Mathematical Modelling and Numerical Simulation with Applications, 2025, 5(2), 307–347

https://dergipark.org.tr/en/pub/mmnsa
ISSN Online: 2791-8564 / Open Access
https://doi.org/10.53391/mmnsa.1567457

RESEARCH PAPER

Strong convergence multi-step inertial golden ratio-based algorithms for split feasibility problems with applications

Abdulwahab Ahmad [1,2,‡], Poom Kumam [1,*,‡] and Thidaporn Seangwattana [3,‡]

[1]Center of Excellence in Theoretical and Computational Science (TaCS-CoE) & Fixed Point Research Laboratory, Room SCL 802, Science Laboratory Building, Departments of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-Uthit Road Thung Khru, Bang Mod, 10140, Bangkok, Thailand, [2]Department of Mathematics, School of Secondary Education, Sciences, Federal College of Education, Katsina, 2041, Katsina, Nigeria, [3]Faculty of Science Energy and Environment, King Mongkut's University of Technology North Bangkok, Rayong Campus (KMUTNB), 21120, Rayong, Thailand

[*] Corresponding Author
[‡] abumuhammaddm@gmail.com (Abdulwahab Ahmad); poom.kum@kmutt.ac.th (Poom Kumam); thidaporn.s@sciee.kmutnb.ac.th (Thidaporn Seangwattana)

Abstract

In this work, we construct four efficient multi-step inertial relaxed algorithms based on the monotonic adaptive step length rule, which work very effectively without requiring information about the norm of the underlying operator or a line search procedure for split feasibility problems in infinite-dimensional Hilbert spaces. The first and third are general multi-step inertial-type methods, which unify two steps of the improved version of the classical inertial term (i.e., multi-step inertial terms) and new extrapolation steps, for which the golden ratio and alternating golden ratio extrapolation steps are particular cases, respectively. These procedures significantly improve the speed of convergence of their sequences toward a solution. The second and fourth are multi-step inertial and three-term conjugate gradient-like methods, which integrate the three-term conjugate gradient-like direction, the multi-step inertial term and new extrapolation steps, for which the golden ratio and alternating golden ratio extrapolation steps are particular cases, respectively. These techniques greatly accelerate their sequences toward a solution. Under some simple and weaker assumptions, we prove the strong convergence of each of these algorithms based on the convergence of one of the two proposed algorithms with perturbations and new extrapolation techniques, namely, the beyond the golden ratio and beyond the alternating golden ratio algorithms with perturbations, to a minimum-norm solution of a split feasibility problem in infinite-dimensional real Hilbert spaces. Finally, we analyze their possible applications in classification problems for an interesting real-world dataset based on the extreme learning machine (ELM) with the $\ell_1 - \ell_2$ hybrid regularization approach and in solving constrained minimization problems in infinite-dimensional Hilbert spaces. In all the experiments, our proposed algorithms, which generalize several algorithms in the literature, comparatively achieve better performance than some related algorithms.

Keywords: Split feasibility problem; golden ratio algorithm; multi-step inertial method; three-term conjugate gradient method; classification problem

AMS 2020 Classification: 47J20; 47J25; 47J30; 65K15

1 Introduction

Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be real Hilbert spaces, $\mathcal{C} \subseteq \mathcal{H}_1$ and $\mathcal{Q} \subseteq \mathcal{H}_2$ be nonempty, closed and convex sets. The notion of split feasibility problem was initially introduced in Euclidean spaces by Censor and Elfving [1]. Its mathematical formulation is to find a point $u \in \mathcal{C}$ such that

$$\mathcal{B}u \in \mathcal{Q}, \tag{1}$$

where $\mathcal{B} : \mathcal{H}_1 \to \mathcal{H}_2$ is a bounded linear operator. The study of this problem has attracted the attention of several researchers for many years, which perhaps stems from their usefulness in dealing with many significant inverse problems arising from various real-world applications, such as X-ray tomography [2], machine learning [3], image and signal reconstructions, and jointly constrained Nash equilibrium problems [4–6], to mention but just a few. Since its introduction, several researchers have focused on developing robust and efficient iterative algorithms for its approximate solutions. Byrne [7], used the notion of a fixed point problem $u = P_{\mathcal{C}}\big(I - \tau\mathcal{B}^*(I - P_{\mathcal{Q}})\mathcal{B}\big)u$ and a particular case of a real-valued function $g : \mathcal{H}_1 \to \mathbb{R}$ defined by

$$g(u) = \frac{1}{2}||\mathcal{B}u - P_{\mathcal{Q}}\mathcal{B}u||^2, \tag{2}$$

with its L-Lipschitz continuous gradient $\nabla g = \mathcal{B}^*(I - P_{\mathcal{Q}})\mathcal{B}$, to construct the $\mathcal{CQ}$ algorithm for approximating a solution to problem (1), where $P_{\mathcal{C}} : \mathcal{H}_1 \to \mathcal{C}$ and $P_{\mathcal{Q}} : \mathcal{H}_2 \to \mathcal{Q}$ are the metric (orthogonal) projection operators, I is the identity operator in $\mathcal{H}_1$, $\mathcal{B}^*$ is the adjoint of $\mathcal{B}$, $L = ||\mathcal{B}||^2$ and τ is a positive constant. For any initial point $u_0 \in \mathcal{H}_1$, it iteratively generates a sequence $\{u_n\}$ by

$$u_{n+1} = P_{\mathcal{C}}\big(u_n - \tau\nabla g(u_n)\big), \quad \forall n \geq 0, \tag{3}$$

where $\tau \in \left(0, \frac{2}{||\mathcal{B}||^2}\right)$ is the step length. However, in many practical applications, there are two major difficulties associated with the implementation of Algorithm 3. The first is its requirement in each iteration to computes two projections $P_{\mathcal{C}}$ and $P_{\mathcal{Q}}$, which depends heavily on the geometry of the sets $\mathcal{C}$ and $\mathcal{Q}$. These operations are known to be extremely expensive and sometimes not even possible for a wide range of practical problems. The second is in the choice of the step length τ, which requires the calculation of $||\mathcal{B}||$ in each iteration. This also appears very difficult to obtain in many practices. However, several efforts are devoted to improving the Algorithm 3 from both the theoretical and numerical implementation perspectives. Specifically, some researchers focused on the need to address the two aforementioned setbacks. For instance, Yang [8], applied the sub-level sets $\hat{\hat{\mathcal{C}}}$ and $\hat{\hat{\mathcal{Q}}}$ defined by

$$\hat{\hat{\mathcal{C}}} = \{u \in \mathcal{H}_1 : c(u) \leq 0\} \quad \text{and} \quad \hat{\hat{\mathcal{Q}}} = \{t \in \mathcal{H}_2 : q(t) \leq 0\}, \tag{4}$$

➤ **Received:** 03.12.2024 ➤ **Revised:** 21.04.2025 ➤ **Accepted:** 12.05.2025 ➤ **Published:** 30.06.2025

where $c : \mathcal{H}_1 \to \mathbb{R}$ and $q : \mathcal{H}_2 \to \mathbb{R}$ are weakly lower semicontinuous and convex functions and two half-spaces at a point u_n defined by

$$\mathcal{C}_n = \{u \in \mathcal{H}_1 : c(u_n) \leq \langle \phi_n, u_n - u \rangle\} \quad \textit{and} \quad \mathcal{Q}_n = \{t \in \mathcal{H}_2 : q(\mathcal{B}u_n) \leq \langle \varphi_n, \mathcal{B}u_n - t \rangle\}, \tag{5}$$

where $\phi_n \in \partial c(u_n)$, $\varphi_n \in \partial q(\mathcal{B}u_n)$, $\hat{\mathcal{C}} \subseteq \mathcal{C}_n$ and $\hat{\mathcal{Q}} \subseteq \mathcal{Q}_n$ for every $n \geq 0$, to develop the relaxed algorithm. It is iteratively defined for any initial point $u_0 \in \mathcal{H}_1$ by

$$u_{n+1} = P_{\mathcal{C}_n}(u_n - \tau \nabla g_n(u_n)), \quad \forall n \geq 0, \tag{6}$$

where $\tau \in \left(0, \frac{2}{\|\mathcal{B}\|^2}\right)$ and $\nabla g_n = \mathcal{B}^*(I - P_{\mathcal{Q}_n})\mathcal{B}$. Observe that Algorithm 6 is easier to implement than Algorithm 3, since the projections $P_{\mathcal{C}_n}$ and $P_{\mathcal{Q}_n}$ can easily be computed using their known closed-form expressions (see, [9], Example 29.20). However, it still requires calculating $\|\mathcal{B}\|$ in each iteration. Several methods that do not require the calculations of $\|\mathcal{B}\|$ have been suggested. One of such methods is the following weakly convergent relaxed algorithm of Qu and Xiu [10], which is a modified version of Algorithm 6 in Euclidean spaces.

Algorithm 1 Qu and Xiu's Algorithm 4.1 [10]

Initialization: Take $\gamma > 0$, $\mu \in (0,1)$ and $\varepsilon \in (0,1)$. Select arbitrary $u_1 \in \mathbb{R}^m$ and set $n = 1$.

Step 1. Compute

$$h_n = (u_n - \tau_n \nabla g_n(u_n)),$$

where $\tau_n = \gamma^{\mu^{t_n}}$ and t_n is the least positive integer such that

$$\|\nabla g_n(u_n) - \nabla g_n(h_n)\| \leq \frac{\varepsilon}{\tau_n}\|u_n - h_n\|.$$

Step 2. Compute $u_{n+1} = P_{\mathcal{C}_n}(u_n - \tau_n \nabla g_n(h_n))$, set $n := n + 1$ and go back to Step 1.

Observe that the authors of Algorithm 1 and others in [11, 12], have adopted the Armijo-like step-length procedure. However, it has been observed in several instances that finding a suitable step length in each iteration using an Armijo-like step-length technique requires multiple search procedures, which may lead the algorithmic performance to be ineffective. Dong et al. [13], recently introduced a self-adaptive relaxed version of Algorithm 6, which suggests to generate the sequence of step lengths $\{\tau_n\}$ by the following monotonic step length criterion:

$$\tau_{n+1} = \begin{cases} \min\left\{\frac{\varepsilon\|u_n - h_n\|}{\|\nabla g_n(u_n) - \nabla g_n(h_n)\|}, \ \tau_n\right\}, & \text{if } \nabla g_n(u_n) \neq \nabla g_n(h_n), \\ \tau_n, & \text{otherwise}, \end{cases} \tag{7}$$

with $\tau_1 > 0$, $\varepsilon \in (0,1)$. Very recently, Tan et al. [5] considered the following quasi-monotonic step length criterion to update the step length τ_{n+1} in each iteration:

$$\tau_{n+1} = \begin{cases} \min\left\{\frac{\varepsilon\|u_n - h_n\|}{\|\nabla g_n(u_n) - \nabla g_n(h_n)\|}, \ \tau_n\alpha_n + \omega_n\right\}, & \text{if } \nabla g_n(u_n) \neq \nabla g_n(h_n), \\ \tau_n\alpha_n + \omega_n, & \text{otherwise}, \end{cases}$$

where $\tau_1 > 0$, $\varepsilon \in (0,1)$, $\alpha_n \in [1, \infty)$ such that $\sum_{n=1}^{\infty}(\alpha_n - 1) < \infty$ and $\omega_n \in [0, \infty)$ such that

$\sum_{n=1}^{\infty}\varpi_n < \infty$. For recent developments concerning splitting problems and various iterative techniques, see [6, 11, 12, 14–23]. It is observed that only weak convergence properties have been established for the algorithms presented in [3–5, 7, 8, 10–13, 17, 19–21]. However, it is well known that strong convergence is more desirable, especially in infinite-dimensional spaces. In this regard, several authors [6, 16, 18], have introduced iterative schemes that guarantee strong convergence for problem (1). For instance, Ma and Liu [6] proposed the following Halpern-type relaxed algorithm.

Algorithm 2 Ma and Liu's Algorithm 1 [6]

Initialization: Take $\tau_1 > 0$, $\delta \in]0, b] \subset]0, 1]$, $\{\gamma_n\} \subset [0, \gamma) \subset [0, 1)$, $\beta_n \in [1, \infty)$ such that $\sum_{n=1}^{\infty}(\beta_n - 1) < \infty$, $\varpi_n \in [0, \infty)$ such that $\sum_{n=1}^{\infty}\varpi_n < \infty$, $\{\alpha_n\}$ and $\{\varrho_n\}$ such that ([6], (A2)) holds. Select $u_0,\ u_1 \in H_1$ and a random vector x. Set $n = 1$.

Step 1. Compute

$$w_n = u_n + \gamma_n(u_n - u_{n-1}) \text{ and } u_{n+1} = \alpha_n x + (1-\alpha_n)P_{C_n}(w_n - \tau_n\nabla g_n(w_n)),$$

where

$$\gamma_n = \begin{cases} \min\left\{\frac{\varrho_n}{||u_n - u_{n-1}||},\ \gamma\right\}, & \text{if } u_n \neq u_{n-1}, \\ \gamma, & \text{otherwise.} \end{cases}$$

Step 2. Update the step length τ_{n+1} by

$$\tau_{n+1} = \begin{cases} \min\left\{\frac{2\varepsilon g_n(w_n)}{||\nabla g_n(w_n)||^2},\ \tau_n\beta_n + \varpi_n\right\}, & \text{if } \nabla g_n(w_n) \neq 0, \\ \tau_n\beta_n + \varpi_n, & \text{otherwise,} \end{cases}$$

set $n := n+1$ and go back to Step 1.

Nowadays, many researchers focus on developing methods with fast convergence properties, as these are often required for solving various problems arising in real-world applications [24, 25]. In this spirit, several numerical algorithms have recently been constructed based on certain acceleration processes, popularly known as methods with extrapolation procedures. The common extrapolation procedure studied by several researchers is the Polyak's inertial technique [26]. For any given points u_n and u_{n-1}, the Polyak's inertial extrapolation step is given by

$$w_n = u_n + \lambda(u_n - u_{n-1}), \quad \forall\lambda > 0. \tag{8}$$

The term (8) is popularly known as the one-step inertial term, and it has recently become very attractive, being applied by several authors [3, 12, 16, 19–23, 27–31]. However, it has been observed in several studies that some methods incorporating the step (8) appear to be slower than their counterparts that do not include it, see [32, 33] and the references therein. Ortega and Rheinboldt [34] proposed the following general iterative procedure:

$$u_{n+1} = f_n(u_n, u_{n-1}, \ldots, u_{n-k+1}),\ \forall n \geq 1, \tag{9}$$

with $k \geq 1$ as an integer and $\forall n \geq 1$, f_n is a function, whose task is to perform the extrapolation onto the points $u_n, u_{n-1}, \ldots, u_{n-k+1}$. The iterative procedure (9) is known as the k-step method. It is noted that the inertial step (8) corresponds to a special case of the procedure (9) with $k = 2$.

Consequently, Polyak [35] improved the convergence property of step (8) by suggesting the idea of applying the procedure (9) with $k > 2$, which is referred to as a multi-step inertial method. Since then, some authors [6, 36, 37] adopted the idea of multi-step inertial method to improve the speed of convergence of their schemes. Additionally, to improve the speed of algorithms with the inertial step (8), Dong et al. [38] proposed unifying two steps of (8) into a single iterative method, which is termed a general inertial method. By integrating the idea of the multi-step inertial method and that of the general inertial method, Dong et al. [39] proposed the general fixed point iterative method based on the Krasnosel'skíí-Mann algorithm for a nonexpansive mapping in real Hilbert space with two steps of the procedure (9) and $k \in K_n \subseteq \{0, 1, 2, \cdots, n-1\}$, $\forall n \geq 1$. Their iterative algorithm is defined for any points $u_0,\ u_1 \in \mathcal{H}$ by

$$\begin{cases} w_n = u_n + \sum_{k\in K_n} \gamma_{n,k}(u_{n-k} - u_{n-k-1}), \\ v_n = u_n + \sum_{k\in K_n} \delta_{n,k}(u_{n-k} - u_{n-k-1}), \\ u_{n+1} = (1-\alpha_n)w_n + \alpha_n T v_n, \quad \forall n \geq 1, \end{cases}$$

where $\gamma_{n,k},\ \delta_{n,k} \in (-1, 2]^{|K_n|}$ for each $k \in K_n$ and $|K_n|$ denotes the cardinality of the set K_n. They proved its weak convergence based on the convergence of the Krasnosel'skíí-Mann algorithm with perturbations to a fixed point of a nonexpansive mapping T in real Hilbert space and numerically demonstrated that it is faster than some inertial methods in solving certain problems. In [40], Malitsky introduced the golden ratio-based algorithm (GRAAL) for variational inequalities in Euclidean spaces, which iteratively generates the sequence $\{u_n\}$ for any initial points $u_1,\ v_0 \in \mathbb{R}^m$ by

$$v_n = \frac{\phi - 1}{\phi} u_n + \frac{1}{\phi} v_{n-1}, \tag{10}$$

$$u_{n+1} = \text{Prox}_{\lambda f}(v_n - \lambda F(u_n)),\ \ \forall n \geq 1,$$

where $\phi = \frac{\sqrt{5}+1}{2}$ is the golden ratio parameter. However, we observe from (10) that GRAAL provides a new extrapolation technique based on the golden ratio parameter. Very recently, Zhang and Chu [41], modified the step (10) and introduced an alternating extrapolation step based on the golden ratio parameter. For any points u_n and v_{n-1}, their alternating extrapolation step is defined by

$$v_n = \begin{cases} u_n, & \text{if } n \text{ is even}, \\ \frac{\phi-1}{\phi} u_n + \frac{1}{\phi} v_{n-1}, & \text{if } n \text{ is odd}. \end{cases} \tag{11}$$

Additionally, from (2) and its gradient, it is observe that the aforementioned methods for problem (1), such as those in [3–8, 10–13, 16–21, 23], are hybrid steepest-types with directions $d_n = -\nabla g_n(u_n)$ at a point u_n. However, as noted from [42], their performance may be improved when considered with the following conjugate gradient-like direction (12) or the three-term conjugate gradient-like direction (13) [43, 44].

$$d_n = -\nabla g_n(u_n) + \varsigma_n^{(1)} d_{n-1}, \tag{12}$$

and

$$d_n = -\nabla g_n(u_n) + \varsigma_n^{(1)} d_{n-1} - \varsigma_n^{(2)} x_n, \tag{13}$$

where, for each $i = 1, 2$, $\varsigma_n^{(i)} \in [0, \infty)$ and $\{x_n\} \subseteq \mathcal{H}_1$ is a sequence of an arbitrary points. As numerically shown in [42–44], provided that, for each $i = 1, 2$, $\lim_{n\to\infty} \varsigma_n^{(i)} = 0$ and $\{x_n\}$ is bounded, the method with the direction (13) is faster than its variant with the direction (12). In this spirit, several authors improved their iterative methods based on the direction (12) or (13), [45–50]. Recently, motivated by the self-adaptive relaxed algorithm [51], one-step inertial term (8) and the conjugate gradient-like direction (12), Che et al. [52], proposed the accelerated relaxed algorithm for problem (1). Although the algorithm proposed in [52] with the conjugate gradient-like direction (12) and inertial term (8) has recorded better performance when its numerical results on signal and image recovery problems are compared with some methods in the literature, but, its convergence results are found to heavily rely on a condition presented in [52], Lemma 5, (iii). This condition seems overly restrictive, and it would be highly beneficial to consider waiving or relaxing it.

Motivated and inspired by the results in [5, 13, 34, 40–42, 52], We first construct two algorithms incorporating new extrapolation techniques and perturbations, called the beyond the golden ratio algorithm with perturbations (BGRAP) and the beyond the alternating golden ratio algorithm with perturbations (BAGRAP). The new extrapolation steps in BGRAP and BAGRAP, form the generalizations of the golden ratio and the alternating golden ratio extrapolation steps (10) and (11), respectively, by expanding the selection region of the parameter ϕ, which is no longer limited to the golden ratio $\frac{\sqrt{5}+1}{2}$. Each of BGRAP and BAGRAP uses the monotonic adaptive step length rule (7) to generate a sequence of step lengths. This allows them to work very effectively without requiring a calculation of $\|\mathcal{B}\|$ or a line search procedure in each iteration, which improves their convergence properties and implementations. Under some simple and weaker assumptions, we prove the strong convergence of BGRAP and BAGRAP to a minimum-norm solution of problem (1) in infinite-dimensional real Hilbert spaces. In addition to the new extrapolation steps in BGRAP and BAGRAP, we construct two extensions of each. The first of each is a general multi-step inertial-type algorithm, denoted by BGRGMiA and BAGRGMiA, respectively. To the best of our knowledge, these are the first algorithms proposed in the literature to incorporate two steps of the improved version of the classical inertial term (8) (i.e., the procedure (9) with $k > 2$), along with new extrapolation steps based on (10) and (11), respectively. These techniques help to improve the convergence speed of their sequences toward the desired solution. The second of each is a multi-step inertial and three-term conjugate gradient-like algorithm, represented by BGRMiTTCG and BAGRMiTTCG, respectively. To the best of our knowledge, these are also the first algorithms of their kind to employ both the three-term conjugate gradient-like direction (13), one step of the procedure (9) with $k > 2$, and the new extrapolation steps based on (10) and (11), respectively. These procedures effectively accelerate the sequences generated by these algorithms toward a solution of the problem. Moreover, we analyze their possible applications in classification problems for an interesting real-world dataset based on the extreme learning machine (ELM) with the $\ell_1 - \ell_2$ hybrid regularization approach and in solving constrained minimization problems in infinite-dimensional Hilbert spaces. In all the experiments, the proposed algorithms, which generalize and improve several algorithms in the literature, such as those in [5, 10, 11, 13, 20–22, 52], comparatively demonstrate superior performance in achieving significantly better results than some related algorithms.

The rest of this work is organized as follows. In Section 2, we consider the definitions of some basic concepts and existing results. In Section 3, we introduce the first algorithm with perturbations, called the beyond the golden ratio algorithm with perturbations (BGRAP), establish its strong convergence to a minimum-norm solution of problem (1) and present two of its extensions, namely, beyond the golden ratio and general multi-step inertial algorithm (BGRGMiA) and beyond the golden ratio and multi-step inertial algorithm with three-term conjugate gradient-like direction (BGRMiTTCG). In Section 4, we introduce the second algorithm with perturbations, namely, beyond the alternating golden ratio algorithm with perturbations (BAGRAP), establish its strong convergence to a minimum-norm solution of problem (1) and similarly present two of its extensions: namely, beyond the alternating golden ratio general multi-step inertial algorithm (BAGRGMiA) and beyond the alternating golden ratio multi-step inertial and three-term conjugate gradient-like algorithm (BAGRMiTTCG). In Section 5, we provide some numerical illustrations of the proposed algorithms with some related algorithms in constrained minimization and classification problems. In Section 6, we conclude the paper by summarizing its general findings and presenting several directions for further research. Finally, for more convenience, in Section 7, we provide a table of notations Table 6, which summarizes the main notations used throughout the paper.

2 Preliminaries

Throughout this work, we use $u_n \rightharpoonup u$ (resp., $u_n \to u$) to denote the weak (resp., strong) convergence of a sequence $\{u_n\}$ to u. Let $\mathcal{H}$ be a real Hilbert space. For all $u,\ v \in \mathcal{H}$ and $\alpha \in [0,1]$, we use the following:

$$\|u+v\|^2 = \|u\|^2 + \|v\|^2 + 2\langle u, v\rangle, \tag{14}$$

and

$$\|\alpha u + (1-\alpha)v\|^2 = \alpha\|u\|^2 + (1-\alpha)\|v\|^2 - \alpha(1-\alpha)\|u-v\|^2. \tag{15}$$

Definition 1 *[9]* Let $\mathcal{T} : \mathcal{H} \to \mathcal{H}$ be an operator. Then $\mathcal{T}$ is called

• L - Lipschitz continuous with $L > 0$, if

$$\|\mathcal{T}u - \mathcal{T}v\| \le L\|u-v\|, \quad \forall u,\ v \in \mathcal{H}. \tag{16}$$

• Nonexpansive, if (16) holds with $L = 1$.

• Firmly nonexpansive, if

$$\|\mathcal{T}u - \mathcal{T}v\| \le \langle u-v, \mathcal{T}u - \mathcal{T}v\rangle, \quad \forall u,\ v \in \mathcal{H}. \tag{17}$$

Recall that $P_\mathcal{C} : \mathcal{H} \to \mathcal{C}$ is known as the metric (orthogonal) projection operator and for any $u \in \mathcal{H}$, an element $P_\mathcal{C}u \in \mathcal{C}$ exists, such that

$$\|u - P_\mathcal{C}u\| \le \|u-v\|, \quad \forall v \in \mathcal{C}.$$

Moreover, for all $u \in \mathcal{H}$ and $v \in \mathcal{C}$, the following properties hold for an element $P_{\mathcal{C}}u$, [53].

$$\langle u - P_{\mathcal{C}}u, v - P_{\mathcal{C}}u \rangle \leq 0, \tag{18}$$

which is equivalent to

$$\|u - P_{\mathcal{C}}u\|^2 + \|v - P_{\mathcal{C}}u\|^2 \leq \|u - v\|^2. \tag{19}$$

Remark 1 It is commonly known that $I - P_{\mathcal{C}}$ satisfies the inequality (17), [54].

Definition 2 [9] Let $f : \mathcal{H} \to (-\infty, +\infty]$ be a convex and proper function. Then
(1) f is said to be (weakly) lower semi-continuous (w-lsc) if for any sequence $u_n \in \mathcal{H}$ such that ($u_n \rightharpoonup u^*$) $u_n \to u^*$ as $n \to \infty$, we have

$$\liminf_{n\to\infty} f(u_n) \geq f(u^*).$$

(2) $\partial f(u)$ is known as the subdifferential of f at a point u, which is defined by

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

An element $y \in \partial f(u)$ is called a subgradient of f at u.

Lemma 1 *[54, 55] Let $\tau > 0$ and $u^* \in \mathcal{H}_1$, then, the following statements are equivalent.*

- *u^* solves problem (1);*
- *u^* solves the fixed point problem $u^* = P_{\mathcal{C}}(u^* - \tau\nabla g(u^*))$.*

Lemma 2 [56] *Let $\{u_n\}$ be a sequence of nonnegative real numbers, such that $\forall n \geq 1$,*

$$u_{n+1} \leq (1 - \beta_n)u_n + \beta_n\eta_n \text{ and } u_{n+1} \leq u_n - \chi_n + \Phi_n,$$

where $\beta_n \in (0, 1)$, $\chi_n \in [0, +\infty)$ and η_n, $\Phi_n \in (-\infty, +\infty)$ such that
(B1) $\lim_{n\to\infty} \beta_n = 0$ *and* $\sum_{n=1}^{\infty} \beta_n = \infty$;
(B2) $\lim_{n\to\infty} \Phi_n = 0$;
(B3) $\lim_{j\to\infty} \chi_{n_j} = 0$ *implies that* $\limsup_{r\to\infty} \eta_{n_j} \leq 0$ *for any subsequence* $\{n_j\}$ *of* $\{n\}$,
then $\lim_{n\to\infty} u_n = 0$.

3 Beyond the golden ratio algorithm with perturbations

In this part, we first introduce the following algorithm with perturbations and a new extrapolation technique, which generalizes the golden ratio extrapolation step (10). We analyze its strong convergence to a minimum-norm solution of problem (1) in real Hilbert spaces. To construct this algorithm, we define $\hat{\hat{\mathcal{C}}}$, $\hat{\hat{\mathcal{Q}}}$, $\mathcal{C}_n$, $\mathcal{Q}_n$, g_n and ∇g_n as in (4), (5) and (2), respectively. For its convergence analysis, we make the following assumptions:

Assumption 1 $(A1)$ The solutions' set of problem (1) is denoted by $\Omega \neq \varnothing$.

$(A2)$ $c : \mathcal{H}_1 \to \mathbb{R}$ and $q : \mathcal{H}_2 \to \mathbb{R}$ are respectively convex, subdifferentiable and weakly lower semicontinuous functions on $\mathcal{H}_1$ and $\mathcal{H}_2$.
$(A3)$ For any $u \in \mathcal{H}_1$ and $t \in \mathcal{H}_2$, there exists at least one subgradient $\phi \in \partial c(u)$ and $\varphi \in \partial q(t)$, and the subdifferential operators ∂c and ∂q are bounded on bounded sets.
$(A4)$ Let $\tau_1 > 0$, $\varepsilon > 0$, $\rho \in (0, \frac{1}{\varepsilon})$, $\xi \in [0,1)$ and $\{\beta_n\} \subset (0,1)$ such that $\lim_{n\to\infty} \beta_n = 0$ and $\sum_{n=0}^{\infty} \beta_n = \infty$.

Algorithm 3 Beyond the Golden Ratio Algorithm with Perturbations (BGRAP)

Initialization: Take τ_1, ε, ρ, ξ and $\{\beta_n\}$ such that Assumption 1, $(A4)$ holds. Choose $u_1, v_0 \in \mathcal{H}_1$ and set $n = 1$.

Step 1. Compute
$$v_n = (1-\xi)u_n + \xi v_{n-1}. \tag{20}$$

Step 2. Compute
$$h_n = P_{C_n}(v_n - \rho\tau_n \nabla g_n(v_n) + e_1(v_n)).$$

Step 3. Compute
$$u_{n+1} = (1-\beta_n)P_{C_n}(v_n - \rho\tau_n \nabla g_n(h_n) + e_2(v_n)),$$

update the step length τ_{n+1} by
$$\tau_{n+1} = \begin{cases} \min\left\{ \frac{\varepsilon\|v_n - h_n\|}{\|\nabla g_n(v_n) - \nabla g_n(h_n)\|}, \tau_n \right\}, & \text{if } \nabla g_n(v_n) \neq \nabla g_n(h_n), \\ \tau_n, & \text{otherwise}, \end{cases} \tag{21}$$

set $n := n+1$ and go back to Step 1.

To establish the convergence of Algorithm 3, we provide the following additional assumption:

Assumption 2 Take $\nu \geq 1$, $\delta_n := \frac{\beta_n(1-\xi)}{\nu}$, $\forall n \geq 1$ and assume that for each $i = 1, 2$, the sequence of perturbations $\{e_i(v_n)\}$ satisfies $\|e_2(v_n)\| < \frac{\delta_n}{1-\delta_n}$, $\forall n \geq 1$ and $\lim_{n\to\infty} \frac{\|e_i(v_n)\|}{\delta_n} = 0$.

Remark 2
- It is observed that if we take $\xi = \frac{1}{\phi}$, for $\phi \in (1, +\infty)$, the extrapolation step (20) corresponds to the golden ratio-based extrapolation step (10), with the parameter ϕ not only limited to the golden ratio $\frac{\sqrt{5}+1}{2}$. Thus, Algorithm 3 (i.e., BGRAP) includes a golden ratio-based algorithm with perturbations (GRAP) when $\xi = \frac{1}{\phi}$ and $\phi = \frac{\sqrt{5}+1}{2}$ as a particular case. It is also noticed that if $\xi = 0$, Algorithm 3 reduces to a strong convergence relaxed algorithm with perturbations, which generalizes and improves several algorithms in [5, 10, 11, 13, 20–22, 52] and the references therein.
- It also appears from Algorithm 3 that

$$u_{n+1} = (1-\beta_n)\big(P_{C_n}(v_n - \rho\tau_n \nabla g_n(h_n)) + \hat{e}_2(v_n)\big), \tag{22}$$

and

$$\|\hat{e}_2(v_n)\| = \|P_{C_n}(v_n - \rho\tau_n \nabla g_n(h_n) + e_2(v_n)) - P_{C_n}(v_n - \rho\tau_n \nabla g_n(h_n))\| \leq \|e_2(v_n)\|. \tag{23}$$

So, from (23), we find that the sequence $\{\hat{e}_2(v_n)\}$ satisfies the stated conditions of Assumption 2.

- It is immediately seen from Assumption 2 that $\forall n \geq 1$, $\delta_n \in (0,1)$, $\lim_{n\to\infty} \delta_n = 0$, $\sum_{n=1}^{\infty} \delta_n = \infty$ and

$$(1-\beta_n) + \frac{\xi}{1-\xi} \leq (1-\delta_n)\frac{1}{1-\xi}. \tag{24}$$

- It is also noticed that since $\frac{\delta_n - 1}{1-\delta_n} < 0 \leq ||\hat{e}_2(v_n)||, \ \forall n \geq 1,$ then

$$\delta_n(1 + ||\hat{e}_2(v_n)||) - ||\hat{e}_2(v_n)|| < 1, \ \forall n \geq 1, \tag{25}$$

  and since from Assumption 2 and (23), we have $||\hat{e}_2(v_n)|| < \frac{\delta_n}{1-\delta_n}, \ \forall n \geq 1,$ then

$$\delta_n(1 + ||\hat{e}_2(v_n)||) - ||\hat{e}_2(v_n)|| > 0, \ \forall n \geq 1. \tag{26}$$

  Thus, taking $\Gamma_n = \delta_n(1 + ||\hat{e}_2(v_n)||) - ||\hat{e}_2(v_n)||, \ \forall n \geq 1$, it follows from (25) and (26) that $\Gamma_n \in (0,1), \ \forall n \geq 1$. In view of (24), we find that for any $\forall n \geq 1$,

$$(1 + ||\hat{e}_2(v_n)||)\Big((1-\beta_n) + \frac{\xi}{1-\xi}\Big) \leq (1-\delta_n)(1 + ||\hat{e}_2(v_n)||)\frac{1}{1-\xi} = \Big(1-\Gamma_n\Big)\frac{1}{1-\xi}. \tag{27}$$

We first prove the following lemma.

Lemma 3 *Suppose that $\{\tau_n\}$ is the sequence of step lengths generated by (21). Then it is well defined and $\tau_n \geq \frac{\varepsilon}{||\mathcal{B}||^2}$ for all $n \geq 1$.*

Proof By the Lipschitz contuinity of ∇g_n with constant $||\mathcal{B}||^2$, we obtain

$$\frac{\varepsilon||v_n - h_n||}{||\nabla g_n(v_n) - \nabla g_n(h_n)||} \geq \frac{\varepsilon||v_n - h_n||}{||\mathcal{B}||^2||v_n - h_n||} = \frac{\varepsilon}{||\mathcal{B}||^2},$$

together with (21), we see that $\tau_{n+1} \geq \min\{\tau_n, \frac{\varepsilon}{||\mathcal{B}||^2}\}$. By induction, one finds that $\tau_n \geq \min\{\tau_1, \frac{\varepsilon}{||\mathcal{B}||^2}\}$. It is also seen from (21) that $\tau_{n+1} \leq \tau_n, \ \forall n \geq 1$. By the monotonicity and the existence of the lower bound of the sequence $\{\tau_n\}$, we obtain that $\lim_{n\to\infty} \tau_n$ exists and since $\min\{\tau_1, \frac{\varepsilon}{||\mathcal{B}||^2}\}$ is a lower bound of the sequence $\{\tau_n\}$, then, we can find $\tau > 0$ such that $\lim_{n\to\infty} \tau_n = \tau$.
Next, we prove that the sequence $\{u_n\}$ of Algorithm 3 is bounded.

Lemma 4 *Suppose that Assumptions 1, (A4) and 2 hold. Let $\{v_n\}$ be a sequence generated by Algorithm 3. Then, for any point $z \in \Omega$, the sequence $\{||v_n - z||\}$ is bounded.*

Proof Let $z \in \Omega$. Then $\mathcal{B}z \in \mathcal{Q}_n$ and consequently, $\nabla g_n(z) = \mathcal{B}^*(I - P_{\mathcal{Q}_n})\mathcal{B}z = 0$. Therefore, combining with the fact that $I - P_{\mathcal{Q}_n}$ satisfies (17), we have

$$\begin{aligned} \langle \nabla g_n(h_n), h_n - z\rangle &= \langle (I - P_{\mathcal{Q}_n})\mathcal{B}h_n - (I - P_{\mathcal{Q}_n})\mathcal{B}z, \mathcal{B}h_n - \mathcal{B}z\rangle \\ &\geq ||(I - P_{\mathcal{Q}_n})\mathcal{B}h_n||^2 \\ &= 2g_n(h_n). \end{aligned} \tag{28}$$

Taking $m_n = P_{\mathcal{C}_n}(v_n - \rho\tau_n\nabla g_n(h_n))$, it follows from inequalities (19) and (28), and the convexity of $||.||^2$ that

$$\begin{aligned} ||m_n - z||^2 &\leq ||v_n - \rho\tau_n \nabla g_n(h_n) - z||^2 - ||v_n - \rho\tau_n \nabla g_n(h_n) - m_n||^2 \\ &= ||v_n - z||^2 - ||v_n - m_n||^2 - 2\rho\tau_n \langle \nabla g_n(h_n), v_n - z \rangle + 2\rho\tau_n \langle \nabla g_n(h_n), v_n - m_n \rangle \\ &\leq ||v_n - z||^2 - ||v_n - m_n||^2 - 4\rho\tau_n g_n(h_n) - 2\rho\tau_n \langle \nabla g_n(h_n), m_n - h_n \rangle . \end{aligned} \tag{29}$$

Now, we estimate the rightmost term in (29) as follows. Observe that

$$||v_n - h_n||^2 + ||h_n - m_n||^2 - ||v_n - m_n||^2 = 2 \langle v_n - h_n, m_n - h_n \rangle . \tag{30}$$

Since $m_n \in \mathcal{C}_n$, then (18), (21) and the mean value inequality yield

$$\begin{aligned} 2 \langle v_n - h_n, m_n - h_n \rangle &= 2 \langle v_n - \rho\tau_n \nabla g_n(v_n) + e_1(v_n) - h_n, m_n - h_n \rangle - 2 \langle e_1(v_n), m_n - h_n \rangle \\ &\quad + 2\rho\tau_n \langle \nabla g_n(v_n) - \nabla g_n(h_n), m_n - h_n \rangle + 2\rho\tau_n \langle \nabla g_n(h_n), m_n - h_n \rangle \\ &\leq 2\rho\tau_n ||\nabla g_n(v_n) - \nabla g_n(h_n)|| ||m_n - h_n|| + 2||e_1(v_n)|| ||m_n - h_n|| \\ &\quad + 2\rho\tau_n \langle \nabla g_n(h_n), m_n - h_n \rangle \\ &\leq \left(\frac{\varepsilon\rho\tau_n}{\tau_{n+1}} + ||e_1(v_n)|| \right) (||v_n - h_n||^2 + ||m_n - h_n||^2) \\ &\quad + ||e_1(v_n)|| + 2\rho\tau_n \langle \nabla g_n(h_n), m_n - h_n \rangle . \end{aligned} \tag{31}$$

From (30) and (31), we deduce

$$\begin{aligned} 2\rho\tau_n \langle \nabla g_n(h_n), m_n - h_n \rangle &\geq \left(1 - \left(\frac{\varepsilon\rho\tau_n}{\tau_{n+1}} + ||e_1(v_n)|| \right) \right) (||v_n - h_n||^2 + ||m_n - h_n||^2) \\ &\quad - ||e_1(v_n)|| - ||v_n - m_n||^2. \end{aligned} \tag{32}$$

In view of the inequalities (29) and (32), one sees that

$$||m_n - z||^2 \leq ||v_n - z||^2 - \frac{4\rho\varepsilon}{||\mathcal{B}||^2} g_n(h_n) + ||e_1(v_n)|| - \rho_n (||v_n - h_n||^2 + ||m_n - h_n||^2), \tag{33}$$

where

$$\rho_n = \left(1 - \left(\frac{\varepsilon\rho\tau_n}{\tau_{n+1}} + ||e_1(v_n)|| \right) \right). \tag{34}$$

Note that for $\varepsilon > 0$ and $\rho \in (0, \frac{1}{\varepsilon})$, we immediately see from Lemma 3, Assumption 2 and Eq. (34) that there exists $\rho^* > 0$ such that $\lim_{n\to\infty} \rho_n = \rho^*$, where

$$\rho^* = (1 - \varepsilon\rho). \tag{35}$$

Thus, we can find a positive number W, such that $\rho_n > 0, \ \forall n \geq W$. Combining (22) and (33), we see that

$$\begin{aligned} ||u_{n+1}-z||^2 &= ||(1-\beta_n)(m_n+\hat{e}_2(v_n))-z||^2 \\ &\leq \beta_n||z||^2+(1-\beta_n)(1+||\hat{e}_2(v_n)||)||m_n-z||^2+(1+||\hat{e}_2(v_n)||)||\hat{e}_2(v_n)|| \\ &\leq (1-\beta_n)(1+||\hat{e}_2(v_n)||)||v_n-z||^2+(1+||\hat{e}_2(v_n)||)\big(||e_1(v_n)||+||\hat{e}_2(v_n)||\big) \\ &\quad +\beta_n||z||^2-\rho_n(1-\beta_n)(1+||\hat{e}_2(v_n)||)\big(||v_n-h_n||^2+||m_n-h_n||^2\big) \\ &\quad -4(1-\beta_n)\frac{\rho\varepsilon(1+||\hat{e}_2(v_n)||)}{||\mathcal{B}||^2}g_n(h_n). \end{aligned} \tag{36}$$

In view of the identity (15) and the fact that $u_{n+1}=\frac{1}{1-\xi}v_{n+1}-\frac{\xi}{1-\xi}v_n$, one sees that

$$||u_{n+1}-z||^2 = \frac{1}{1-\xi}||v_{n+1}-z||^2-\frac{\xi}{1-\xi}||v_n-z||^2+\frac{\xi}{(1-\xi)^2}||v_{n+1}-v_n||^2. \tag{37}$$

Combining (27), (34), (35), (36), (37) and Assumptions 1, $(A4)$ and 2, we obtain

$$\begin{aligned} \frac{1}{1-\xi}||v_{n+1}-z||^2 &\leq (1+||\hat{e}_2(v_n)||)\big((1-\beta_n)+\frac{\xi}{1-\xi}\big)||v_n-z||^2-\frac{\xi}{(1-\xi)^2}||u_{n+1}-v_n||^2 \\ &\quad +\Psi_n+\beta_n||z||^2-\rho_n(1-\beta_n)(1+||\hat{e}_2(v_n)||)\big(||v_n-h_n||^2+||m_n-h_n||^2\big) \\ &\quad -4(1-\beta_n)\frac{\rho\varepsilon(1+||\hat{e}_2(v_n)||)}{||\mathcal{B}||^2}g_n(h_n) \qquad (38) \\ &\leq (1-\Gamma_n)\frac{1}{1-\xi}||v_n-z||^2+\Gamma_n\Big(\frac{\beta_n}{\Gamma_n}||z||^2+\frac{\Psi_n}{\Gamma_n}\Big), \forall n\geq \mathcal{W}, \qquad (39) \end{aligned}$$

where $\Psi_n=(1+||\hat{e}_2(v_n)||)\big(||e_1(v_n)||+||\hat{e}_2(v_n)||\big)$. From Assumptions 1, $(A4)$ and 2, we see that for any $\nu\geq 1$, $\frac{\beta_n}{\delta_n}=\frac{\nu}{1-\xi}$. So, taking $M_1, M_2>0$, for which $\frac{\beta_n}{\Gamma_n}=\frac{\nu}{(1-\xi)(1+||\hat{e}_2(v_n)||)}\times\frac{1}{1-\frac{||\hat{e}_2(v_n)||}{\delta_n(1+||\hat{e}_2(v_n)||)}}\leq M_1$ and $\frac{(1+||\hat{e}_2(v_n)||)}{\Gamma_n}\big(||e_1(v_n)||+||\hat{e}_2(v_n)||\big)\leq M_2$. Then, by representing $M:=M_1||z||^2+M_2$ and the fact that $\Gamma_n\in(0,1)$, we obtain from (39) that

$$\begin{aligned} \frac{1}{1-\xi}||v_{n+1}-z||^2 &\leq (1-\Gamma_n)\frac{1}{1-\xi}||v_n-z||^2+\Gamma_n M \\ &\leq \max\{\frac{1}{1-\xi}||v_n-z||^2, M\} \\ &\ \ \vdots \\ &\leq \max\{\frac{1}{1-\xi}||v_0-z||^2, M\}, \forall n\geq W, \end{aligned}$$

thus, $\{\frac{1}{1-\xi}||v_n-z||^2\}$ is bounded. Consequently, the sequences $\{v_n\}$, $\{h_n\}$ and $\{u_n\}$ are bounded. Now, we state and prove the following as our first main convergence theorem.

Theorem 1 *Let $\{u_n\}$ be a sequence produced by Algorithm 3 such that the conditions of Assumptions 1 and 2 hold. Then, the sequence $\{u_n\}$ converges strongly to a minimum-norm solution of problem* (1) *(i.e., a point $z^*=P_\Omega 0$).*

Proof Let $z\in\Omega$. Using identity (14) and inequality (33), we find from (22) that

$$
\begin{aligned}
\|u_{n+1}-z\|^2 &= \|(1-\beta_n)(m_n+\hat{e}_2(v_n))-z\|^2 \\
&\le (1-\beta_n)^2(1+\|\hat{e}_2(v_n)\|)\|m_n-z\|^2+(1+\|\hat{e}_2(v_n)\|)\|\hat{e}_2(v_n)\| \\
&\quad +\beta_n^2\|z\|^2+2\beta_n(1-\beta_n)\langle m_n-z,-z\rangle+\Xi_n \\
&\le (1-\beta_n)(1+\|\hat{e}_2(v_n)\|)\|v_n-z\|^2+\Psi_n+\beta_n^2\|z\|^2+2\beta_n(1-\beta_n)\langle m_n-z,-z\rangle+\Xi_n, \quad (40)
\end{aligned}
$$

where $\Xi_n = 2\beta_n(1-\beta_n)\|\hat{e}_2(v_n)\|\|z\|$. Combining (27), (37) and (40), one sees that

$$
\begin{aligned}
\frac{1}{1-\xi}\|v_{n+1}-z\|^2 &\le (1+\|\hat{e}_2(v_n)\|)\left((1-\beta_n)+\frac{\xi}{1-\xi}\right)\|v_n-z\|^2+2\beta_n(1-\beta_n)\langle m_n-z,-z\rangle \\
&\quad +\Psi_n+\Xi_n+\beta_n^2\|z\|^2 \\
&\le (1-\delta_n)\frac{1}{1-\xi}\|v_n-z\|^2+\Xi_n+2\beta_n(1-\beta_n)\langle m_n-z,-z\rangle \\
&\quad +\Psi_n+\frac{1}{1-\xi}(1-\delta_n)\|\hat{e}_2(v_n)\|\|v_n-z\|^2+\beta_n^2\|z\|^2. \quad (41)
\end{aligned}
$$

Now, without loss of generality, we see from Assumptions 1, $(A4)$ and 2 that there exist $a, b > 0$, such that $\forall\ n \ge 1$,

$$4(1-\beta_n)\frac{\rho\varepsilon(1+\|\hat{e}_2(v_n)\|)}{\|\mathcal{B}\|^2} \ge a, \quad \text{and} \quad \rho_n(1-\beta_n)(1+\|\hat{e}_2(v_n)\|) \ge b.$$

So that from (38) and (41), one finds that

$$\frac{1}{1-\xi}\|v_{n+1}-z\|^2 \le \frac{1}{1-\xi}\|v_n-z\|^2-\xi_n+\Phi_n,$$

and

$$\frac{1}{1-\xi}\|v_{n+1}-z\|^2 \le (1-\delta_n)\frac{1}{1-\xi}\|v_n-z\|^2+\delta_n\eta_n,$$

where

$$\xi_n = ag_n(h_n)+b\left(\|v_n-h_n\|^2+\|m_n-h_n\|^2\right)+\frac{\xi}{(1-\xi)^2}\|u_{n+1}-v_n\|^2,\ \Phi_n=\Psi_n+\beta_n\|z\|^2,$$

and

$$\eta_n = \frac{1}{\delta_n}\left(\Psi_n+\Xi_n+(1-\delta_n)\frac{1}{1-\xi}\|\hat{e}_2(v_n)\|\|v_n-z\|^2+\beta_n^2\|z\|^2+2\beta_n(1-\beta_n)\langle m_n-z,-z\rangle\right).$$

By Assumptions 1, $(A4)$ and 2, we find that $\lim_{n\to\infty}\Phi_n = 0$. Therefore, to apply Lemma 2, it suffices to show that for any subsequence $\{\xi_{n_j}\}$ of $\{\xi_n\}$, the following holds:

$$\lim_{j\to\infty}\xi_{n_j} = 0 \Rightarrow \limsup_{j\to\infty}\eta_{n_j} \le 0.$$

Now, suppose that $\{\xi_{n_j}\}$ is a subsequence of $\{\xi_n\}$ such that $\lim_{j\to\infty}\xi_{n_j} = 0$, then, in view of (34), Assumptions 1, $(A4)$ and 2, and the fact that $\lim_{j\to\infty}\rho_{n_j} = \rho^* > 0$, we see that

$$\lim_{j\to\infty}\|v_{n_j} - h_{n_j}\| = 0, \quad \lim_{j\to\infty}\|m_{n_j} - h_{n_j}\| = 0, \lim_{j\to\infty}\|u_{n_j+1} - v_{n_j}\| = 0, \text{ and}$$

$$\lim_{j\to\infty} g_{n_j}(h_{n_j}) = 0 \Leftrightarrow \lim_{j\to\infty}\|(I - P_{\mathcal{Q}_{n_j}})\mathcal{B}h_{n_j}\|^2 = 0. \tag{42}$$

Since the sequence $\{u_n\}$ is bounded, then there exists a subsequence $\{u_{n_j}\}$ of $\{u_n\}$ converging weakly to a point say u^*. Observe that Assumption 1, $(A3)$ ensures the existence of a constant $\breve{\varrho} > 0$ such that $\|\varphi_{n_j}\| \le \breve{\varrho}$, thus, since $P_{\mathcal{Q}_{n_j}}\mathcal{B}h_{n_j} \in \mathcal{Q}_{n_j}$, we find from the definition of $\mathcal{Q}_{n_j}$ and (42) that

$$\begin{aligned} q(\mathcal{B}h_{n_j}) &\le \left\langle \varphi_{n_j}, \mathcal{B}h_{n_j} - P_{\mathcal{Q}_{n_j}}\mathcal{B}h_{n_j} \right\rangle \\ &\le \|\varphi_{n_j}\|\|\mathcal{B}h_{n_j} - P_{\mathcal{Q}_{n_j}}\mathcal{B}h_{n_j}\| \\ &\le \breve{\varrho}\|(I - P_{\mathcal{Q}_{n_j}})\mathcal{B}h_{n_j}\|^2 \to 0 \; as \; j \to \infty. \end{aligned} \tag{43}$$

We also see from the weak lower semicontuinity of q and inequality (43) that

$$q(\mathcal{B}u^*) \le \liminf_{r\to\infty} q(\mathcal{B}h_{n_j}) \le 0, \text{ which implies } \mathcal{B}u^* \in \hat{\hat{\mathcal{Q}}}. \tag{44}$$

Similarly, the boundedness of ∂c on bounded sets guarantees the existence of $\breve{\sigma} > 0$, such that $\|\phi_{n_j}\| \le \sigma$. Since $u_{n_j+1} \in \mathcal{C}_{n_j}$, we obtain from the definition of $\mathcal{C}_{n_j}$ and (42) that

$$\begin{aligned} c(h_{n_j}) &\le \left\langle \phi_{n_j}, h_{n_j} - u_{n_j+1} \right\rangle \\ &\le \|\phi_{n_j}\|\|h_{n_j} - u_{n_j+1}\| \\ &\le \breve{\sigma}\|h_{n_j} - u_{n_j+1}\| \to 0 \; as \; j \to \infty. \end{aligned} \tag{45}$$

Similar arguments used in obtaining (44) lead to find that $c(u^*) \le 0$, showing that $u^* \in \hat{\hat{\mathcal{C}}}$. Then, we conclude that $u^* \in \Omega$. Since the choice of u^* was arbitrarily, it follows that $\omega_w(u_n) \subset \Omega$. By (42) and the metric projection property (18), one finds that

$$\limsup_{j\to\infty}\left\langle m_{n_j} - z, u - z \right\rangle = \max_{u^*\in\omega_w(u_n)} \langle u^* - z, u - z\rangle \le 0. \tag{46}$$

From Assumptions 1, $(A4)$ and 2, the boundedness of $\{\|v_n - z\|\}$ and (46), we observe that $\limsup_{j\to\infty} \eta_{n_j} \le 0$. Therefore, it follows from Lemma 2 that $\lim_{n\to\infty} \frac{1}{1-\xi}\|v_n - z^*\| = 0$. Thus, $v_n \to z^* = P_\Omega 0$ as $n \to \infty$. Combining with the definition of v_n in Algorithm 3, one deduces that

$$\|u_n - v_n\| = \frac{\xi}{1-\xi}\|v_n - v_{n-1}\| \le \frac{\xi}{1-\xi}\Big(\|v_n - z^*\| + \|v_{n-1} - z^*\|\Big) \to 0 \text{ as } n \to \infty. \tag{47}$$

By (47) and the fact that $v_n \to z^* = P_\Omega 0$ as $n \to \infty$, we finally obtain $u_n \to z^* = P_\Omega 0$ as $n \to \infty$.

Hence, this completes the proof. To obtain some extensions of Algorithm 3, we make the following assumption:

Assumption 3 Let $k \in S_n \subseteq \{0, 1, 2, \ldots, n-1\}$, v_{n-k} and v_{n-k-1} be arbitrary points in $\mathcal{H}_1$, $\forall n \geq 1$. Select $\delta_n = \frac{\beta_n(1-\xi)}{\nu}$ for $\nu \geq 1$, $\varsigma_{n,k} \in [0, +\infty)$ and $\sigma_{n,k} \in \left[0, \frac{\delta_n}{1-\delta_n}\right)$ such that $\lim_{n\to\infty} \frac{\sum_{k\in S_n} \varsigma_{n,k}}{\delta_n} = 0$ and $\lim_{n\to\infty} \frac{\sum_{k\in S_n} \sigma_{n,k}}{\delta_n} = 0$. Choose $\vartheta_{n,k} \in [0, \bar{\vartheta}_{n,k}]$, $\zeta_{n,k} \in [0, \bar{\zeta}_{n,k}]$, $\forall n \geq 1$ and any $\eta_1, \eta_2 > 0$ such that

$$\bar{\vartheta}_{n,k} := \begin{cases} \min\left\{\frac{\varsigma_{n,k}}{\|v_{n-k}-v_{n-k-1}\|}, \eta_1\right\}, & \textit{if } v_{n-k} \neq v_{n-k-1}, \\ \eta_1, & \text{otherwise}, \end{cases} \tag{48}$$

and

$$\bar{\zeta}_{n,k} := \begin{cases} \min\left\{\frac{\sigma_{n,k}}{\|v_{n-k}-v_{n-k-1}\|}, \eta_2\right\}, & \textit{if } v_{n-k} \neq v_{n-k-1}, \\ \eta_2, & \text{otherwise}. \end{cases} \tag{49}$$

Remark 3 We can easily see from Assumption 3 that for every $n \geq 1$ and $k \in S_n$, we have

$$\vartheta_{n,k}\|v_{n-k} - v_{n-k-1}\| \leq \varsigma_{n,k}, \quad \text{and} \quad \zeta_{n,k}\|v_{n-k} - v_{n-k-1}\| \leq \sigma_{n,k},$$

then, by the fact that for every $k \in S_n$, $\lim_{n\to\infty} \frac{\sum_{k\in S_n} \varsigma_{n,k}}{\delta_n} = 0$ and $\lim_{n\to\infty} \frac{\sum_{k\in S_n} \sigma_{n,k}}{\delta_n} = 0$, (48) and (49) yield

$$\lim_{n\to\infty} \frac{\sum_{k\in S_n} \vartheta_{n,k}\|v_{n-k} - v_{n-k-1}\|}{\delta_n} = 0, \text{ and } \lim_{n\to\infty} \frac{\sum_{k\in S_n} \zeta_{n,k}\|v_{n-k} - v_{n-k-1}\|}{\delta_n} = 0.$$

It is therefore easy to observe that for every $n \geq 1$, taking

$$e_1(v_n) = \sum_{k\in S_n} \vartheta_{n,k}(v_{n-k} - v_{n-k-1}), \tag{50}$$

and

$$e_2(v_n) = \sum_{k\in S_n} \zeta_{n,k}(v_{n-k} - v_{n-k-1}), \tag{51}$$

then, Algorithm 3 (i.e., BGRAP) becomes the following algorithm and its strong convergence to a minimum-norm solution of problem (1) follows from that of Theorem 1 with Assumption 2 replaced by Assumption 3.

Remark 4 Based on Algorithm 4 (i.e., BGRGMiA), we make the following remarks.

(i) In view of Remark 2, (i), we similarly see that Algorithm 4 (i.e., BGRGMiA) includes a general multi-step inertial golden ratio-based algorithm with perturbations (GMiGRA) when $\xi = \frac{1}{\phi}$ and $\phi = \frac{\sqrt{5}+1}{2}$ as a particular case.

(ii) If $S_n = \{0\}$, $\gamma_{n,0} = \gamma_n$ and $\zeta_{n,0} = \zeta_n$, $\forall n \geq 1$, BGRGMiA becomes a beyond the golden ratio and general inertial algorithm (BGRGiA), which particularly includes a general inertial golden ratio-based algorithm GiGRA, when $\xi = \frac{1}{\phi}$ and $\phi = \frac{\sqrt{5}+1}{2}$.

Algorithm 4 Beyond the Golden Ratio and General Multi-Step Inertial Algorithm (BGRGMiA)

Initialization: Take $\tau_1,\ \varepsilon,\ \rho,\ \xi$ and $\{\beta_n\}$ such that Assumption 1, $(A4)$ holds. Select $S_n,\ \nu,\ \{\delta_n\},\ \{\vartheta_{n,k}\}$ and $\{\zeta_{n,k}\},\ \forall k \in S_n$ as described in Assumption 3. Choose $u_1,\ v_0 \in \mathcal{H}_1$ and set $n = 1$.

Step 1. Compute v_n by (20) and $w_n = v_n + \sum_{k\in S_n} \gamma_{n,k}(v_{n-k} - v_{n-k-1})$.
Step 2. Compute $h_n = P_{C_n}(w_n - \rho\tau_n \nabla g_n(v_n))$.
Step 3. Compute

$$y_n = v_n + \sum_{k\in S_n} \zeta_{n,k}(v_{n-k} - v_{n-k-1}),$$

$$u_{n+1} = (1-\beta_n)P_{C_n}(y_n - \rho\tau_n \nabla g_n(h_n)),$$

update the step length τ_{n+1} by (21), set $n := n+1$ and go back to Step 1.

(iii) If $\zeta_{n,k} = 0,\ \forall n \geq 1$ and $k \in S_n$, BGRGMiA reduces to a beyond the golden ratio and multi-step inertial algorithm (BGRMiA), which involves a multi-step inertial golden ratio-based algorithm MiGRA, when $\xi = \frac{1}{\phi}$ and $\phi = \frac{\sqrt{5}+1}{2}$ as a particular case.
(iv) If $\xi = 0$, BGRGMiA reduces to a general multi-step inertial algorithm GMiA with two steps of the procedure (9) when $k > 2$, which further generalizes and improves several methods in the literature (see, [5, 10, 11, 13, 20–22, 52]) and the references therein.

We also construct the following three-term conjugate gradient-like algorithm from Algorithm 3.

Algorithm 5 Beyond the Golden Ratio and Multi-Step Inertial Algorithm with Three-Term Conjugate Gradient-Like Direction (BGRMiATTCG)

Initialization: Take $\tau_1,\ \varepsilon,\ \rho,\ \xi$ and $\{\beta_n\}$ such that the Assumption 1, $(A4)$ holds. Select $S_n,\ \nu,\ \{\delta_n\}$ and $\{\zeta_{n,k}\},\ \forall k \in S_n$ as described in Assumption 3, $\lambda_n,\ \varsigma_n^{(2)} \in [0,+\infty)$ such that $\lim\limits_{n\to\infty} \frac{\lambda_n}{\delta_n} = 0,\ \lim\limits_{n\to\infty} \frac{\varsigma_n^{(2)}}{\delta_n} = 0$, and a bounded sequence $\{x_n\} \subset \mathcal{H}_1$. Choose $u_1,\ v_0 \in \mathcal{H}_1$ and $d_1 = -\nabla g_0(v_0)$. Set $n = 1$.

Step 1. Compute v_n by (20).
Step 2. Compute $d_{n+1} = -\rho\tau_n \nabla g_n(v_n)/\lambda + \varsigma_n^{(1)} d_n - \varsigma_n^{(2)} x_n$ and $h_n = P_{C_n}(v_n + \lambda d_{n+1})$, where

$$\varsigma_n^{(1)} = \frac{\lambda_n}{\max\{\|d_n\|, \sigma\}}. \tag{52}$$

Step 3. Compute $y_n = v_n + \sum_{k\in S_n} \zeta_{n,k}(v_{n-k} - v_{n-k-1})$,

$$u_{n+1} = (1-\beta_n)P_{C_n}(y_n - \rho\tau_n \nabla g_n(h_n)),$$

update the step length τ_{n+1} by (21), set $n := n+1$ and go back to Step 1.

Remark 5 Observe that taking $e_2(v_n)$ as in (51) and defining $e_1(v_n)$ as follows

$$e_1(v_n) = \lambda(\varsigma_n^{(1)} d_n - \varsigma_n^{(2)} x_n),$$

with $\varsigma_n^{(1)}$ defined in (52), then, from the conditions on $\lambda_n,\ \varsigma_n^{(2)}$ and the boundedness of the sequence $\{x_n\}$, Algorithm 3 becomes Algorithm 5. So that the strong convergence of Algorithm 5 to a minimum-norm solution of problem (1) follows from that of Theorem 1.

Remark 6 For Algorithm 5 (i.e., BGRMiATTCG), we provide the following remarks.

(i) Similarly, from Remark 2, (i), we find that Algorithm 5 (i.e., BGRMiATTCG) particularly includes a golden ratio and multi-step inertial algorithm with three-term conjugate gradient-like direction (GRMiATTCG) when $\xi = \frac{1}{\phi}$, and $\phi = \frac{\sqrt{5}+1}{2}$.

(ii) If $S_n = \{0\}$, and $\zeta_{n,0} = \zeta_n$, $\forall n \geq 1$, the BGRMiATTCG becomes a beyond the golden ratio inertial algorithm with three-term conjugate gradient-like direction (BGRiATTCG), which further particularly includes a three-term conjugate gradient-like and inertial golden ratio-based algorithm TTCiGRA, when $\xi = \frac{1}{\phi}$, and $\phi = \frac{\sqrt{5}+1}{2}$.

(iii) If $\zeta_{n,k} = 0$, $\forall n \geq 1$ and $k \in S_n$, the BGRMiATTCG reduces to a beyond the golden ratio algorithm with three-term conjugate gradient-like direction (BGRATTCG), which similarly includes a three-term conjugate gradient-like and golden ratio-based algorithm TTCGRA, when $\xi = \frac{1}{\phi}$, and $\phi = \frac{\sqrt{5}+1}{2}$ as a particular case.

(iv) If $\xi = 0$, BGRMiATTCG becomes a three-term conjugate gradient-like and multi-step inertial algorithm TTCGMiA, which further generalizes and improves several methods in the literature (see., [5, 10, 11, 13, 20–22, 52]) and the references therein.

4 Beyond the alternating golden ratio algorithm with perturbations

In this part, we first introduce the following algorithm with perturbations and a new extrapolation technique, which generalizes the alternating golden ratio extrapolation step (11), and analyze its strong convergence to a minimum-norm solution of problem (1) in real Hilbert spaces.

Algorithm 6 Beyond the Alternating Golden Ratio Algorithm with Perturbations (i.e., BAGRAP)

Initialization: Take τ_1, ε, ρ, ξ and $\{\beta_n\}$ such that Assumption 1, $(A4)$ holds. Choose u_1, $v_0 \in \mathcal{H}_1$ and set $n = 1$.

Step 1. Compute

$$v_n = \begin{cases} u_n, & \text{if } n \text{ is even} \\ (1-\xi)u_n + \xi v_{n-1}, & \text{if } n \text{ is odd.} \end{cases} \tag{53}$$

Step 2. Compute $h_n = P_{C_n}(v_n - \rho\tau_n \nabla g_n(v_n) + e_1(v_n))$.

Step 3. Compute

$$u_{n+1} = (1-\beta_n)P_{C_n}(v_n - \rho\tau_n \nabla g_n(h_n) + e_2(v_n)),$$

update the step size τ_{n+1} by (21), set $n := n+1$ and go back to Step 1.

Assumption 4 Take $\mu_n = \frac{3\beta_n(1-\xi)}{4}, \forall n \geq 1$ and for each $i = 1, 2$, assume that the sequence of perturbations $\{e_i(v_n)\}$ in Algorithm 6 satisfies

$$\|e_2(v_n)\| < \frac{\mu_n}{3(1-\mu_n)}, \ \forall n \geq 1, \text{ and } \lim_{n\to\infty} \frac{\|e_i(v_n)\|}{\mu_n} = 0.$$

Remark 7 (i) It is observed that if $\xi = \frac{1}{\phi}$, for $\phi \in (1, +\infty)$, the extrapolation step (53) corresponds to the alternating golden ratio-based extrapolation step (11), with the parameter ϕ not only limited to the golden ratio $\frac{\sqrt{5}+1}{2}$. Thus, for $\phi = \frac{\sqrt{5}+1}{2}$, an alternating golden ratio-based algorithm with perturbations (AGRAP), which is also first mentioned in this work, becomes a particular case of Algorithm 6 (i.e., ABGRAP). It is also noticed that if $\xi = 0$, Algorithm 6 reduces to a strong convergence relaxed $\mathcal{CQ}$ algorithm with perturbations, which further generalizes and improves several algorithms in [5, 10, 11, 13, 20–22, 52] and the references therein.

(ii) It is similarly easy to see from Assumption 4 that $\forall n \geq 1$, $\mu_n \in (0, \frac{3}{4})$, $\lim_{n\to\infty} \mu_n = 0$ and

$$(1-\beta_{2n})(1-\xi)+\xi \leq (1-\mu_n). \tag{54}$$

(iii) Equivalently, since $\frac{\mu_n - 1}{3(1-\mu_n)} < 0 \leq \|\hat{e}_2(v_n)\|,\ \forall n \geq 1,$ then

$$\mu_n - 3(1-\mu_n)\|\hat{e}_2(v_n)\| < 1,\ \forall n \geq 1, \tag{55}$$

and since from Assumption 4 and (23), we have $\|\hat{e}_2(v_n)\| < \frac{\mu_n}{3(1-\mu_n)},\ \forall n \geq 1,$ then

$$\mu_n - 3(1-\mu_n)\|\hat{e}_2(v_n)\| > 0,\ \forall n \geq 1. \tag{56}$$

In a similar fashion, taking $\Gamma_n^* = \mu_n - 3(1-\mu_n)\|\hat{e}_2(v_n)\|$, $\forall n \geq 1$, it follows from (55) and (56) that $\Gamma_n^* \in (0,1),\ \forall n \geq 1$. In view of the fact that $\mu_n \leq \frac{3}{4}$, $\forall n \geq 1$, one sees from Assumption 4 that $\|\hat{e}_2(v_n)\| < 1,\ \forall n \geq 1$. Thus, we obtain from (54) that

$$\begin{aligned}&\left(1+\|\hat{e}_2(v_n)\|\right)\left(1+\|\hat{e}_2(v_{n+1})\|\right)\Big((1-\beta_n)(1-\xi)+\xi\Big)\\&\quad\leq (1-\mu_n)\left(1+\|\hat{e}_2(v_n)\|\right)\left(1+\|\hat{e}_2(v_{n+1})\|\right) < \left(1-\Gamma_n^*\right),\ \forall n \geq 1.\end{aligned} \tag{57}$$

Now, to establish the convergence of a sequence $\{u_n\}$ generated by Algorithm 6, we first show that an even subsequence $\{u_{2n}\}$ of $\{u_n\}$ is bounded.

Lemma 5 *Let $\{u_{2n}\}$ be an even subsequence of $\{u_n\}$ generated by Algorithm 6 and suppose that the Assumptions 1, $(A4)$ and 4 hold. Then, for any point $z \in \Omega$, the even subsequence $\{\|u_{2n} - z\|\}$ of $\{\|u_n - z\|\}$ is bounded.*

Proof Applying the estimates in the proof of Lemma 3 and Lemma 4, we deduce from inequality (36) and (53) that

$$\|v_{2n+1}-z\|^2 = (1-\xi)\|u_{2n+1}-z\|^2 + \xi\|u_{2n}-z\|^2 - \frac{\xi}{(1-\xi)^2}\|u_{2n+1}-u_{2n}\|^2, \tag{58}$$

and

$$\|u_{2n+1}-z\|^2 \quad\leq\quad (1-\beta_{2n})(1+\|\hat{e}_2(v_{2n})\|)\|u_{2n}-z\|^2 + \Psi_{2n} + \beta_{2n}\|z\|^2 - \Upsilon_{2n}, \tag{59}$$

where
$\Upsilon_{2n} = \rho_{2n}(1-\beta_{2n})(1+\|\hat{e}_2(v_{2n})\|)\left(\|u_{2n}-h_{2n}\|^2+\|p_{2n}-h_{2n}\|^2\right)+4(1-\beta_{2n})\frac{\rho\varepsilon\left(1+\|\hat{e}_2(v_{2n})\|\right)}{\|\mathcal{B}\|^2}g_{2n}(h_{2n}).$
One also sees from (36), (57), (58), (59) and similar arguments in deriving (39) that

$$\begin{aligned}\|u_{2n+2}-z\|^2 &\leq (1+\|\hat{e}_2(v_{2n+1})\|)\|v_{2n+1}-z\|^2 + \Psi_{2n+1} + \beta_{2n+1}\|z\|^2 - \Upsilon_{2n+1}\\&\leq (1+\|\hat{e}_2(v_{2n+1})\|)(1+\|\hat{e}_2(v_{2n})\|)\Big((1-\beta_{2n})(1-\xi)+\xi\Big)\|u_{2n}-z\|^2\\&\quad + (1+\|\hat{e}_2(v_{2n+1})\|)(1-\xi)\Big(\Psi_{2n}+\beta_{2n}\|z\|^2-\Upsilon_{2n}\Big) + \Psi_{2n+1}+\beta_{2n+1}\|z\|^2\\&\quad - \frac{\xi}{(1-\xi)^2}\|u_{2n+1}-u_{2n}\|^2 - \Upsilon_{2n+1}\end{aligned}$$

$$\leq \quad (1-\Gamma^*_{2n})||u_{2n}-z||^2 + (1+||\hat{e}_2(v_{2n+1})||)(1-\xi)\Big(\Psi_{2n}+2\beta_{2n}||z||^2-\Upsilon_{2n}\Big)$$
$$+\Psi_{2n+1}-\Upsilon_{2n+1}-\frac{\xi}{(1-\xi)^2}||u_{2n+1}-u_{2n}||^2 \tag{60}$$
$$\leq \quad (1-\Gamma^*_{2n})||u_{2n}-z||^2 + 2\Gamma^*_{2n}\frac{\theta_{2n}}{\Gamma^*_{2n}}(\Psi_{2n}+\beta_{2n}||z||^2), \forall n \geq W, \tag{61}$$

where $\theta_{2n} = (1+||\hat{e}_2(v_{2n+1})||)$. Similarly, from Assumptions 1, $(A4)$ and 4, one finds that for any $\frac{\beta_n}{\mu_n} = \frac{4}{3(1-\xi)}$, $\forall n \geq 1$. So that taking M^*_1, $M^*_2 > 0$, for which $\frac{\theta_{2n}\beta_{2n}}{\Gamma^*_{2n}} = \frac{4\theta_{2n}}{3(1-\xi)\left(1-\frac{3(1-\mu_n)||e_2(v_{2n})||}{\mu_{2n}}\right)} \leq M^*_1$ and $\frac{\theta_{2n}\Psi_{2n}}{\Gamma^*_{2n}} \leq M^*_2$. Then, by denoting $M^* = 2(M^*_1||z||^2 + M^*_2)$, the fact that $\Gamma^*_{2n} \in (0,1)$ and (61) yield

$$\begin{aligned}
||u_{2n+2}-z||^2 &\leq (1-\Gamma^*_{2n})||u_{2n}-z||^2 + \Gamma^*_{2n}M^* \\
&\leq \max\left\{||u_{2n}-z||^2, M^*\right\} \\
&\vdots \\
&\leq \max\left\{||u_0-z||^2, M^*\right\}, \forall n \geq W,
\end{aligned}$$

thus, an even subsequence $\{||u_{2n}-z||^2\}$ is bounded. Consequently, the even subsequences $\{u_{2n}\}$, $\{v_{2n}\}$ and $\{h_{2n}\}$ are bounded.
Next, we formulate and prove the following convergence theorem for Algorithm 6.

Theorem 2 *Let $\{u_n\}$ be a sequence produced by Algorithm 6 such that the conditions of Assumptions 1 and 4 hold. Then, the sequence $\{u_n\}$ converges strongly to a minimum-norm solution of problem (1) (i.e., a point $z^* = P_\Omega 0$).*

Proof Let $z \in \Omega$. Then, we find from (40), (58) and the assignment of v_{2n} in (53) that

$$||u_{2n+1}-z||^2 \leq (1-\beta_{2n})(1+||\hat{e}_2(v_{2n})||)||u_{2n}-z||^2 + \Psi_{2n}$$
$$+\beta^2_{2n}||z||^2 + \Xi_{2n} + 2\beta_{2n}(1-\beta_{2n})\langle p_{2n}-z, -z\rangle. \tag{62}$$

Similarly, one deduces from (40), (54), (58) and (62) that

$$\begin{aligned}
||u_{2n+2}-z||^2 \leq\ & (1+||\hat{e}_2(v_{2n+1})||)||v_{2n+1}-z||^2 + \Xi_{2n+1} + \Psi_{2n+1} + \beta^2_{2n+1}||z||^2 \\
& + 2\beta_{2n+1}(1-\beta_{2n+1})\langle p_{2n+1}-z, -z\rangle \\
\leq\ & (1+||\hat{e}_2(v_{2n+1})||)(1+||\hat{e}_2(v_{2n})||)\Big((1-\beta_{2n})(1-\xi)+\xi\Big)||u_{2n}-z||^2 \\
& + \Xi_{2n+1} + \Psi_{2n+1} + (1+||\hat{e}_2(v_{2n+1})||)(\Psi_{2n}+\Xi_{2n}+2\beta^2_{2n}||z||^2) \\
& + 2(1+||\hat{e}_2(v_{2n+1})||)\beta_{2n}(1-\beta_{2n})\langle p_{2n}-z, -z\rangle \\
& + 2\beta_{2n+1}(1-\beta_{2n+1})\langle p_{2n+1}-z, -z\rangle \\
\leq\ & (1-\mu_{2n})||u_{2n}-z||^2 + (1+||\hat{e}_2(v_{2n+1})||)(\Psi_{2n}+\Xi_{2n}+2\beta^2_{2n}||z||^2) \\
& + ((1+||\hat{e}_2(v_{2n+1})||)||\hat{e}_2(v_{2n})|| + ||\hat{e}_2(v_{2n+1})||)(1-\mu_{2n})||u_{2n}-z||^2 \\
& + \Xi_{2n+1} + \Psi_{2n+1} + 2(1+||\hat{e}_2(v_{2n+1})||)\beta_{2n}(1-\beta_{2n})\langle p_{2n}-z, -z\rangle \\
& + 2\beta_{2n+1}(1-\beta_{2n+1})\langle p_{2n+1}-z, -z\rangle.
\end{aligned} \tag{63}$$

Thus, without loss of generality, we similarly obtain from Assumptions 1, $(A4)$ and 4, and the fact that $\lim_{n\to\infty}\rho_n = \rho^* > 0$ that there exist $r, s, t > 0$, such that $\forall\ n \geq 1$,

$$4(1-\beta_n)\frac{\rho\varepsilon(1+\|\hat{e}_2(v_n)\|)}{\|\mathcal{B}\|^2} \geq r,\ \rho_n(1-\beta_n)(1+\|\hat{e}_2(v_n)\|) \geq s,\ \text{and}\ \theta_{2n} \geq t.$$

Therefore, (60) and (63) yield

$$\|u_{2n+2}-z\|^2 \leq \|u_{2n}-z\|^2 - \hat{\xi}_{2n} + \hat{\Phi}_{2n},$$

and

$$\|u_{2n+2}-z\|^2 \leq (1-\mu_{2n})\|u_{2n}-z\|^2 + \mu_{2n}\hat{\eta}_{2n},$$

where, for $\Lambda_n = \|u_n - h_n\|^2 + \|m_n - h_n\|^2$, $\quad \hat{\Phi}_{2n} = 2\theta_{2n}\Big(\Psi_{2n} + \beta_{2n}\|z\|^2\Big)$,
$\hat{\xi}_{2n} = r\Bigg(t(1-\xi)g_{2n}(h_{2n}) + g_{2n+1}(h_{2n+1})\Bigg) + s\Bigg(t(1-\xi)\Lambda_{2n} + \Lambda_{2n+1}\Bigg) + \frac{\xi}{(1-\xi)^2}\|u_{2n+1}-u_{2n}\|^2$,

$$\text{and}\ \hat{\eta}_{2n} = \frac{1}{\mu_{2n}}\Bigg((1+\|\hat{e}_2(v_{2n+1})\|)\left(\Psi_{2n} + \Xi_{2n} + 2\beta_{2n}^2\|z\|^2\right) + \big((1+\|\hat{e}_2(v_{2n+1})\|)\|\hat{e}_2(v_{2n})\|$$

$$+\|\hat{e}_2(v_{2n+1})\|\big)(1-\mu_{2n})\|u_{2n}-z\|^2 + \Xi_{2n+1} + \Psi_{2n+1} + 2(1+\|\hat{e}_2(v_{2n+1})\|)\beta_{2n}(1-\beta_{2n})\langle p_{2n}-z, -z\rangle +$$
$$2\beta_{2n+1}(1-\beta_{2n+1})\langle p_{2n+1}-z, -z\rangle\Bigg).$$

We equivalently find from Assumptions 1, $(A4)$ and 4 that $\lim_{n\to\infty}\hat{\Phi}_{2n} = 0$. Therefore, to apply Lemma 2, it also suffices to show that for any subsequence $\{\hat{\xi}_{2n_j}\}$ of $\{\hat{\xi}_{2n}\}$, the following holds: $\lim_{j\to\infty}\xi_{2n_j} = 0 \Rightarrow \limsup_{j\to\infty}\ \hat{\eta}_{2n_j} \leq 0$. Now, suppose that $\{\hat{\xi}_{2n_j}\}$ is a subsequence of $\{\hat{\xi}_{2n}\}$ such that $\lim_{j\to\infty}\hat{\xi}_{2n_j} = 0$, then, it follows from (34), Assumptions 1, $(A4)$ and 4 that

$$\lim_{j\to\infty}\|u_{2n_j} - h_{2n_j}\| = 0,\ \lim_{j\to\infty}\|p_{2n_j} - h_{2n_j}\| = 0,\ \lim_{j\to\infty}\|u_{2n_j+1} - u_{2n_j}\| = 0,\ \lim_{j\to\infty}\|u_{2n_j+1} - h_{2n_j+1}\| = 0,$$

$$\lim_{j\to\infty}\|p_{2n_j+1} - h_{2n_j+1}\| = 0,\ \lim_{j\to\infty}g_{2n_j}(h_{2n_j}) = 0 \Leftrightarrow \lim_{j\to\infty}\|(I - P_{Q_{2n_j}})\mathcal{B}h_{2n_j}\|^2 = 0,\ \text{and}$$

$$\lim_{j\to\infty}g_{2n_j+1}(h_{2n_j+1}) = 0 \Leftrightarrow \lim_{j\to\infty}\|(I - P_{Q_{2n_j+1}})\mathcal{B}h_{2n_j+1}\|^2 = 0. \tag{64}$$

Since $\{u_{2n}\}$ is bounded, then, using (64) and following the same proof lines of (43) and (45), one sees that $\omega_w(u_{2n}) \subset \Omega$. In view of (64) and the metric projection property in (18), one finds that

$$\limsup_{j\to\infty}\left\langle p_{2n_j} - z, u - z\right\rangle = \max_{u^*\in\omega_w(u_{2n})}\langle u^* - z, u - z\rangle \leq 0,\ \text{and}$$

$$\limsup_{j\to\infty}\left\langle p_{2n_j+1}-z,u-z\right\rangle=\max_{u^*\in\omega_w(u_{2n})}\langle u^*-z,u-z\rangle\leq 0. \quad (65)$$

By Assumptions 1, $(A4)$ and 4, we observe from (65) that $\limsup_{j\to\infty}\eta_{2n_j}\leq 0$. Therefore, it follows from Lemma 2 that $\lim_{n\to\infty}\|u_{2n}-z^*\|=0$. Thus, $u_{2n}\to z^*=P_\Omega 0$ as $n\to\infty$. Combining with (64), one easily obtains that $u_{2n+1}\to z^*=P_\Omega 0$ as $n\to\infty$. Therefore, the whole sequence $\{u_n\}$ generated by Algorithm 6 converges strongly to $z^*=P_\Omega 0$. Hence, this completes the proof.

Equivalently, to obtain the first extension of Algorithm 6, we make the following assumption.

Assumption 5 Let $k\in S_n\subseteq\{0,1,2,\ldots,n-1\}$, v_{n-k} and v_{n-k-1} be arbitrary points in $\mathcal{H}_1$, $\forall n\geq 1$. Select $\mu_n=\frac{3\beta_n(1-\xi)}{4}$, $\varsigma_{n,k}\in[0,+\infty)$ and $\sigma_{n,k}\in\left[0,\frac{\mu_n}{3(1-\mu_n)}\right)$ such that $\lim_{n\to\infty}\frac{\sum_{k\in S_n}\varsigma_{n,k}}{\mu_n}=0$ and $\lim_{n\to\infty}\frac{\sum_{k\in S_n}\sigma_{n,k}}{\mu_n}=0$. Choose $\{\vartheta_{n,k}\}$ and $\{\zeta_{n,k}\}$ as defined in Assumption 3.

Remark 8 It easy to see that taking the sequences $\{e_1(v_n)\}$ and $\{e_2(v_n)\}$ as in (50) and (51), respectively, Algorithm 6 becomes the following algorithm and its strong convergence to a minimum-norm solution of problem (1) follows from that of Theorem 2 with Assumption 4 replaced by Assumption 5.

Algorithm 7 Beyond the Alternating Golden Ratio and General Multi-Step Inertial Algorithm (BAGRGMiA)

Initialization: Take τ_1, ε, ρ, ξ and $\{\beta_n\}$ such that Assumption 1, $(A4)$ holds. Select S_n, $\{\mu_n\}$, $\{\vartheta_{n,k}\}$ and $\{\zeta_{n,k}\}$, $\forall k\in S_n$ as described in Assumption 5. Choose u_1, $v_0\in\mathcal{H}_1$ and set $n=1$.

Step 1. Compute v_n by (53) and $w_n=v_n+\sum_{k\in S_n}\vartheta_{n,k}(v_{n-k}-v_{n-k-1})$.
Step 2. Compute $h_n=P_{C_n}(w_n-\rho\tau_n\nabla g_n(v_n))$.
Step 3. Compute

$$y_n=v_n+\sum_{k\in S_n}\zeta_{n,k}(v_{n-k}-v_{n-k-1}),$$

$$u_{n+1}=(1-\beta_n)P_{C_n}(y_n-\rho\tau_n\nabla g_n(h_n)),$$

update the step size τ_{n+1} by (21), set $n:=n+1$ and go back to Step 1.

Remark 9 In a similar fashion, we make the following remarks based on Algorithm 7 (i.e., BAGRGMiA).

(i) Equivalently, one sees from Remark 7, (i) that Algorithm 7 (i.e., BAGRGMiA) particularly includes a general multi-step inertial and golden ratio-based algorithm (GMiAGRA) when $\xi=\frac{1}{2}$ and $\phi=\frac{\sqrt{5}+1}{2}$.

(ii) If $S_n=\{0\}$, $\gamma_{n,0}=\gamma_n$ and $\zeta_{n,0}=\zeta_n$, $\forall n\geq 1$, BAGRGMiA becomes a beyond the alternating golden ratio and general inertial algorithm (BAGRGiA), which particularly involves a general inertial and alternating golden ratio-based algorithm (GiAGRA) when $\xi=\frac{1}{\phi}$ and $\phi=\frac{\sqrt{5}+1}{2}$.

(iii) If $\zeta_{n,k}=0$, $\forall n\geq 1$ and $k\in S_n$, BAGRGMiA reduces to a beyond the alternating golden ratio and multi-step inertial algorithm (BAGRMiA), which similarly includes a multi-step inertial and alternating golden ratio-based algorithm (MiAGRA) when $\xi=\frac{1}{\phi}$ and $\phi=\frac{\sqrt{5}+1}{2}$ as a particular case.

(iv) If $\xi = 0$, BAGRGMiA reduces to the same algorithm provided in Remark 4, (iv).

We also construct the following three-term conjugate gradient-like algorithm as another extension of Algorithm 6 and its strong convergence to a minimum-norm solution of problem (1) follows from that of Theorem 2 with Assumption 4 replaced by Assumption 5 and Remark 5.

Algorithm 8 Beyond the Alternating Golden Ratio and Multi-Step Inertial Algorithm with Three-Term Conjugate Gradient-Like Direction (BAGRMiATTCG)

Initialization: Take τ_1, ε, ρ, ξ and $\{\beta_n\}$ such that Assumption 1, $(A4)$ holds. Select S_n, $\{\mu_n\}$ and $\{\zeta_{n,k}\}$, $\forall k \in S_n$ as described in Assumption 5, λ_n, $\varsigma_n^{(2)} \in [0, +\infty)$ such that $\lim_{n\to\infty} \frac{\lambda_n}{\mu_n} = 0$, $\lim_{n\to\infty} \frac{\varsigma_n^{(2)}}{\mu_n} = 0$, and a bounded sequence $\{x_n\} \subset \mathcal{H}_1$. Choose u_1, $v_0 \in \mathcal{H}_1$ and $d_1 = -\nabla g_0(v_0)$. Set $n = 1$.

Step 1. Compute v_n by (53).

Step 2. Compute $\varsigma_n^{(1)}$ by (52), $d_{n+1} = -\rho\tau_n\nabla g_n(v_n)/\lambda + \varsigma_n^{(1)} d_n - \varsigma_n^{(2)} x_n$, and

$$h_n = P_{\mathcal{C}_n}(v_n + \lambda d_{n+1}).$$

Step 3. Compute

$$y_n = v_n + \sum_{k\in S_n} \zeta_{n,k}(v_{n-k} - v_{n-k-1}),$$

$$u_{n+1} = (1-\beta_n)P_{\mathcal{C}_n}(y_n - \rho\tau_n\nabla g_n(h_n)),$$

update the step size τ_{n+1} by (21), set $n := n+1$ and go back to Step 1.

Remark 10 From Algorithm 8 (i.e., BAGRMiATTCG), we make the following remarks.

(i) Similarly, from Remark 7, (i), we find that Algorithm 8 (i.e., BAGRMiATTCG) particularly includes an alternating golden ratio and multi-step inertial algorithm with three-term conjugate gradient-like direction (AGRMiATTCG) when $\xi = \frac{1}{\phi}$, and $\phi = \frac{\sqrt{5}+1}{2}$.

(ii) If $S_n = \{0\}$, and $\zeta_{n,0} = \zeta_n$, $\forall n \geq 1$, BAGRMiATTCG becomes a beyond the alternating golden ratio inertial algorithm with three-term conjugate gradient-like direction (BAGRiATTCG), which similarly includes a three-term conjugate gradient-like and inertial alternating golden ratio-based algorithm TTCiAGRA when $\xi = \frac{1}{\phi}$, and $\phi = \frac{\sqrt{5}+1}{2}$ as a particular case.

(iii) If $\zeta_{n,k} = 0$, $\forall n \geq 1$, and $k \in S_n$, BAGRiATTCG reduces to a beyond the alternating golden ratio and three-term conjugate gradient-like algorithm (BAGRTTCGA), which similarly includes a three-term conjugate gradient-like and alternating golden ratio-based algorithm (TTCAGRA) when $\xi = \frac{1}{\phi}$, and $\phi = \frac{\sqrt{5}+1}{2}$ as a particular case.

(iv) If $\xi = 0$, BAGRiATTCG reduces to the same algorithm provided in Remark 6, (iv).

5 Numerical experiments

This section is devoted to investigating the performance and efficiency of the proposed algorithms in addressing classification problems and solving constrained minimization problems. We conduct the experiments using MATLAB R2023b on a PC with a 12th Gen Intel(R) Core(TM) i5-124P 1.70 GHz processor and 16.0 GB of RAM.

The constrained minimization problem

In this part, we consider the following constrained minimization problem:

$$\min_{x\in\mathcal{C}}\frac{1}{2}\|\mathcal{B}x-P_{\mathcal{Q}}\mathcal{B}x\|^2, \tag{66}$$

where $\mathcal{C}=\{x\in L_2[0,1]:\langle x(t),3t^2\rangle=0\}$ and $\mathcal{Q}=\{x\in L_2[0,1]:\langle x(t),\frac{t}{3}\rangle\geq -1\}$ are in $L_2[0,1]$.

The problem (66) can be transformed into problem (1) with $\mathcal{H}_1=\mathcal{H}_2=L_2[0,1]$. The norm and inner product in $L_2[0,1]$ are defined by $\|x\|=\left(\int_0^1|x(t)|^2dt\right)^{1/2}$ and $\langle x,y\rangle=\int_0^1 x(t)y(t)dt$, respectively. In all experiments, we consider $\mathcal{B}=I$, where I is the identity mapping, i.e., $\mathcal{B}x=x$. Since $\mathcal{Q}$ and $\mathcal{C}$ are half-space and hyper-plane, respectively, to apply the proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG), we take $\mathcal{Q}_n=\mathcal{Q}$ for all $n\geq 1$ and $\hat{\mathcal{C}}=\mathcal{C}$ with $c(x)=\langle x(t),3t^2\rangle$ for all $x\in L_2[0,1]$, so that $\hat{\mathcal{C}}$ satisfies (4). We consider $\mathcal{C}_n, g_n$ and its gradient ∇g_n as defined in (5) and (2), respectively. We use the explicit projection formula provided in [57] to compute the projection $P_{\mathcal{Q}_n}$ and the projection $P_{\mathcal{C}_n}$ as follows

$$P_{\mathcal{C}_n}(u_n(t))=\begin{cases}u_n(t), & \text{if } \langle 3t^2,t-u_n(t)\rangle\geq c(u_n(t)),\\ u_n(t)-\frac{c(u_n(t))+\langle 3t^2,u_n(t)-t\rangle}{\|3t^2\|^2_{L_2}}3t^2, & \text{otherwise.}\end{cases}$$

We compare the performance results of the proposed algorithms to the algorithms of Tan et al. [5], Dong et al. [13], and Suantai et al. [3], abbreviated in this work as TQW Alg 3.1, DLY Alg 4-II, and SPAW Alg 3.3, respectively. For the experiments, we select the following parameters:

(1) We set $\tau_1=0.4641,\ \eta_2=5,\ \varepsilon=0.1,\ \rho=5.001,\ \nu=50,\ \phi=\frac{\sqrt{5}+1}{2},\ \xi=\frac{1}{\phi},\ \beta_n=\frac{1}{10^4n+10},\ \delta_n=\frac{\beta_n(1-\xi)}{\nu}$ and $\sigma_{n,k}=\frac{\delta_n^2}{k^{2.2}(1-\delta_n)}$ for the BGRGMiA and BGRMiATTCG. In particular, we select $\eta_1=5$ and $\varsigma_{n,k}=\frac{1}{n^2k^{2.4}}$ for BGRGMiA and $\sigma=0.1,\ \lambda=0.01,\ \lambda_n=\delta_n^3$ and $\varsigma_n^{(2)}=\delta_n^3$ for BGRMiATTCG.
(2) We choose $\tau_1=0.4641,\ \eta_1=5,\ \varepsilon=0.1,\ \rho=5.001,\ \phi=\frac{\sqrt{5}+1}{2},\ \xi=\frac{1}{\phi},\ \beta_n=\frac{1}{10^4n+10},\ \mu_n=\frac{3\beta_n(1-\xi)}{4}$ and $\sigma_{n,k}=\frac{\mu_n^{1.1}}{3.5(1-\mu_n)k^3}$ for BAGRGMiA and BAGRMiATTCG. In particular, we set $\eta_1=5$ and $\varsigma_{n,k}=\frac{1}{n^2k^{2.4}}$ for BAGRGMiA and $\sigma=0.1,\ \lambda=0.01,\ \lambda_n=\mu_n^3$ and $\varsigma_n^{(2)}=\mu_n^3$ for BAGRMiATTCG.
(3) In TQW Alg 3.1, we adopted $\lambda_1=0.4641,\ \mu=0.1,\ \beta=1.3,\ \alpha=1,\ \theta_n=0.2,\ \rho_n=\frac{10^{-1}}{(n+1)^2}$ and $\xi_n=1+\frac{10^{-1}}{(n+1)^2}$ in [5].
(4) In DLY Alg 4-II, we set $\tau_1=0.4641,\ \varepsilon=0.1,\ \rho=0.2$ and $\lambda_n=\frac{1}{50n+1}-1$.
(5) In SPAW Alg 3.3, we set $\rho_1=\rho_2=0.1, \lambda_1=0.4641, \alpha_n=0.5$ and $\bar{\sigma}_n=\frac{2^{13}}{\|u_n-u_{n-1}\|_L^3+n^3+2^{13}}$.

For the implementations of the algorithms, we consider four different cases of the initial values of $u_0(s),\ u_1(s),\ v_0(s)$and $x_n(s)$:
Case I: $u_0(s)=v_0(s)=\frac{\sin(s^2)}{10},\ u_1(s)=s^5,$ and $x_n(s)=2\sqrt[4]{s}$;
Case II: $u_0(s)=v_0(s)=\frac{\cos(s^2)}{100},\ u_1(s)=\frac{\sqrt{s}}{10}$ and $x_n(s)=10e^{s^3}$;
Case III: $u_0(s)=v_0(s)=11\cosh(s^2),\ u_1(t)=\frac{s^3}{100}$ and $x_n(s)=10e^{s^3}$;
Case IV: $u_0(s)=v_0(s)=s^2,\ u_1(s)=e^{s^5}$ and $x_n(s)=5\tanh(s^3)$.

We applied the stopping criterion

$$E_n = \frac{1}{2}\left(\|u_n(t) - P_{C_n}u_n(t)\|^2_{L_2} + \|u_n(t) - P_{Q_n}u_n(t)\|^2_{L_2}\right) < 10^{-10},$$

and maximum of 200 iterations to terminate the process for all the algorithms. We use the number of iterations denoted by "Iter.", the execution time in seconds denoted by "Time (s)" and the error E_n to investigate the performance of all algorithms. Table 1 reports the numerical results of the suggested algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG), along with some of their respective variants in Remark 4, Remark 6, Remark 9 and Remark 10 for Case I. The corresponding error results are depicted in Figure 1. The performance results of all algorithms for the four cases are reported in Table 2 and the corresponding error results are depicted in Figure 2. However, in Figure 2c, we omitted the error trajectory for SPAW Alg 3.3, because its number of iterations to reach the stopping rule in this case is very large.

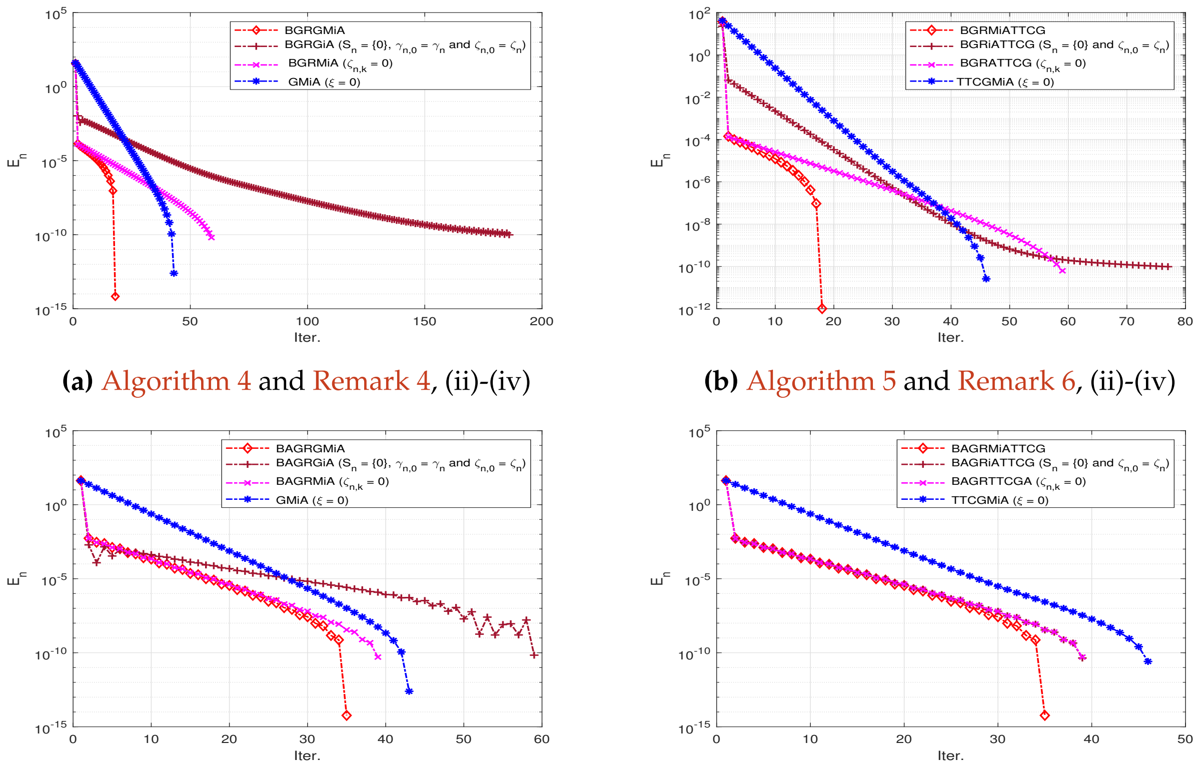
Table 1. Compare the performance of BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG, and their variants in Remark 4, Remark 6, Remark 9 and Remark 10, respectively, for Case I

Case I							
Algorithms	Iter.	Time (s)	E_n	Algorithms	Iter.	Time (s)	E_n
BGRGMiA	18	0.0301	6.87E-15	BGRMiATTCG	18	0.048	1.01E-12
BGRGiA	186	0.144	9.82E-11	BGRiATTCG	77	0.0832	9.74E-11
BGRMiA	59	0.0649	6.40E-11	BGRATTCG	59	0.0757	6.23E-11
GMiA	43	0.0509	2.49E-13	TTCGMiA	46	0.0642	2.58E-11
BAGRGMiA	35	0.0232	5.85E-15	BAGRMiATTCG	35	0.0224	5.85E-15
BAGRGiA	59	0.0365	6.85E-11	BAGRiATTCG	39	0.0276	4.40E-11
BAGRMiA	39	0.0282	5.08E-11	BAGRTTCGA	39	0.0298	5.08E-11
GMiA	43	0.0284	2.49E-13	TTCGMiA	46	0.0423	2.58E-11

Remark 11 From Table 1 and Figure 1, we observe that each of the proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA, and BAGRMiATTCG) reaches the smallest error within the shortest execution time and fewest iterations than its corresponding variants in Remark 4, Remark 6, Remark 9, and Remark 10, respectively. We also find that the proposed algorithms based on the golden ratio extrapolation step (10) (i.e., BGRGMiA and BGRMiATTCG) have fewer iterations than those developed based on the alternating golden ratio extrapolation step (11) (i.e., BAGRGMiA and BAGRMiATTCG), while BAGRGMiA and BAGRMiATTCG reach the smallest errors and shorter execution times than BGRGMiA and BGRMiATTCG.

Remark 12 From Table 2 and Figure 2, we make the following remarks.

- Each of the proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA, and BAGRMiATTCG) achieves the smallest error, shorter execution time, and fewer iterations than any of the compared algorithms (i.e., TQW Alg 3.1, DLY Alg 4-II, and SPAW Alg 3.3) in all four experimental cases.
- The proposed algorithms based on the golden ratio extrapolation step (10) (i.e., BGRGMiA and BGRMiATTCG) appear to be faster in terms of the number of iterations and execution times than those based on the alternating golden ratio extrapolation (11) (i.e., BAGRGMiA and BAGRMiATTCG) in Cases I, II, and IV. They are also found to achieve smaller errors than



(a) Algorithm 4 and Remark 4, (ii)-(iv)

(b) Algorithm 5 and Remark 6, (ii)-(iv)

(c) Algorithm 7 and Remark 9, (ii)-(iv)

(d) Algorithm 8 and Remark 10, (i)-(iii)

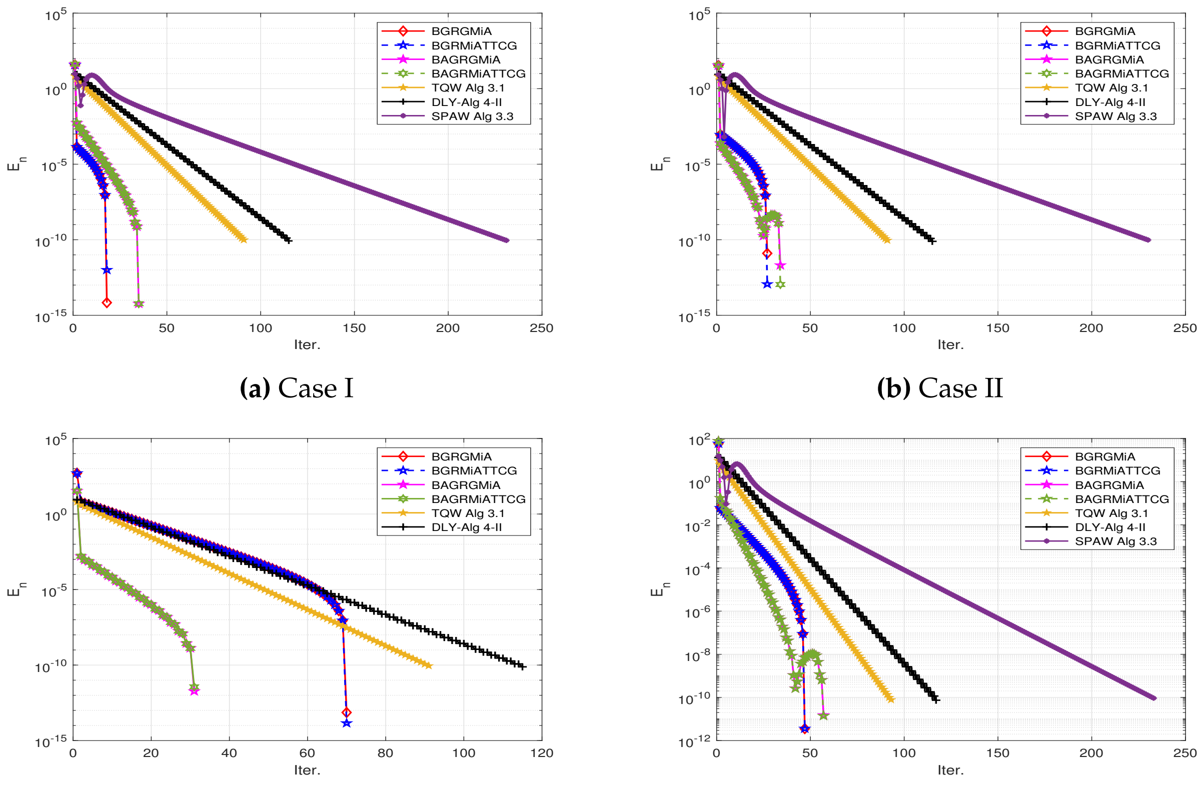
Figure 1. Error plotting of E_n of BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG, and their variants in Remark 4, Remark 6, Remark 9 and Remark 10, respectively for Case I

Table 2. Performance results of the algorithms for the four cases

	Case I			Case II		
Algorithms	Iter.	Time (s)	E_n	Iter.	Time (s)	E_n
BGRGMiA	18	0.0464	6.87E-15	27	0.0323	1.26E-11
BGRMiATTCG	18	0.032	1.01E-12	27	0.0295	1.15E-13
BAGRGMiA	35	0.0664	5.85E-15	34	0.0349	2.06E-12
BAGRMiATTCG	35	0.0495	5.85E-15	34	0.0353	1.09E-13
TQW Alg 3.1	91	0.0915	9.79E-11	91	0.0579	9.16E-11
DLY Alg 4-II	115	0.1046	8.36E-11	115	0.0693	7.82E-11
SPAW Alg 3.3	231	0.1813	9.07E-11	230	0.1552	9.68E-11

	Case III			Case IV		
Algorithms	Iter.	Time (s)	E_n	Iter.	Time (s)	E_n
BGRGMiA	70	0.06	7.26E-14	47	0.0462	3.58E-12
BGRMiATTCG	70	0.0482	1.43E-14	47	0.0368	3.61E-12
BAGRGMiA	31	0.0454	1.85E-12	57	0.0503	1.52E-11
BAGRMiATTCG	31	0.0359	3.60E-12	57	0.0457	1.39E-11
TQW Alg 3.1	91	0.0631	8.93E-11	93	0.0668	7.80E-11
DLY Alg 4-II	115	0.0804	7.63E-11	117	0.0742	7.43E-11
SPAW Alg 3.3	3160	73.2561	9.97E-11	233	0.169	9.28E-11

BAGRGMiA and BAGRMiATTCG in Cases III and IV. However, it is observed that BAGRGMiA and BAGRMiATTCG have fewer iterations and shorter execution times than BGRGMiA and BGRMiATTCG in Case III, and each of them comparatively achieves a smaller error than its corresponding counterpart (i.e., BGRGMiA and BGRMiATTCG) in Cases I and II, respectively.



(a) Case I **(b)** Case II

(c) Case III **(d)** Case IV

Figure 2. Error plotting of E_n of all algorithms for all the cases

Classification problems

In this part, we conduct a series of experiments to analyze the performance of the proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG), along with the algorithms of Dong et al. [13] and Suantai et al. [3], abbreviated as DLY Alg 4-II and SPAW Alg 3.3, respectively, in addressing classification problems for an interesting real-world dataset. In all the experiments, we employ an efficient learning algorithm called extreme learning machine ELM for single-hidden layer feedforward neural networks SLFNs, [58] and take $\mathcal{K} = \{(u_j, t_j) \in \mathbb{R}^k \times \mathbb{R}^m, \ j = 1, 2, \cdots, \mathcal{N}\}$ as an $\mathcal{N}$ distinct training data points set, where for each input point $u_j = \left[u_{j1}, u_{j2}, \cdots, u_{jk}\right]^T$, $t_j = \left[t_{j1}, t_{j2}, \cdots, t_{jm}\right]^T$ is its corresponding target. The network output function with $\mathcal{L}$ nodes in the hidden layer is formulated as follows: $h_j = \sum_{i=1}^{\mathcal{L}} \beta_i f_i(u_j), \quad \forall j = 1, 2, \cdots, \mathcal{N}$, where $f_i(u_j) = \mathcal{F}\Big(\langle \omega_i, u_j \rangle + b_i \Big)$, $\mathcal{F}$ is an activation function, $\omega_i = (\omega_{i1}, \omega_{i2} \cdots, \omega_{ik})^T$ is an input weight vector linking the i^{th} hidden node and the input nodes, $\beta_i = (\beta_{i1}, \beta_{i2}, \cdots, \beta_{im})^T$ is an output weight vector linking the i^{th} hidden node and the output nodes and b_i is a bias of an i^{th} hidden node. To train the network simply means to solve the linear system:

$$\mathcal{G}\beta = \mathsf{T}, \tag{67}$$

where the hidden layer output matrix $\mathcal{G}$ of order $\mathcal{N} \times \mathcal{L}$ is given by $\mathcal{G} = \left[f_1(u), f_2(u), \cdots, f_{\mathcal{L}}(u)\right]$, $\beta = (\beta_1, \beta_2, \cdots, \beta_{\mathcal{L}})^T$ and $\mathsf{T} = (t_1, t_2, \cdots, t_{\mathcal{N}})^T$ are the output weights and the target data matrices, respectively. The i^{th} column of $\mathcal{G}$ is the i^{th} hidden node output based on $u_1, u_2 \cdots, u_{\mathcal{N}}$, which is defined by $f_i(u) = \left[f_i(u_1), f_i(u_2), \cdots, f_i(u_{\mathcal{N}})\right]^T$. To solve (67) by ELM is simply to find an optimal output weight $\hat{\beta} = \mathcal{G}^\dagger \mathsf{T}$, where $\mathcal{G}^\dagger$ represents the Moore-Penrose generalized inverse of the matrix $\mathcal{G}$, [59]. To address the sparsity of the output weight parameter in high-dimensional data and to enhance stability and generalization performance, Ye et al. [60] consider the following constrained minimization problem, which incorporates both the ℓ_1 and the ℓ_2 norms:

$$\min_{\beta \in \mathbb{R}^{\mathcal{L}\times m}} \left\{ \frac{1}{2}\|\mathsf{T} - \mathcal{G}\beta\|_2^2 \ : \ \gamma_1\|\beta\|_1 + \gamma_2\|\beta\|_2^2 \le l \right\}, \tag{68}$$

where $\gamma_1,\ \gamma_2 \ge 0$ and $l > 0$ are the regularization parameters. Inspired by the sparsity, stability, and generalization performance of the model (68), we explore its relationship with problem (1) by considering

$$\mathcal{C} = \beta \in \mathbb{R}^{\mathcal{L}\times m} \ : \ \gamma_1\|\beta\|_1 + \gamma_2\|\beta\|_2^2 \le l,\ \mathcal{Q} = \{\mathsf{T}\} \subseteq \mathbb{R}^{\mathcal{K}\times m} \text{ and } \quad q(x) = \frac{1}{2}\|x - \mathsf{T}\|^2.$$

Moreover, it is easy to see that the function $c(\beta) = \gamma_1\|\beta\|_1 + \gamma 2\|\beta\|_2^2 - l$ is strongly convex and, therefore, convex, thus, $\hat{\hat{\mathcal{C}}}$ and $\hat{\hat{\mathcal{Q}}}$ are taking as $\mathcal{C}$ and $\mathcal{Q}$ in $\mathbb{R}^{\mathcal{L}\times m}$ and $\mathbb{R}^{\mathcal{K}\times m}$, respectively. We then use $\mathcal{C}_n$, $\mathcal{Q}_n$, g_n, and its gradient ∇g_n as defined in (5) and (2), respectively. The proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA, and BAGRMiATTCG), along with DLY Alg 4-II and SPAW Alg 3.3, are applied to solve the problem (67) based on the model (68). We conduct the experiments on the Wisconsin Breast Cancer dataset, [61], which contains 569 instances, 30 real-valued features, and two predictive labels, namely B = benign (i.e., non-cancerous tumor) and M = malignant (i.e., cancerous tumor). The instances are distributed across two classes, including 357 B instances and 212 M instances. Ten real-valued features (i.e., Radius, Texture, Perimeter, Area, Smoothness, Compactness, Concavity, Concave Points, Symmetry, and Fractal Dimension) are computed for each cell nucleus. More details about this dataset can be found in [61]. We performed the experiments after normalizing the original data using

$$\bar{u}_{ji} = \frac{u_{ji} - u_i^{\min}}{u_i^{\max} - u_i^{\min}},$$

where $u_i^{\max} = \max_{j=1,2,\ldots,\mathcal{N}}(u_{ji})$ and $u_i^{\min} = \min_{j=1,2,\ldots,\mathcal{N}}(u_{ji})$ represent the maximum and minimum of i^{th} attribute over all input data points u_j, respectively, and $\bar{u}_{ji}$ represents the normalized value of u_{ji}.

For the implementations of all the algorithms, we use $\mathcal{L} = 100$ and select $\gamma_1 = 0.9999$, $\gamma_2 = 0.00505$, $u_0 = v_0 = u_1 = \frac{1}{100}\text{randn}(\mathcal{L}, 2)$, $x_n = \text{one}(\mathcal{L}, 2)$ and set the parameters of the algorithms as follows:

• We set $\tau_1 = 0.0001,\ \varepsilon = 0.6,\ \rho = \frac{1}{\varepsilon} - 0.5$ and $\beta_n = \frac{1}{n+10}$ and the rest as in the constrained minimization problem Section 5 for BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG.
• In SPAW Alg 3.3, we set $\rho_1 = \rho_2 = 0.6$, $\lambda_1 = 0.0001$, $\alpha_n = 0.3$ and $\bar{\sigma}_n = \frac{2^{13}}{\|u_n - u_{n-1}\|^3 + n^3 + 2^{13}}$.
• In DLY Alg 4-II, we set $\tau_1 = 0.0001,\ \varepsilon = 0.6,\ \rho = \frac{1}{\varepsilon} - 0.5$ and $\lambda_n = \frac{1}{1.5n} - 1$.
Throughout the experiments, we use a stopping criterion of $\|u_{n+1} - u_n\| < 10^{-5}$, and a maximum of 200 iterations to terminate the process for all algorithms. We applied the execution time in seconds denoted by "Time (s)", number of iterations denoted by "Iter.", Area Under the ROC Curve (i.e., Receiver Operating Characteristics curve) denoted by "AUC" and accuracy as metrics to

evaluate the performance of each algorithm. The accuracy is calculated as

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

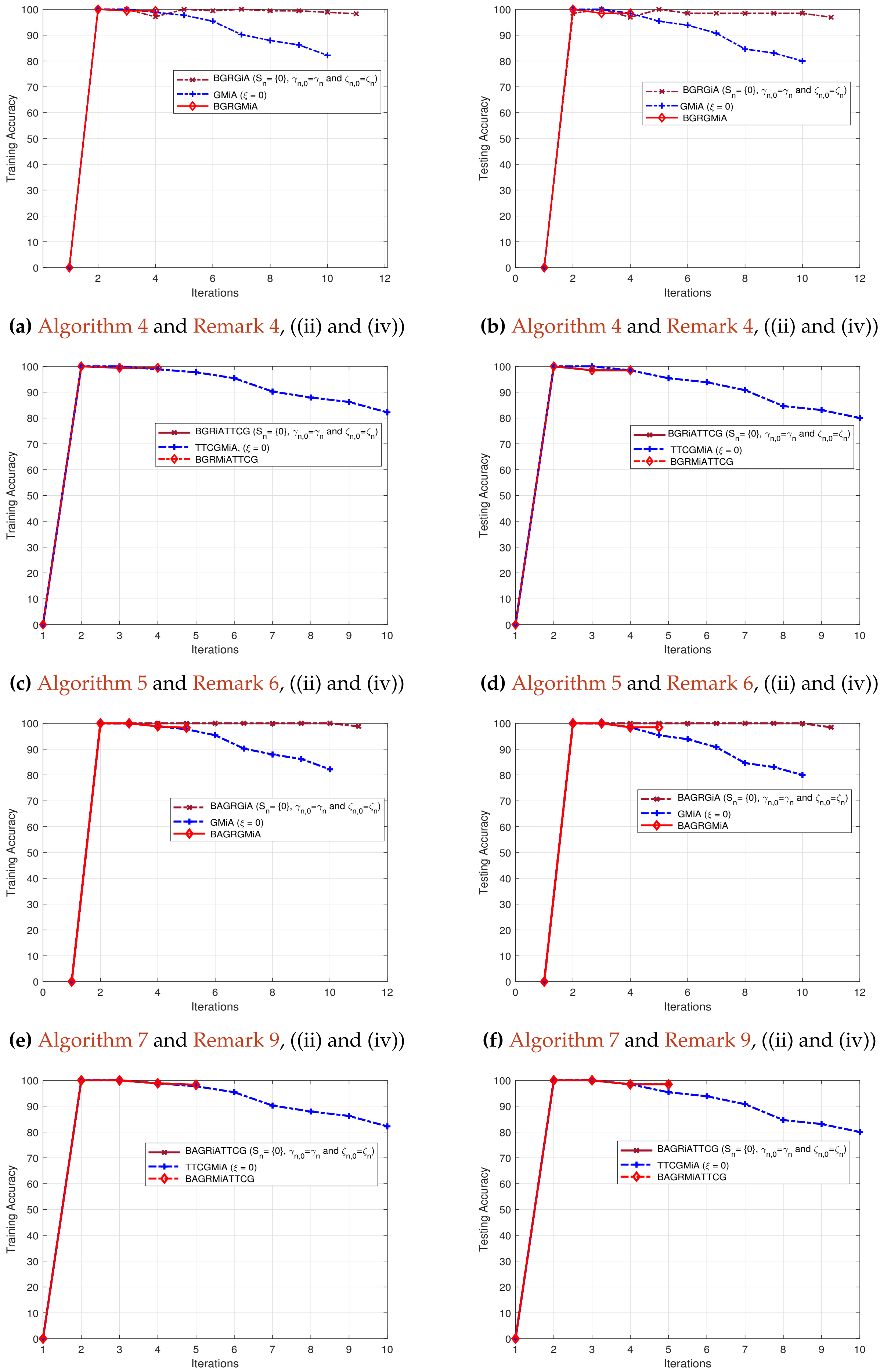
where TP := True Positive, TN := True Negative, FP = False Positive and FN = False Negative. In the first part of the experiments, we consider the proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG), along with some of their variants in Remark 4, Remark 6, Remark 9 and Remark 10 using the sigmoid activation function (i.e., $\mathcal{F}(x) = \frac{1}{1+e^{-x}}$) and $\rho = \frac{1}{\varepsilon} - 1$. The performance results are reported in Table 3 and the corresponding training and testing accuracy trajectories are displayed in Figure 3. In the second part of the experiments, we analyze the sensitivity of all algorithms for eight different activation functions and the regularization parameter l is selected according to the activation functions shown in Table 5. The performance results of all algorithms are shown in Table 5. The corresponding training and testing accuracy trajectories for all algorithms are shown in Figure 4 and Figure 5. Similarly, to demonstrate each algorithmic classification performance, the ROC curves of all algorithms are depicted in Figure 6, where the approach with the highest AUC value is said to be better at distinguishing positive and negative classes. Furthermore, we report the comparison results of all algorithms based on the number of wins, ties, and losses in Table 4, which summarizes the performance results in Table 5.

Table 3. Performance results of BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG, and some of their variants in Remark 4, Remark 6, Remark 9 and Remark 10, respectively

		Time (s)		Accuracy %	
Algorithms	Iter.	Training	Testing	Training	Testing
BGRGMiA	4	0.0512	0.0018	99.4253	98.4615
BGRGiA	11	0.0875	0.002	98.2759	96.9231
GMiA	10	0.0975	0.002	82.1839	80
BGRMiATTCG	4	0.0462	0.0016	99.4253	98.4615
BGRiATTCG	4	0.0491	0.0021	99.4253	98.4615
TTCGMiA	10	0.0932	0.0019	82.1839	80
BAGRGMiA	5	0.058	0.0015	98.2759	98.4615
BAGRGiA	11	0.0856	0.0022	98.8506	98.4615
GMiA	10	0.0947	0.0017	82.1839	80
BAGRMiATTCG	5	0.0476	0.0017	98.2759	98.4615
BAGRiATTCG	5	0.0524	0.0021	98.2759	98.4615
TTCGMiA	10	0.0842	0.0018	82.1839	80

Remark 13 From the performance results reported in Table 3 and Figure 3, we make the following remarks.

- We see that each of the proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA, and BAGRMiATTCG) has the shortest training and testing times than its variants in (ii) and (iv) of Remark 4, Remark 6, Remark 9 and Remark 10, respectively.
- We also observe that each of BGRGMiA and BAGRGMiA, which unifies two steps of the procedure (9) with $k > 2$ (i.e., multi-step inertial terms) and new extrapolation steps based on (10) and (11), respectively, has fewer iterations than their respective variants in (ii) and (iv) of Remark 4 and Remark 9.



(a) Algorithm 4 and Remark 4, ((ii) and (iv)) **(b)** Algorithm 4 and Remark 4, ((ii) and (iv))

(c) Algorithm 5 and Remark 6, ((ii) and (iv)) **(d)** Algorithm 5 and Remark 6, ((ii) and (iv))

(e) Algorithm 7 and Remark 9, ((ii) and (iv)) **(f)** Algorithm 7 and Remark 9, ((ii) and (iv))

(g) Algorithm 8 and Remark 10, ((ii) and (iv)) **(h)** Algorithm 8 and Remark 10, ((ii) and (iv))

Figure 3. Accuracy performance of GMiGRA, TTCMiGRA, GMiAGRA and TTCMiAGRA, and their variants in Remark 4, Remark 6, Remark 9 and Remark 10 respectively

- Eventhough, the procedure (9) with $k > 2$ (i.e., multi-step inertial technique) is not applied in the variant of BGRMiATTCG and BAGRMiATTCG in Remark 4, (ii) and Remark 9, (ii) (i.e., BGRiATTCG and BAGRiATTCG), respectively, we noticed that the presence of the three-term conjugate gradient-like direction (13) in these algorithms improves their performance to maintain the same iterations and accuracies of BGRMiATTCG and BAGRMiATTCG, respectively.
- In all the experiments, we found that the variant in (iv) of Remark 4, Remark 6, Remark 9 and Remark 10, without golden ratio and alternating golden ratio-based extrapolation steps (i.e., (20) and (53)) appear to be slower with the highest iterations and training times, and lowest training and testing accuracies than the proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG) and their variants in (ii) of Remark 4, Remark 6, Remark 9 and Remark 10, based on (20) and (53) extrapolation steps. These signify the advantages of incorporating the new extrapolation techniques (20) and (53), which are respectively based on the golden ratio and alternating golden ratio-based extrapolation steps (10) and (11) in these experiments.

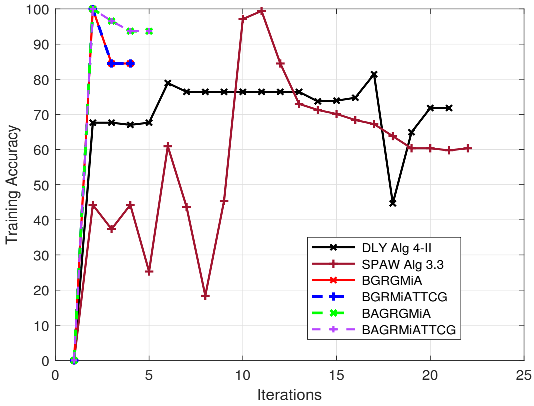
Table 4. Number of wins, ties and losses of all algorithms

	wins / ties / losses			
	BGRGMiA vs. DLY Alg 4-II	BGRMiATTCG vs. DLY Alg 4-II	BAGRGMiA vs. DLY Alg 4-II	BAGRMiATTCG vs. DLY Alg 4-II
Training Acc.	8/0/0	8/0/0	8/0/0	8/0/0
Testing Acc.	8/0/0	8/0/0	8/0/0	8/0/0
iter.	7/1/0	7/1/0	6/1/1	6/1/1
Time (s)	8/0/0	8/0/0	7/0/1	7/0/1
AUC	8/0/0	8/0/0	8/0/0	8/0/0
	BGRGMiA vs. SPAW Alg 3.3	BGRMiATTCG vs. SPAW Alg 3.3	GMiAGRA vs. SPAW Alg 3.3	TTCMiAGRA vs. SPAW Alg 3.3
Training Acc.	8/0/0	8/0/0	8/0/0	8/0/0
Testing Acc.	8/0/0	8/0/0	8/0/0	8/0/0
iter.	8/0/0	8/0/0	8/0/0	8/0/0
Time (s)	8/0/0	8/0/0	8/0/0	8/0/0
AUC	8/0/0	8/0/0	8/0/0	8/0/0
	BGRGMiA vs. BAGRGMiA			BGRMiATTCG vs. BAGRMiATTCG
Training Acc.	0/0/8			0/0/8
Testing Acc.	1/0/7			1/0/7
Iter.	7/1/0			7/1/0
Time (s)	6/0/2			6/0/2
AUC	5/0/3			5/0/3

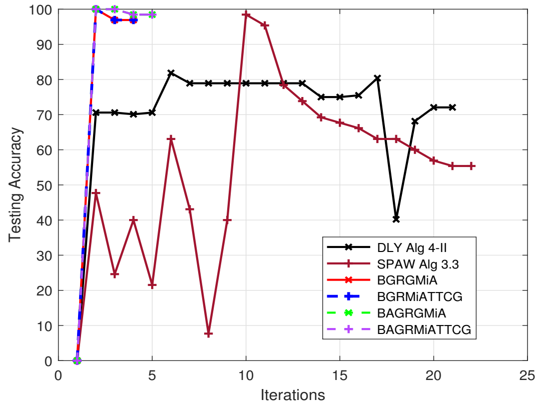
Remark 14 To further investigate the comparative performance of the proposed algorithms in these practical applications, we perform more statistical analyses. In this regard, we used the Wilcoxon signed ranks and the sign test, [62] as statistical methods to compare the reported results of all algorithms in Table 5 and Table 4, as well as Figure 4, Figure 5 and Figure 6. Consequently, based on the Wilcoxon signed ranks, we obtained the following observations.

- The proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA, and BAGRMiATTCG) considerably achieve higher accuracies and AUC, with fewer iterations and shorter execution times compared to DLY Alg 4-II and SPAW Alg 3.3 across all experiments.
- In particular, the proposed algorithms BAGRGMiA and BAGRMiATTCG, which are based on the alternating golden ratio extrapolation step (11), achieve notably better training and testing accuracy than those based on the golden ratio extrapolation step (10) (i.e., BGRGMiA and BGRMiATTCG). Meanwhile, each of the BGRGMiA and BGRMiATTCG, attains the highest AUC values in most cases and appears to be faster, requiring fewer iterations and shorter execution times than BAGRGMiA and BAGRMiATTCG, respectively.

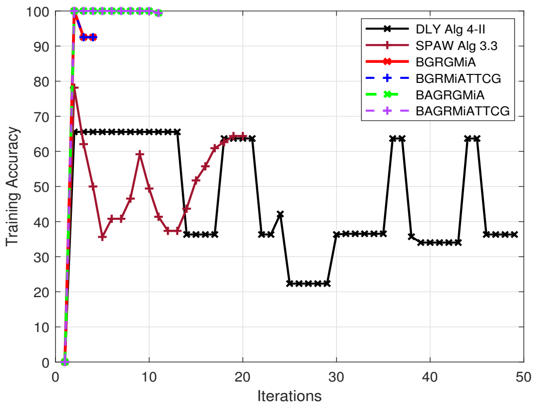
On the other hand, using the null hypothesis in the sign test [62], it is observed that the normal



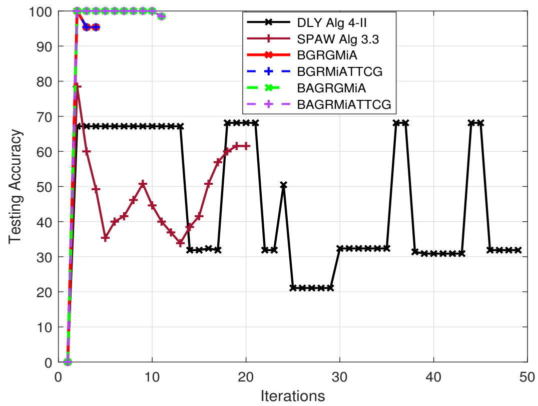
(a) Hardlim



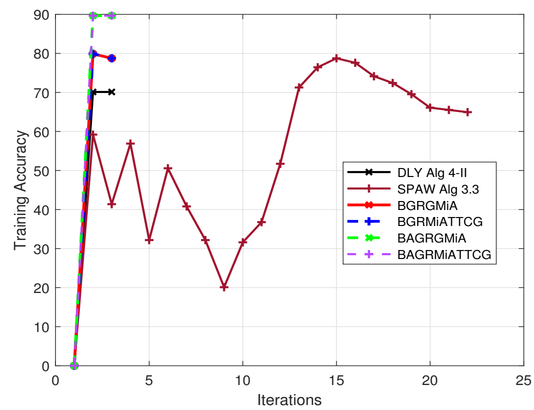
(b) Hardlim



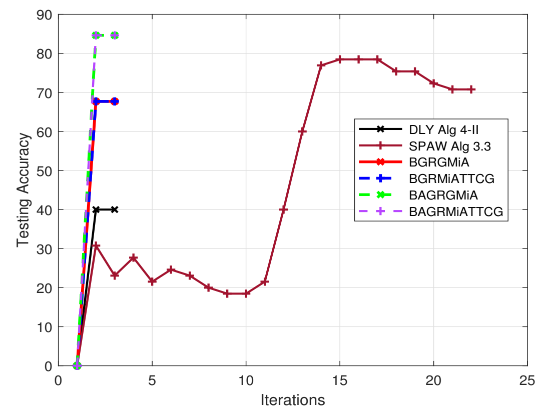
(c) Radbas



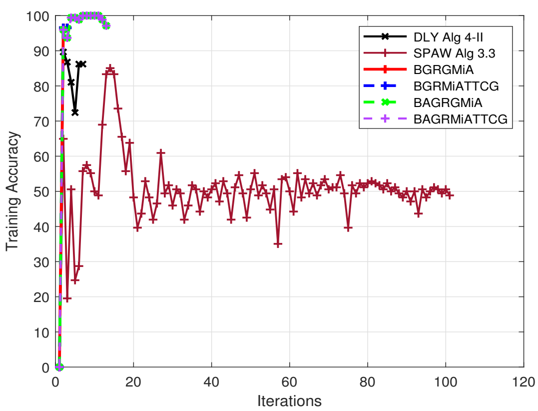
(d) Radbas



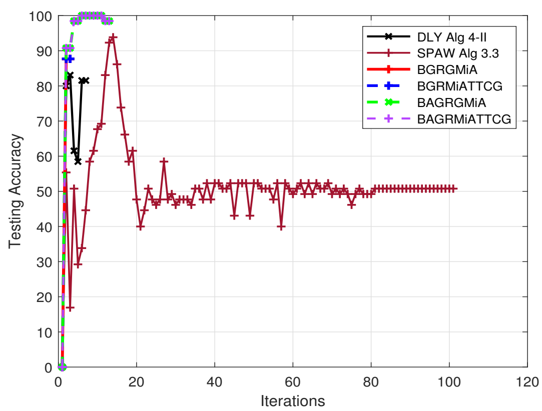
(e) ELU



(f) ELU

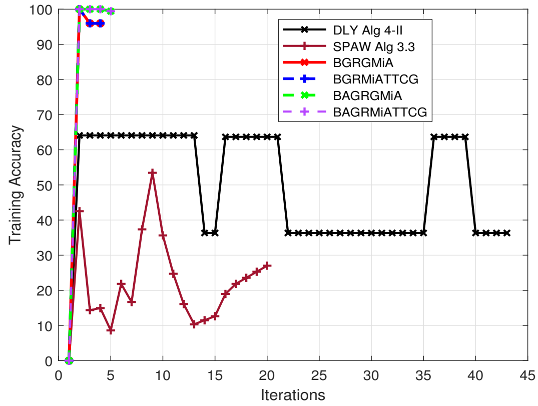


(g) ReLU

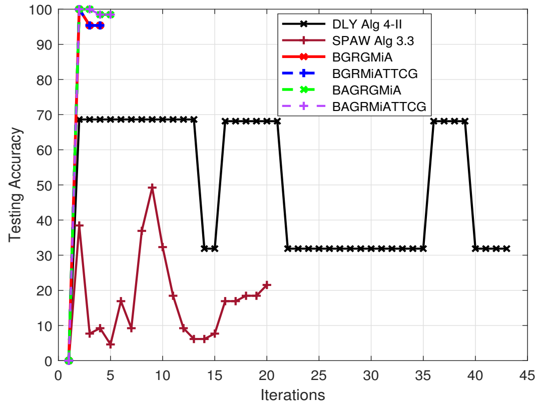


(h) ReLU

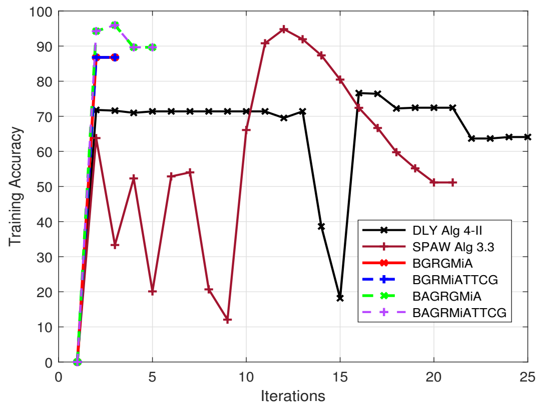
Figure 4. Accuracy performance of all algorithms for four activation functions



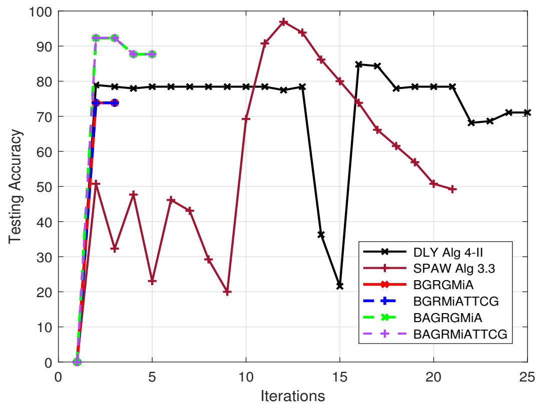
(a) Sigmoid



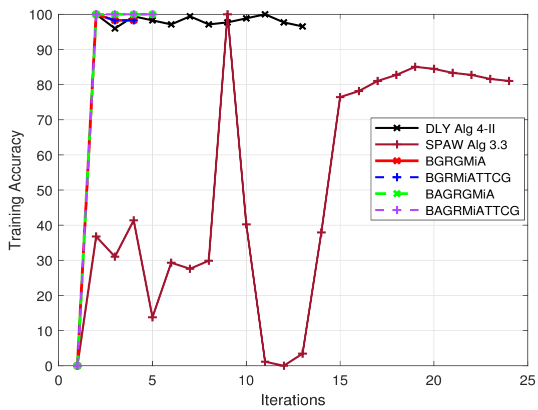
(b) Sigmoid



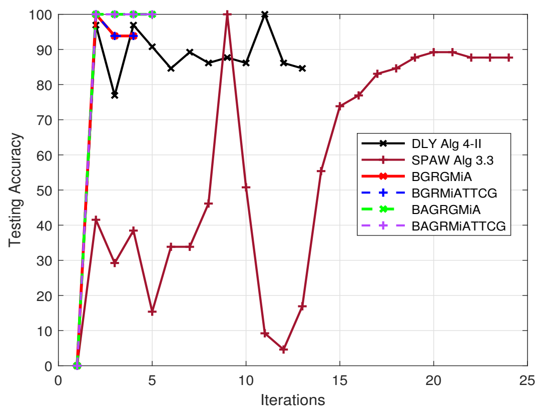
(c) Swish



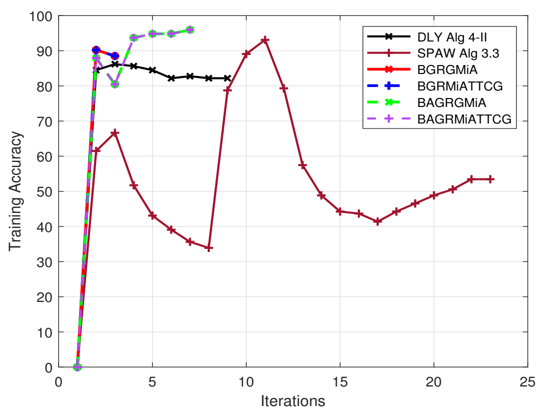
(d) Swish



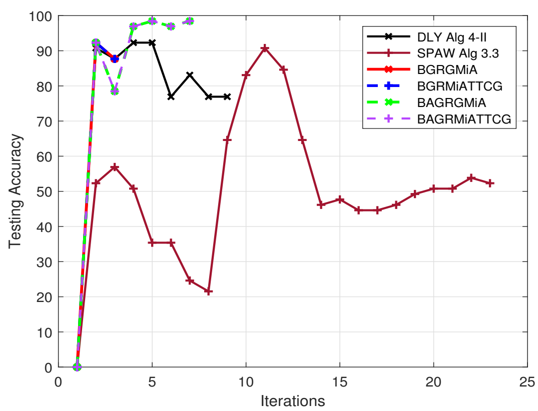
(e) SoftPlus



(f) SoftPlus

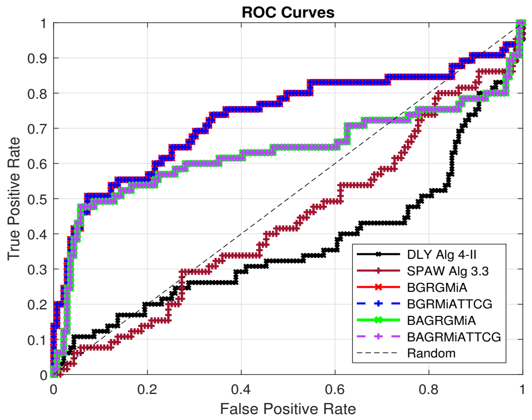


(g) Tribas

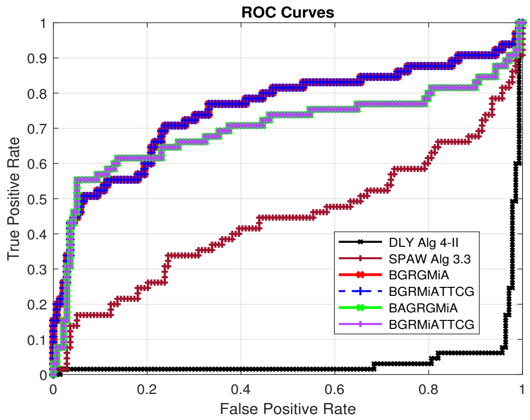


(h) Tribas

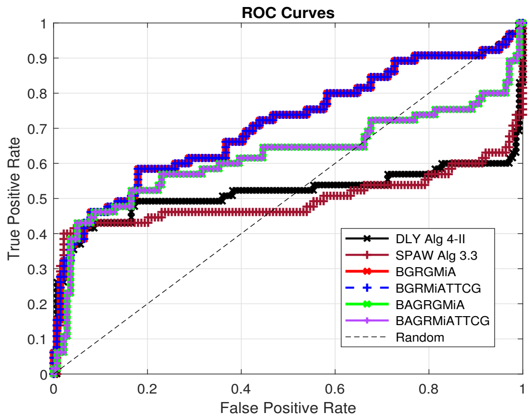
Figure 5. Accuracy performance of all algorithms for four activation functions



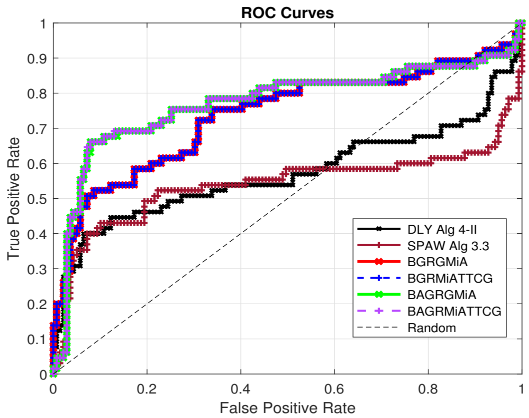
(a) Hardlim



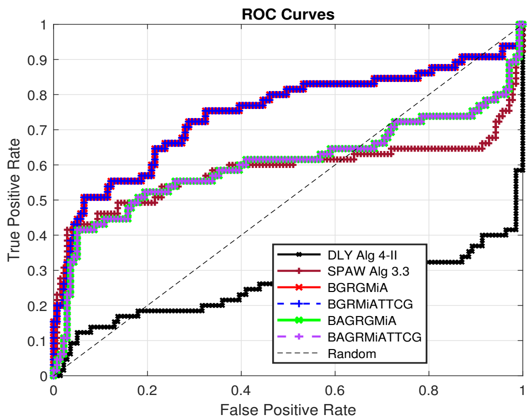
(b) Radbas



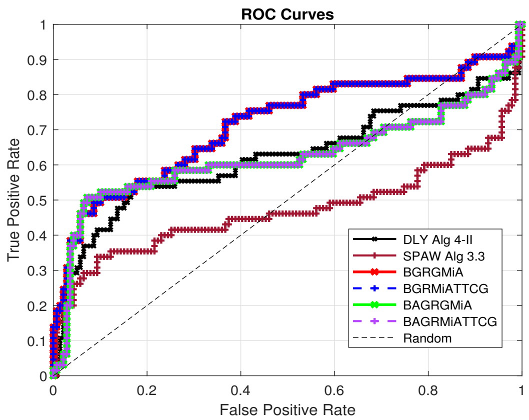
(c) ELU



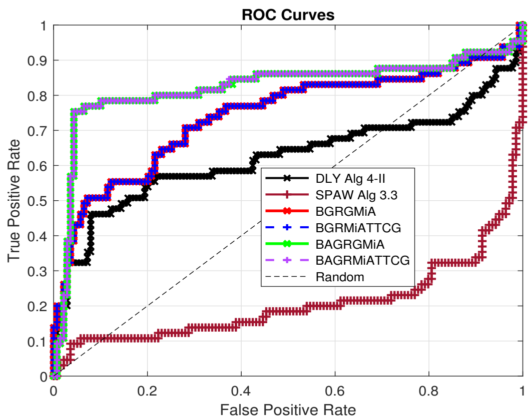
(d) RELU



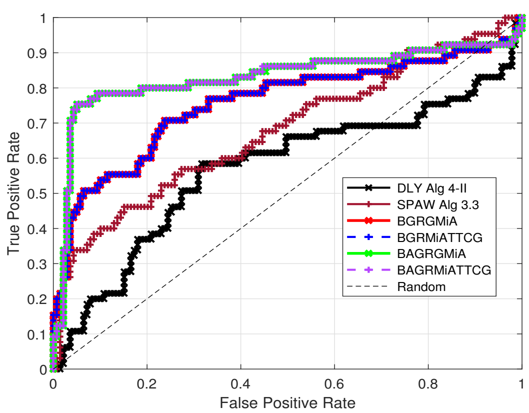
(e) Sigmoid



(f) Swish



(g) SoftPlus



(h) Tribas

Figure 6. AUC-ROC analysis of all algorithms for all activation functions

Table 5. Performance results of all the algorithms for eight activation functions

Act. func.	Hardlim						Radbas					
			Time (s)	Accuracy %					Time (s)	Accuracy %		
Algorithms	l	iter.	Training	Training	Testing	AUC	l	iter.	Training	Training	Testing	AUC
BGRGMiA	0.82	4	0.0587	84.4828	96.9231	0.7319	2.5	4	0.0478	92.5287	96.9231	0.7543
BGRMiATTCG		4	0.0468	84.4828	96.9231	0.7319		4	0.0401	92.5287	96.9231	0.7543
BAGRGMiA		5	0.0661	93.6782	98.4615	0.6338		11	0.0941	99.4253	98.4615	0.7013
BAGRMiATTCG		5	0.0528	93.6782	98.4615	0.6338		11	0.0927	99.4253	98.4615	0.7013
DLY Alg 4-II		21	0.1572	71.8163	72.0588	0.3714		49	0.2261	36.3257	31.8627	0.0417
SPAW Alg 3.3		22	0.1653	60.3448	55.3846	0.4373		20	0.1587	64.3678	61.5385	0.4414

Act. func.	ELU						RELU					
			Time (s)	Accuracy %					Time (s)	Accuracy %		
Algorithms	l	iter.	Training	Training	Testing	AUC	l	iter.	Training	Training	Testing	AUC
BGRGMiA	0.75	3	0.0433	78.7356	98.4615	0.7104	0.5	3	0.0304	96.5517	87.6923	0.7374
BGRMiATTCG		3	0.0352	78.7356	98.4615	0.7104		3	0.0296	96.5517	87.6923	0.7374
BAGRGMiA		3	0.0338	89.6552	84.6154	0.6207		13	0.0951	97.1264	98.4615	0.7726
BAGRMiATTCG		3	0.0294	89.6552	84.6154	0.6207		13	0.0952	97.1264	98.4615	0.7728
DLY Alg 4-II		3	0.0522	70.1149	40	0.5159		7	0.0866	86.2069	81.5385	0.5716
SPAW Alg 3.3		22	0.1535	64.9425	70.7692	0.4952		101	0.3712	48.8506	50.7692	0.5399

Act. func.	Sigmoid						Swish					
			Time (s)	Accuracy %					Time (s)	Accuracy %		
Algorithms	l	iter.	Training	Training	Testing	AUC	l	iter.	Training	Training	Testing	AUC
BGRGMiA	2.5	4	0.0575	95.977	95.3846	0.7443	0.76	3	0.031	86.7816	73.8462	0.7142
BGRMiATTCG		4	0.0518	95.977	95.3846	0.7443		3	0.032	86.7816	73.8462	0.7142
BAGRGMiA		5	0.0549	99.4253	98.4615	0.6031		5	0.0525	89.6552	87.6923	0.6234
BAGRMiATTCG		5	0.0475	99.4253	98.4615	0.6031		5	0.0518	89.6552	87.6923	0.6234
DLY Alg 4-II		43	0.2355	36.3257	31.8627	0.2494		25	0.1414	64.0919	71.0784	0.6228
SPAW Alg 3.3		20	0.1443	27.0115	21.5385	0.5826		21	0.1345	51.1494	49.2308	0.4711

Act. func.	SoftPlus						Tribas					
			Time (s)	Accuracy %					Time (s)	Accuracy %		
	l	iter.	Training	Training	Testing	AUC	l	iter.	Training	Training	Testing	AUC
BGRGMiA	0.7	4	0.0608	98.2759	93.8462	0.742	0.705	3	0.0445	88.5057	87.6923	0.7563
BGRMiATTCG		4	0.0436	98.2759	93.8462	0.742		3	0.033	88.5057	87.6923	0.7563
BAGRGMiA		5	0.0733	100	100	0.8223		7	0.0781	95.977	98.4615	0.831
BAGRMiATTCG		5	0.0506	100	100	0.8224		7	0.0813	95.977	98.4615	0.8311
DLY Alg 4-II		13	0.1494	96.5517	84.6154	0.6236		9	0.1024	82.1839	76.9231	0.57
SPAW Alg 3.3		24	0.1535	81.0345	87.6923	0.2065		23	0.1484	53.4483	52.3077	0.6787

distribution $q\left(\frac{q}{2}, \frac{\sqrt{q}}{2}\right)$ is satisfied by the number of wins for an algorithm, where $q = (1 \text{ dataset} \times 8 \text{ activation functions})$. In this test, an algorithm is considered significantly better than another if its number of wins is at least $\frac{q}{2} + Z_{h/2} \times \frac{\sqrt{q}}{2}$, where h denotes the significance level. In our experiments, we set $h = 0.1$, which gives the inequality $6 < \frac{8}{2} + 1.645 \times \frac{\sqrt{8}}{2} < 7$. This implies that an algorithm must achieve at least 7 wins to be considered significantly better. Based on this criterion and the results reported in Table 4, we make the following observations.

- The proposed algorithms (i.e., BGRGMiA, BGRMiATTCG, BAGRGMiA and BAGRMiATTCG) significantly outperform DLY Alg 4-II and SPAW Alg 3.3 in terms of accuracies, AUC, and execution times. They also exhibit significantly fewer iterations than SPAW Alg 3.3. In particular, based on the sign test analysis described above, only BGRGMiA and BGRMiATTCG are found to have significantly fewer iterations than DLY Alg 4-II. Although BAGRGMiA and BAGRMiATTCG generally require fewer iterations than DLY Alg 4-II in most cases, the number of wins with respect to iterations for each, when compared to DLY Alg 4-II, falls short of the threshold required for statistical significance.
- In particular, the proposed algorithms BAGRGMiA and BAGRMiATTCG, which are based on the alternating golden ratio extrapolation step (11), significantly outperform those based on the golden ratio extrapolation step (10) (i.e., BGRGMiA and BGRMiATTCG) in terms of

training and testing accuracies. On the other hand, BGRGMiA and BGRMiATTCG exhibit significantly fewer iterations than BAGRGMiA and BAGRMiATTCG. Meanwhile, the number of wins for BGRGMiA and BGRMiATTCG in terms of AUC values and execution times, when compared to BAGRGMiA and BAGRMiATTCG, respectively, falls below the statistical threshold. Nevertheless, we found that both BGRGMiA and BGRMiATTCG achieve noticeably better AUC values and execution times than BAGRGMiA and BAGRMiATTCG, respectively.

6 Conclusion

This paper introduces four efficient multi-step inertial relaxed algorithms for split feasibility problems in infinite-dimensional real Hilbert spaces. The first and third are general multi-step inertial algorithms with new extrapolation techniques, referred to as the beyond the golden ratio and general multi-step inertial algorithm (BGRGMiA) and the beyond the alternating golden ratio and general multi-step inertial algorithm (BAGRGMiA), respectively. The BGRGMiA and BAGRGMiA algorithms unify two components: two steps of the improved version of the classical inertial term (8), specifically, the procedure (9) with $k > 2$, called the multi-step inertial term, and the new extrapolation techniques (20) and (53), for which the golden ratio and alternating golden ratio extrapolation steps (10) and (11) are particular cases. These techniques significantly enhance the convergence speed of sequences generated by BGRGMiA and BAGRGMiA toward a solution. The second and fourth algorithms are three-term conjugate gradient-like and multi-step inertial methods with new extrapolation procedures, called the beyond the golden Ratio and multi-step inertial algorithm with three-term conjugate gradient-like direction (BGRMiTTCG) and the beyond the alternating golden ratio and multi-step inertial algorithm with three-term conjugate gradient-like direction (BAGRMiTTCG), respectively. These algorithms combine the three-term conjugate gradient-like direction (13), the multi-step inertial term (i.e., the procedure (9) with $k > 2$), and the new extrapolation techniques (20) and (53), respectively. This combination enables the acceleration of their sequences toward a solution. Each of the proposed algorithms operates effectively without requiring the computation of $\|\mathcal{B}\|$ or a line search procedure in each iteration, using the monotonic adaptive step length rule (21). Under some simple and weaker assumptions, the strong convergence of each of these algorithms is established based on the convergence of two proposed algorithms with perturbations and new extrapolation techniques (20) and (53), referred to as the beyond the golden ratio algorithm with perturbations (BGRAP) and the beyond the alternating golden ratio algorithm with perturbations (BAGRAP), respectively, in infinite-dimensional real Hilbert spaces. Additionally, the convergence analysis of each of the proposed three-term conjugate gradient-like algorithms (i.e., BGRMiTTCG and BAGRMiTTCG) is derived without requiring certain restrictive conditions, such as the boundedness assumption in [52], Lemma 5, (iii). This leads to both theoretical and numerical improvements in the convergence properties of some existing conjugate gradient-like algorithms. Finally, the potential applications of the proposed algorithms are investigated in two contexts: classification problems based on the Extreme Learning Machine (ELM) with the ℓ_1–ℓ_2 hybrid regularization approach for detecting breast cancer using a real-world dataset, and constrained minimization problems in infinite-dimensional real Hilbert spaces. In these experiments, the numerical results of the proposed algorithms, which generalize and improve several methods from the literature, such as those in [5, 10, 11, 13, 20–22, 52], demonstrate greater robustness and comparatively better performance in terms of higher accuracies and AUC, lower convergence errors, shorter execution times, and fewer iterations than some related algorithms reported in [3, 5, 13].Finally, in light of the superior performance demonstrated by the proposed methods, it would be interesting for future work to extend them to study more generalized splitting problems, including the split variational inequality problem, split equilibrium problem, and split minimization problem, and investigate their applications to broader real-world contexts, such as cybersecurity investments in supply chain models, fractional delay differential equations, optimal control theory, and nonlinear time-fractional Burgers' equations, see [63–66] and the references therein.

7 Appendix

Table 6. Summary of the mathematical notations used in the paper

Notations	Meaning
$\mathcal{H}_1,\ \mathcal{H}_2,\ H$	Real Hilbert spaces
$\mathcal{C},\ \mathcal{Q}$	Nonempty, closed and convex subsets of $\mathcal{H}_1$ and $\mathcal{H}_2$, respectively
$\mathcal{B}:\mathcal{H}_1\to\mathcal{H}_2$	Bounded and linear operator
$\mathcal{B}^*:\mathcal{H}_2\to\mathcal{H}_1$	Adjoint operator of $\mathcal{B}$
$P_{\mathcal{C}},\ P_{\mathcal{Q}}$	Metric projection operators onto $\mathcal{C}$ and $\mathcal{Q}$, respectively
$\mathbb{R}$	Set of real numbers
$\mathbb{N}$	Set of positive integers
$\mathbb{R}^m$	m-dimensional Euclidean space
$\varnothing$	Empty set
$g:\mathcal{H}_1\to\mathbb{R}$	Function defined by $g(u)=\frac{1}{2}\|(I-P_Q)Bu\|^2$
∇g	Gradient of g defined by $\nabla g(u)=B^*(I-P_Q)Bu$
$c:\mathcal{H}_1\to\mathbb{R}$	Weakly lower semicontinuous and convex function
$q:\mathcal{H}_2\to\mathbb{R}$	Weakly lower semicontinuous and convex function
$\mathring{\mathcal{C}}$	Sub-level set in $\mathcal{H}_1$ defined in (4)
$\mathring{\mathcal{Q}}$	Sub-level set in $\mathcal{H}_2$ defined in (4)
$\partial c(u_n)$	Subdifferential of c at u_n
$\partial q(\mathcal{B}u_n)$	Subdifferential of q at $\mathcal{B}u_n$
$\phi_n\in\partial c(u_n)$	Subgradient of c at u_n
$\varphi_n\in\partial q(\mathcal{B}u_n)$	Subgradient of q at $\mathcal{B}u_n$
$\mathcal{C}_n$	Half-space in $\mathcal{H}_1$ at u_n, defined in (5)
$\mathcal{Q}_n$	Half-space in $\mathcal{H}_2$ at $\mathcal{B}u_n$, defined in (5)
$P_{\mathcal{C}_n},\ P_{\mathcal{Q}_n}$	Metric projection operators onto $\mathcal{C}_n$ and $\mathcal{Q}_n$, respectively
$g_n:\mathcal{H}_1\to\mathbb{R}$	Function defined by $g_n(u)=\frac{1}{2}\|(I-P_{Q_n})Bu\|^2$
∇g_n	Gradient of g defined by $\nabla g_n(u)=B^*(I-P_{Q_n})Bu$.
$\|B\|^2$	Lipschitz constant of ∇g and ∇g_n
T	Nonexpansive mapping
$\neq$	Not equal to

Declarations

Use of AI tools

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Data availability statement

The dataset analyzed in this study is available in https://archive.ics.uci.edu/.

Ethical approval (optional)

This dataset is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) license. This allows for the sharing and adaptation of the datasets for any purpose, provided that the appropriate credit is given.

Consent for publication

All authors unanimously agreed to publish this article in the Mathematical Modeling and Numerical Simulation with Applications.

Conflicts of interest

The authors declare that they have no conflict of interest.

Funding

No funding was received for this research.

Author's contributions

A.A.: Conceptualization, Writing–original draft, Methodology, Investigation, Formal analysis, Writing-Reviewing and editing. P.K.: Methodology, Investigation, Formal analysis, Writing-Reviewing and editing, Supervision. T.S.: Methodology, Investigation, Formal analysis, Writing-Reviewing and editing. All authors discussed the results and agreed to publish the manuscript.

Acknowledgements

The authors acknowledge the financial support of the Center of Excellence in Theoretical and Computational Science (TaCS-CoE), KMUTT. Additionally, the first author acknowledged the support of the Petchra Pra Jom Klao Doctoral Research Scholarship from King Mongkut's University of Technology Thonburi with Contract No. 48/2565 and Center of Excellence in Theoretical and Computational Science (TaCS-CoE), KMUTT. This research was funded by National Science, Research and Innovation Fund (NSRF), King Mongkutís University of Technology, North Bangkok, with Contract no. KMUTNB- FF-68-B-24.

References

[1] Censor, Y. and Elfving, T. A multiprojection algorithm using Bregman projections in a product space. *Numerical Algorithms,* 8, 221–239, (1994). [CrossRef]

[2] Penfold, S., Zalas, R., Casiraghi, M., Brooke, M., Censor, Y. and Schulte, R. Sparsity constrained split feasibility for dose-volume constraints in inverse planning of intensity-modulated photon or proton therapy. *Physics in Medicine & Biology,* 62(9), 3599, (2017). [CrossRef]

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

[4] Dong, Q.L., Li, X.H. and Rassias, T.M. Two projection algorithms for a class of split feasibility problems with jointly constrained Nash equilibrium models. *Optimization,* 70(4), 871–897, (2021). [CrossRef]

[5] Tan, B., Qin, X. and Wang, X. Alternated inertial algorithms for split feasibility problems. *Numerical Algorithms*, 95, 773–812, (2024). [CrossRef]

[6] Ahmad, A., Kumam, P., Yahaya, M.M., Sitthithakerngkiet, K. and Cholamjiak, W. Two-step alternated inertial algorithm for split feasibility problems with some applications. *Journal of Nonlinear and Convex Analysis*, 25(12), 3165-3191, (2024).

[7] Byrne, C. Iterative oblique projection onto convex sets and the split feasibility problem. *Inverse Problems,* 18(2), 441, (2002). [CrossRef]

[8] Yang, Q. The relaxed CQ algorithm solving the split feasibility problem. *Inverse Problems*, 20(4), 1261, (2004). [CrossRef]

[9] Bauschke, H.H. and Combettes, P.L. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer Cham: New York, (2017). [CrossRef]

[10] Qu, B. and Xiu, N. A note on the *CQ* algorithm for the split feasibility problem. *Inverse Problems*, 21(5), 1655, (2005). [CrossRef]

[11] Dong, Q.L., Tang, Y.C., Cho, Y.J. and Rassias, Th.M. "Optimal" choice of the step length of the projection and contraction methods for solving the split feasibility problem. *Journal of Global Optimization*, 71, 341-360, (2018). [CrossRef]

[12] Gibali, A., Liu, L.W. and Tang, Y.C. Note on the modified relaxation CQ algorithm for the split feasibility problem. *Optimization Letters*, 12, 817-830, (2018). [CrossRef]

[13] Dong, Q.L., Liu, L. and Yao, Y. Self-adaptive projection and contraction methods with alternated inertial terms for solving the split feasibility problem. *Journal of Nonlinear and Convex Analysis,* 23(3), 591–605, (2022).

[14] Kratuloek, K., Kumam, P., Sriwongsa, S. and Abubarkar, J. A relaxed splitting method for solving variational inclusion and fixed point problems. *Computational and Applied Mathematics*, 43, 70, (2024). [CrossRef]

[15] Sumalai, P., Abubakar, J., Kumam, P. and Salisu, S. A unified scheme for solving split inclusions with applications. *Mathematical Methods in the Applied Sciences,* 46(13), 14622-14639, (2023). [CrossRef]

[16] Ma, X. and Liu, H. An inertial Halpern-type CQ algorithm for solving split feasibility problems in Hilbert spaces. *Journal of Applied Mathematics and Computing*, 68, 1699–1717, (2022). [CrossRef]

[17] Shehu, Y., Dong, Q.L. and Liu, L.L. Global and linear convergence of alternated inertial methods for split feasibility problems. *Revista de la Real Academia de Ciencias Exactas, Físicas y Naturales. Serie A. Matemáticas*, 115, 53, (2021). [CrossRef]

[18] Abubakar, J., Kumam, P., Taddele, G.H., Ibrahim, A.H. and Sitthithakerngkiet, K. Strong convergence of alternated inertial CQ relaxed method with application in signal recovery. *Computational and Applied Mathematics*, 40, 310, (2021). [CrossRef]

[19] Suantai, S., Panyanak, B., Kesornprom, S. and Cholamjiak, P. Inertial projection and contraction methods for split feasibility problem applied to compressed sensing and image restoration. *Optimization Letters*, 16, 1725-1744, (2022). [CrossRef]

[20] Dang, Y., Sun, J. and Xu, H. Inertial accelerated algorithms for solving a split feasibility problem. *Journal of Industrial and Management Optimization*, 13(3), 1383-1394, (2017). [CrossRef]

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

[22] Vinh, N.T., Hoai, P.T., Dung, L.A. and Cho, Y.J. 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, 2489-2506, (2023). [CrossRef]

[23] Reich, S., Tuyen, T.M. and Van Huyen, P.T. Inertial proximal point algorithms for solving a class of split feasibility problems. *Journal of Optimization Theory and Applications*, 200, 951–977, (2024). [CrossRef]

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

[25] Iiduka, H. Fixed point optimization algorithms for distributed optimization in networked systems. *SIAM Journal on Optimization*, 23(1), 1–26, (2013). [CrossRef]

[26] Polyak, B.T. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5), 1–17, (1964). [CrossRef]

[27] Ahmad, A., Kumam, P., Harbau, M.H. and Sitthithakerngkiet, K. Inertial hybrid algorithm for generalized mixed equilibrium problems, zero problems, and fixed points of some nonlinear mappings in the intermediate sense. *Mathematical Methods in the Applied Sciences*, 47(11), 8527–8550, (2024). [CrossRef]

[28] Harbau, M.H., Ahmad, A., Ali, B. and Ugwunnadi, G.C. Inertial residual algorithm for fixed points of finite family of strictly pseudocontractive mappings in Banach spaces. *International Journal of Nonlinear Analysis and Applications*, 13(2), 2257–2269, (2022).

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

[30] Adamu, A., Kitkuan, D. and Seangwattana, T. An accelerated Halpern-type algorithm for solving variational inclusion problems with applications. *Bangmod International Journal of Mathematical and Computational Science*, 8, 37–55, (2022). [CrossRef]

[31] Thammasiri, P. and Ungchittrakool, K. Accelerated hybrid Mann-type algorithm for fixed point and variational inequality problems. *Nonlinear Convex Analysis and Optimization: An International Journal on Numerical, Computation and Applications*, 1(1), 97–111, (2022).

[32] Beck, A. and Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1), 183–202, (2009). [CrossRef]

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

[34] Ortega, J.M. and Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*. Society for Industrial and Applied Mathematics: USA, (2020).

[35] Polyak, B.T. *Introduction to Optimization*. Optimization Software Inc: New York, (1987).

[36] Liang, J. *Convergence Rates of First-Order Operator Splitting Methods*. Ph.D. Thesis, Department of Applied Mathematics, The University of Normandie, (2016). [https://hal.science/tel-01388978/]

[37] Combettes, P.L. and Glaudin, L.E. Quasi-nonexpansive iterations on the affine hull of orbits: from Mann's mean value algorithm to inertial methods. *SIAM Journal on Optimization*, 27(4), 2356–2380, (2017). [CrossRef]

[38] Dong, Q.L., Cho, Y.J. and Rassias, T.M. General inertial Mann algorithms and their convergence analysis for nonexpansive mappings. In *Applications of Nonlinear Analysis* (pp.175-191). Springer Optimization and Its Applications, New York, USA: Springer, (2018). [CrossRef]

[39] Dong, Q.L., Huang, J.Z., Li, X.H., Cho, Y.J. and Rassias, T.M. MiKM: multi-step inertial Krasnosel'skiĭ–Mann algorithm and its applications. *Journal of Global Optimization*, 73, 801–824, (2019). [CrossRef]

[40] Malitsky, Y. Golden ratio algorithms for variational inequalities. *Mathematical Programming*,

184, 383-410, (2020). [CrossRef]

[41] Zhang, C. and Chu, Z. New extrapolation projection contraction algorithms based on the golden ratio for pseudo-monotone variational inequalities. *AIMS Mathematics*, 8(10), 23291-23312, (2023). [CrossRef]

[42] Iiduka, H. Acceleration method for convex optimization over the fixed point set of a nonexpansive mapping. *Mathematical Programming*, 149, 131-165, (2015). [CrossRef]

[43] Iiduka, H. Three-term conjugate gradient method for the convex optimization problem over the fixed point set of a nonexpansive mapping. *Applied Mathematics and Computation*, 217(13), 6315-6327, (2011). [CrossRef]

[44] Iiduka, H. and Yamada, I. A use of conjugate gradient direction for the convex optimization problem over the fixed point set of a nonexpansive mapping. *SIAM Journal on Optimization*, 19(4), 1881-1893, (2009). [CrossRef]

[45] Harbau, M.H., Ugwunnadi, G.C., Jolaoso, L.O. and Abdulwahab, A. Inertial accelerated algorithm for fixed point of asymptotically nonexpansive mapping in real uniformly convex Banach spaces. *Axioms*, 10(3), 147, (2021). [CrossRef]

[46] Dong, Q.L. and Yuan, H.B. Accelerated Mann and CQ algorithms for finding a fixed point of a nonexpansive mapping. *Fixed Point Theory and Applications*, 2015, 125, (2015). [CrossRef]

[47] Ahmad, A., Kumam, P. and Harbau, M.H. Convergence theorems for common solutions of nonlinear problems and applications. *Carpathian Journal of Mathematics*, 40(2), 207-241, (2024). [CrossRef]

[48] Kiri, A.I. and Abubakar, A.B. A family of conjugate gradient projection method for nonlinear monotone equations with applications to compressive sensing. *Nonlinear Convex Analysis and Optimization: An International Journal on Numerical, Computation and Applications*, 1(1), 47-65, (2022).

[49] Abubakar, J., Chaipunya, P., Kumam, P. and Salisu, S. A generalized scheme for split inclusion problem with conjugate like direction. *Mathematical Methods of Operations Research*, 101, 51-71, (2025). [CrossRef]

[50] Salihu, N., Kumam, P. and Salisu, S. Two efficient nonlinear conjugate gradient methods for Riemannian manifolds. *Computational and Applied Mathematics*, 43, 415, (2024). [CrossRef]

[51] Yang, Q. On variable-step relaxed projection algorithm for variational inequalities. *Journal of Mathematical Analysis and Applications*, 302(1), 166-179, (2005). [CrossRef]

[52] Che, H., Zhuang, Y., Wang, Y. and Chen, H. A relaxed inertial and viscosity method for split feasibility problem and applications to image recovery. *Journal of Global Optimization*, 87, 619-639, (2023). [CrossRef]

[53] Goebel, K. and Reich, S. *Uniform Convexity, Hyperbolic Geometry, and Nonexpansive Mappings* (Vol. 83). Marcel Dekker: New York, (1984).

[54] Xu, H.K. Iterative methods for the split feasibility problem in infinite-dimensional Hilbert spaces. *Inverse Problems*, 26(10), 105018, (2010). [CrossRef]

[55] Ceng, L.C., Ansari, Q.H. and Yao, J.C. An extragradient method for solving split feasibility and fixed point problems. *Computers & Mathematics with Applications*, 64(4), 633-642, (2012). [CrossRef]

[56] He, S. and Yang, C. Solving the variational inequality problem defined on intersection of finite level sets. *Abstract and Applied Analysis*, 2013(1), 942315, (2013). [CrossRef]

[57] Gibali, A., Thong, D.V. and Vinh, N.T. Three new iterative methods for solving inclusion problems and related problems. *Computational and Applied Mathematics*, 39, 187, (2020). [CrossRef]

[58] Huang, G.B., Zhu, Q.Y. and Siew, C.K. Extreme learning machine: a new learning scheme of feedforward neural networks. In Proceedings, *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, pp. 985-990, Budapest, Hungary, (2004, July). [CrossRef]

[59] Serre, D. *Matrices, Theory and Applications* (Vol. 216). Springer: New York, (2002). [CrossRef]

[60] Ye, H., Cao, F. and Wang, D. A hybrid regularization approach for random vector functional-link networks. *Expert Systems with Applications*, 140, 112912, (2020). [CrossRef]

[61] Wolberg, W., Mangasarian, O., Street, N. and Street, W. Breast Cancer Wisconsin (Diagnostic). *UCI Machine Learning Repository*, (1993). [CrossRef]

[62] Demšar, J. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7, 1-30, (2006).

[63] Shukla, S. *Game Theory for Security Investments in Cyber and Supply Chain Networks*. Ph.D. Thesis, Department of Philosophy, The University of Massachusetts Amherst, (2017). [CrossRef]

[64] Abd-Elhameed, W.M., Youssri, Y.H. and Atta, A.G. Tau algorithm for fractional delay differential equations utilizing seventh-kind Chebyshev polynomials. *Journal of Mathematical Modeling*, 12(2), 277–299, (2024). [CrossRef]

[65] Kirk, D.E. *Optimal Control Theory: An Introduction*. Courier Corporation: USA, (2004).

[66] Youssri, Y.H. and Atta, A.G. Modal spectral Tchebyshev Petrov-Galerkin stratagem for the time-fractional nonlinear Burgers' equation. *Iranian Journal of Numerical Analysis and Optimization*, 14(1), 172–199, (2024).

Mathematical Modelling and Numerical Simulation with Applications (MMNSA)
(https://dergipark.org.tr/en/pub/mmnsa)

How to cite this article: Ahmad, A., Kumam, P. & Seangwattana, T. (2025). Strong convergence multi-step inertial golden ratio-based algorithms for split feasibility problems with applications. *Mathematical Modelling and Numerical Simulation with Applications*, 5(2), 307-347. https://doi.org/10.53391/mmnsa.1595657