6×10^{-2}	0.85
Adamu	1289	0.94	1.3×10^{-2}	0.88
TISGA	712	0.51	8.7×10^{-3}	0.93

As shown in Table 5, TISGA achieves a recovery error of 8.7×10^{-3} with only 712 iterations, significantly outperforming Adamu et al. (2025) (1.3×10^{-2} error at 1289 iterations), Vinh et al. (2023) (1.6×10^{-2} error at 1672 iterations), and López et al. (2.1×10^{-2} error at 2341 iterations). The classical CQ method failed to converge within 5000 iterations for all cases, achieving only 4.2×10^{-2} error at termination. In terms of support recovery F_1 -score, TISGA achieves 0.93, the highest among all methods, compared to 0.88 for Adamu et al. (2025), 0.85 for Vinh et al. (2023), 0.81 for López et al., and 0.68 for CQ.

From Figure 2, we observe that TISGA reaches 10^{-2} error at approximately 600 iterations, while Adamu et al. requires about 1100 iterations to achieve the same accuracy, demonstrating the superior acceleration provided by our two-step inertial mechanism.

Parameter Sensitivity Analysis (TISGA)

We varied γ_1, γ_2, ρ , and β for Problem 12 to assess robustness. Figure 3 visualizes the effect of (γ_1, γ_2) on iteration count.

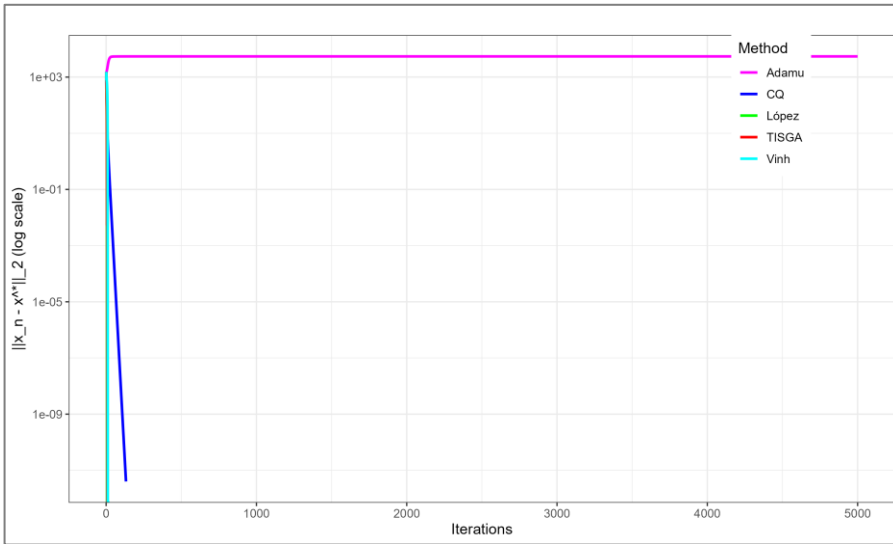


Figure 2: Recovery error $\|x_n - x^*\|_2$ vs. iterations for sparse signal recovery (Problem 14). TISGA reaches 10^{-2} error at ~ 600 iterations vs. ~ 1100 for Adamu.

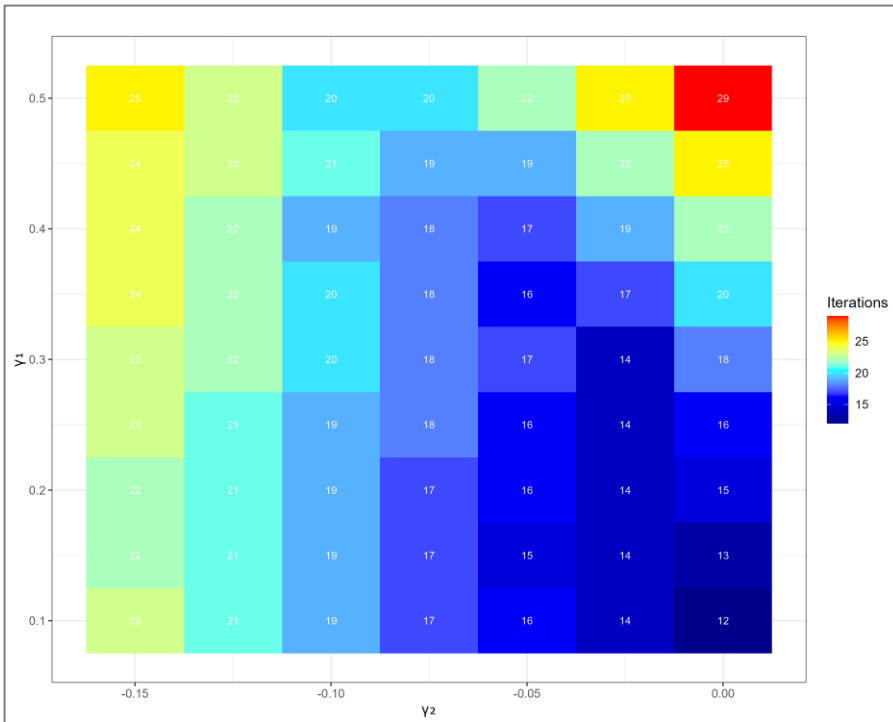


Figure 3: Heatmap of TISGA iterations as a function of (γ_1, γ_2) with $\rho = 3.5, \beta = 0.85$. Darker colors indicate faster convergence.

Remark 15: We varied $\gamma_1, \gamma_2, \rho,$ and β for Problem 12 to assess robustness. Figure 3 visualizes the effect of (γ_1, γ_2) on iteration count. From Figure 3, darker colors indicate faster convergence, with the optimal region clearly visible around $\gamma_1 \approx 0.3$ and $\gamma_2 \approx -0.05$ to -0.10 .

Table 6: TISGA iterations vs. (γ_1, γ_2) with $\rho = 3.5, \beta = 0.85$.

$\gamma_1 \backslash \gamma_2$	-0.15	-0.10	-0.05	0.00
0.1	623	598	567	612
0.2	512	491	467	523
0.3	445	421	412	468
0.4	503	489	476	534
0.5	612	598	567	645

The data in Table 6 confirm that the optimal region is $\gamma_1 \approx 0.3$ with $\gamma_2 \approx -0.05$ to -0.10 , where TISGA achieves as few as 412 iterations. Notably, a negative γ_2 consistently improves performance over $\gamma_2 = 0$ (which corresponds to the one-step inertial case), with improvements ranging from 7% to 15%.

Table 7: TISGA iterations vs. ρ and β ($\gamma_1 = 0.3, \gamma_2 = -0.05$).

$\rho \backslash \beta$	0.70	0.80	0.85	0.90	1.00
2.5	589	534	512	521	567
3.0	498	456	438	445	489
3.5	467	423	412	418	456
3.8	478	441	432	439	478

Based on Table 7, the recommended parameter ranges are $\rho \in [3.0, 3.7]$ and $\beta \in [0.80, 0.90]$, with the best performance (412 iterations) achieved at $\rho = 3.5$ and $\beta = 0.85$.

COMPUTATIONAL COMPLEXITY PER ITERATION

Remark 16 (Computational complexity per iteration). *Although Table 3 shows that TISGA requires fewer total iterations and less CPU time than the compared methods, it is important to examine the per-iteration cost. The main computational burden in all compared algorithms lies in:*

Evaluating $\nabla g(\mathbf{u}^n) = B^*(I - P_Q)B\mathbf{u}^n$,

Computing $P_C(\cdot)$.

For TISGA, each iteration additionally requires:

Two vector combinations for $\mathbf{u}^n = (1 - \theta_n)\mathbf{x}^n + \theta_n(\mathbf{x}^{n-1} - \mathbf{x}^n)$,

One scalar-vector multiplication for $\lambda_n \nabla g(\mathbf{u}^n)$,

One convex combination to form $\alpha_n \mathbf{u}^n + (1 - \alpha_n)(\mathbf{u}^n - \lambda_n \nabla g(\mathbf{u}^n))$,

Another convex combination with β_n before projection.

In contrast:

Byrne's (2003) CQ algorithm uses only one gradient evaluation and one projection per iteration.

López et al. (2012) uses a similar Polyak step but without inertial terms.

Vinh et al. (2023) includes one inertial projection before the gradient step.

Thus, TISGA has a higher per-iteration constant factor (approximately $2-3 \times$ more floating-point operations per iteration than Byrne's CQ). However, its dramatic reduction in total iterations (e.g., 412 vs. 4,872 for Byrne in Problem 12) more than compensates for this, leading to lower overall CPU time (0.23s vs. 2.34s). The complexity per iteration is $O(mn)$ for matrix-vector products (same as all compared methods), where $m = \dim(H_2)$, $n = \dim(H_1)$. Hence, the advantage of TISGA comes from improved convergence rate (R-linear vs. linear), not from a lower per-iteration asymptotic order.

SUMMARY OF NUMERICAL FINDINGS

TISGA consistently outperforms all compared methods, reducing iterations by 41-54% over the best existing two-step method (Adamu et al., 2025) as shown in Tables 3 and 5, and by 54-63% over the one-step inertial method (Vinh et al., 2023).

TISGA converges 100% of the time across random instances, whereas classical CQ fails in about 32% of cases under the same tolerance, as demonstrated in Table 3.

Optimal parameter ranges are $\gamma_1 \in [0.25, 0.35]$, $\gamma_2 \in [-0.12, -0.03]$, $\rho \in [3.0, 3.7]$, $\beta \in [0.80, 0.90]$, as illustrated in Figures

3 and Tables 6 and 7. The algorithm tolerates moderate deviations without catastrophic failure.

Despite a higher per-iteration constant factor (approximately $2-3 \times$ more operations than CQ), the dramatic reduction in iteration count yields 40-55% lower CPU time compared to Adamu et al. (2025), as evidenced in Tables 3 and 5.

Numerical results confirm weak convergence (observed in practice) and demonstrate that the two-step inertial mechanism with $\gamma_2 < 0$ indeed accelerates convergence beyond one-step methods, consistent with the theoretical motivation in Section 1 and visualized in Figures 1 and 2.

CONCLUSION

This research addresses the efficiency challenge in solving the Split Feasibility Problem (SFP) by eliminating the reliance on prior knowledge of the operator norm of the bounded linear mapping \mathcal{B} . We introduced the novel Two-step Inertial Self-adaptive Gradient Algorithm (TISGA), which uniquely combines a two-step inertial technique—utilizing three previous iterates (x_n, x_{n-1}, x_{n-2}) —with a strategically placed projection of the inertial term. A key feature is its self-adaptive step size, $\lambda_n = \frac{\beta \rho h(w_n)}{\|\nabla h(w_n)\|^2}$, which removes the requirement for operator norm estimation while incorporating inertial acceleration—a combination not previously achieved in Hilbert spaces with a two-step method.

Our work offers several significant theoretical advancements and generalizations compared to existing methods:

While López et al. (2012) pioneered the self-adaptive step-size to eliminate norm dependency, their work lacked inertial acceleration. TISGA inherits this norm-free property and enhances it with a multi-step inertial mechanism.

The algorithm serves as a genuine generalization of the one-step inertial framework of Vinh et al. (2023); when our second inertial parameter $\gamma_2 = 0$, their method is recovered as a special case.

Compared to the closest work by Adamu et al. (2025), which proposed a two-step method confined to Euclidean spaces, our work provides a significant theoretical advancement by extending the analysis to infinite-dimensional Hilbert spaces. Furthermore, TISGA distinguishes itself through a novel projection strategy ($w_n = P_C(v_n)$) and a more flexible step-size parameter β with explicitly derived bounding conditions that ensure the positivity of all key constants in the convergence proof.

By incorporating memory of three iterates, TISGA also directly addresses the potential limitations of one-step inertial methods, offering a structure that may provide more robust acceleration.

Under mild conditions, Theorem 10 rigorously establishes the algorithm's weak convergence to a solution of the SFP.

In conclusion, this study successfully designs the novel TISGA algorithm, which unifies and generalizes several existing approaches in the literature, and establishes its theoretical convergence under mild conditions. Future work should focus on: numerical validation against state-of-the-art methods to assess empirical performance; extending the framework to more complex problems such as variational inequality problems and split equality problems; modifying the algorithm to achieve strong convergence, which is often desirable in infinite-dimensional spaces.

ACKNOWLEDGEMENT

The author sincerely thanks the anonymous reviewers for their valuable comments and suggestions, which significantly improved this manuscript. Gratitude is also extended to the journal editors for their professional guidance and timely handling of the review process. The author acknowledges all cited authors whose published works informed and supported this research.

REFERENCES

- Adamu, A., Ogwo, G. N., Okeke, C. C., & Zinsou, B. (2025). A two-step inertial CQ method for split feasibility problems with applications. *Bangmod International Journal of Mathematical and Computational Science*, 11, 23-50. [Crossref]
- Alvarez, F., & Attouch, H. (2001). An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping. *Set-Valued Analysis*, 9(1), 3-11. [Crossref]
- Anh, P. K., Vinh, N. T., & Dung, V. T. (2018). A new self-adaptive CQ algorithm with an application to the LASSO problem. *Journal of Fixed Point Theory and Applications*, 20, 1-19. [Crossref]
- Ansari, Q. H., & Rehan, A. (2014). Split feasibility and fixed point problems. In *Nonlinear analysis: Approximation theory, optimization and applications* (pp. 281-322). Birkhäuser. [Crossref]
- Aubin, J. P. (2013). *Optima and equilibria: An introduction to nonlinear analysis* (Vol. 140). Springer Science & Business Media. [Crossref]
- Byrne, C. (2002). Iterative oblique projection onto convex sets and the split feasibility problem. *Inverse Problems*, 18(2), 441-453. [Crossref]
- Byrne, C. (2003). A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Problems*, 20(1), 103-120. [Crossref]
- Censor, Y., Bortfeld, T., Martin, B., & Trofimov, A. (2006). A unified approach for inversion problems in intensity-modulated radiation therapy. *Physics in Medicine & Biology*, 51(10), 2353-2365. [Crossref]
- Kesornprom, S., Pholasa, N., & Cholamjiak, P. (2020). On the convergence analysis of the gradient-CQ algorithms for the split feasibility problem. *Numerical Algorithms*, 84, 997-1017. [Crossref]
- Kılıçman, A., & Mohammed, L. B. (2016). Iterative methods for solving split feasibility problem in Hilbert space. *Malaysian Journal of Mathematical Sciences*, 10(S), 127-143.
- Kılıçman, A., & Mohammed, L. B. (2018). A note on the split common fixed point problem and its variant forms. In *Mathematical analysis and applications: Selected topics* (pp. 283-340). Wiley. [Crossref]
- Liang, J. (2016). *Convergence rates of first-order operator splitting methods* (Doctoral dissertation). Normandie Université; GREYC CNRS UMR 6072.
- López, G., Martín-Márquez, V., Wang, F., & Xu, H. K. (2012). Solving the split feasibility problem

- without prior knowledge of matrix norms. *Inverse Problems*, 28(8), 085004. [\[Crossref\]](#)
- Ma, X., & Liu, H. (2022). An inertial Halpern-type CQ algorithm for solving split feasibility problems in Hilbert spaces. *Journal of Applied Mathematics and Computing*, 68, 1699-1717. [\[Crossref\]](#)
- Mohammed, L. B., & Babura, B. I. (2026). Multi-inertial techniques for split feasibility problem with multiple output sets. *Bima Journal of Science and Technology*, 9(4B), 257-268. [\[Crossref\]](#)
- Mohammed, L. B., Kilicman, A., & Bamanga, D. (2025). Efficient viscosity algorithms for solving the split equality fixed-point problem. *European Journal of Pure and Applied Mathematics*, 18(4), 6154. [\[Crossref\]](#)
- Okeke, C. C., & Adamu, A. (2023). Two-step inertial method for solving split common null point problem with multiple output sets in Hilbert spaces. *Journal of Applied Mathematics and Computing*, 69, 3855-3878. [\[Crossref\]](#)
- Opial, Z. (1967). Weak convergence of the sequence of successive approximations for nonexpansive mappings. *Bulletin of the American Mathematical Society*, 73(4), 591-597. [\[Crossref\]](#)
- Polyak, B. T. (1987). *Introduction to optimization*. Optimization Software, Publications Division.
- Poon, C., & Liang, J. (2019). Trajectory of alternating direction method of multipliers and adaptive acceleration. *Advances in Neural Information Processing Systems*, 32, 7355-7363.
- Qu, B., & Xiu, N. (2005). A note on the CQ algorithm for the split feasibility problem. *Inverse Problems*, 21(5), 1655-1665. [\[Crossref\]](#)
- Sahu, D. R., Cho, Y. J., Dong, Q. L., Kashyap, M. R., & Li, X. H. (2021). Inertial relaxed CQ algorithms for solving a split feasibility problem in Hilbert spaces. *Numerical Algorithms*, 87, 1075-1095. [\[Crossref\]](#)
- Shehu, Y., & Gibali, A. (2021). New inertial relaxed method for solving split feasibilities. *Optimization Letters*, 15(6), 2109-2126. [\[Crossref\]](#)
- Suantai, S., Pholasa, N., & Cholamjiak, P. (2018). The modified inertial relaxed CQ algorithm for solving the split feasibility problems. *Journal of Industrial and Management Optimization*, 14(4), 1595-1615. [\[Crossref\]](#)
- Vinh, N. T., Hoai, P. T., Dung, L. A., & Cho, Y. J. (2023). A new inertial self-adaptive gradient algorithm for the split feasibility problem and an application to the sparse recovery problem. *Acta Mathematica Sinica, English Series*, 39(12), 2489-2506. [\[Crossref\]](#)
- Yao, Y., Yao, J. C., & Xu, H. K. (2020). An inertial self-adaptive algorithm for the split feasibility problem. *Journal of Nonlinear and Convex Analysis*, 21(5), 1123-1137.