Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Projective Splitting With Deviations With Application Torobust Proton Therapy Treatment Planning

Usman Abdulmalik Bello
African Institute of Science and Technology
Halima Yusuf
hyusuf@aust.edu.ng

African Institute of Science and Technology

Research Article

Keywords: Splitting Methods, Monotone Operators, Inclusion Problems, Convex Optimization

Posted Date: June 16th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9939805/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Projective Splitting with Deviations with Application to Robust Proton Therapy Treatment Planning

Abdulmalik Usman Bello[1,2,†] and Halima Yusuf[1,3,*]

[1]Mathematics Institute, African University of Science and Technology, Abuja, Nigeria

[2]Data Science Institute, University of Chicago, Chicago, United States.
ORCID-ID: 0000-0001-6965-8239.

[3]Department of Mathematics, Nigerian Defence Academy, Kaduna, Nigeria.
ORCID-ID: 0009-0002-4838-7200.

[*]Corresponding author: Halima Yusuf, E-mail: hyusuf@aust.edu.ng
Abdulmalik Usman Bello, E-mail: uabdulmalik@aust.edu.ng
[†]These authors contributed equally to this work.

Abstract

Let $\mathcal{H}$ be a real Hilbert space. We prove a weak convergence result for projective splitting method with deviations for a finite sum of monotone inclusion problems. At each iteration, our algorithm incorporates a deviation vector which can be chosen in the same dimension as the underlying space. This provides additional flexibility as well as potential for speeding up the convergence better than the traditional baseline methods. The only condition required for convergence is that the norm of this deviation vector is bounded by a quantity that can be computed online. We apply our result to a robust proton therapy treatment planning problem, where numerical experiments on a head-and-neck cancer patient dataset confirm the theoretical convergence behaviour and demonstrate a significant improvement over the classical inertial algorithm in both convergence speed and stability. Our result extends and complements several existing results in the literature.

Keywords: Splitting Methods; Monotone Operators; Inclusion Problems; Convex Optimization.
2020 Mathematics Subject Classification: 47H05; 49J25; 49M27; 47N10.

1. Introduction

For a real Hilbert space $\mathcal{H}$, consider the following monotone inclusion problem of finding a point $z \in \mathcal{H}$ such that

$$0 \in \sum_{i=1}^{n} T_i(z), \tag{1.1}$$

where $T_i : \mathcal{H} \rightrightarrows \mathcal{H}$, $i = 1, \ldots, n$, are maximal monotone operators.

The monotone inclusion problem (1.1) provides a unified framework for a broad class of problems arising in different areas of nonlinear analysis such as large-scale optimization, variational inequality problems, split feasibility problems and saddle-point problems, see, for example, [8, 25]. For instance, when $n = 2$ in (1.1), $T_1 := N_C$, where N_C denotes the normal cone operator to a nonempty closed convex set $C \subset \mathcal{H}$, and T_2 is single-valued, then, problem (1.1) becomes the variational inequality problem (VIP) studied in [6]. Convex optimization problems, see, e.g.,

[17, 18] can be formulated as special cases of (1.1) through their first-order optimality conditions. Moreover, the fixed point problem considered in [41], can also be naturally recovered from problem (1.1).

Other important problems arising in applications such as compressed sensing, machine learning, reinforcement learning, signal and image processing can be cast as the inclusion (1.1), see, e.g., [2, 5, 22, 38]. In particular, the inclusion in (1.1) is equivalent to the (feasibility) problem of finding a point in an extended solution set

$$\mathcal{S} := \left\{ (z, w_1, \ldots, w_{n-1}) \in \mathcal{H}^n : w_i \in T_i z \ \forall i \in \{1, \ldots, n-1\}, -\sum_{i=1}^{n-1} w_i \in T_n z \right\}, \tag{1.2}$$

of (1.1). We shall use (1.2) in section 3 to formulate the abstract projection splittting algorithm.

Several methods have been proposed for approximating solutions of (1.1) in the case $n = 2$, see, for example, [3, 6, 7, 8, 10, 23, 29, 39]. These include operator splitting schemes such as the forward-backward method [23, 29], Douglas-Rachford method [35], Peaceman-Rachford method [30], and the double-backward splitting method [12]. However, recent advancements have extended the forward-backward method to handle problems involving three to four operators, see, e.g., [10, 11, 31]. The aforementioned splitting methods for the sum of two operators can still be applied for $n > 2$ through product space reformulation, see, e.g., [34]. The product space reformulation, however, is neither computationally nor memory efficient, especially when n is large, which is the case in most applications such as large scale optimizations that arise in machine and reinforcement learning.

A recent approach for solving the inclusion problem (1.1)that does not require product space reformulation is the projective splitting method introduced by Eckstein and Svaiter in [14]. This method belongs to a broader class of separator-projection algorithms for identifying a member of a closed and convex solution set, say, $\mathcal{S}$. At each iteration k, the algorithm constructs an affine function $\varphi_k : \mathcal{H}^n \to \mathbb{R}$ that defines a hyperplane, which separates the current iterate from the extended solution set $\mathcal{S}$, see [13, 14, 21]. The next iterate is then obtained by projecting the current iterate onto this hyperplane, gradually moving towards the solution set. This process generates a sequence of iterates that converges weakly to an element of $\mathcal{S}$.

This approach addresses inclusion problems involving a finite sum of maximal monotone operators (where $n > 2$) without resorting to the product space reformulation. A crucial step in this method involves selecting the points (x_i^k, y_i^k) from the graph of T_i, $i = 1, 2, .., n$, at each iteration k, for the construction of the affine function φ_k.

The authors in [13] and [14] used a proximal step to compute (x_i^k, y_i^k) from the graph of $T_i's$, which involves evaluating the resolvent operator $(I + \rho T_i)^{-1}$, $i = 1, 2, ..., n$, where $\rho > 0$, and I is the identity operator. We remark that computing the resolvents of each operator T_i at each iteration k may be time-consuming and prohibitively expensive, especially for large-scale problems. Moreover, in practice, it is usually not the case that all the resolvents of the operators, $\{T_i\}_{i=1}^n$, can be readily computed.

To address this problem, the authors in [17] incorporate double forward steps for Lipschitz continuous operators while maintaining backward steps for operators in which their resolvent is relatively easy to evaluate, and weak convergence of the algorithm was established. A recent

improvement in [19] reduces the two forward steps to a single forward step under cocoercivity assumption. These advancements offer flexibility in the projective splitting framework and reduce computational cost. For more recent results on this approach, see, [18, 20, 21, 24]. The authors in [26] extended the projective splitting framework by incorporating inertial effects. They developed a projective splitting-type algorithm that has both inertial and relaxation effects, and additionally, with inexact subproblem solutions within a relative-error criterion.

In this work, we propose a novel variant of the projective splitting method that incorporates a deviation vector at each iteration, offering greater algorithmic flexibility and the potential for significant performance gains. To guarantee convergence, we require only that the deviation vector's norm be bounded by a readily computable quantity that can be evaluated online. This approach builds upon and extends previous developments in this direction for splitting methods [4, 32]. A notable advantage of our framework is its compatibility with learning-based strategies: the deviation vector can be selected via neural networks, as it resides in a subspace of the same dimension as the ambient space. This high-dimensional flexibility enables the algorithm to exploit problem-specific structure and learn optimization strategies from data, potentially achieving faster convergence and better robustness in practice. This stands in sharp contrast to inertial acceleration techniques, where the inertial parameter is confined to a one-dimensional subspace, thereby severely limiting the design space for algorithmic enhancements. By lifting this dimensional constraint, our approach opens new avenues for data-driven algorithm design and adaptive optimization strategies.

The rest of the paper is structured as follows: Section 2 presents preliminary lemmas and definitions. Section 3 introduces an abstract projective splitting framework with deviations, alongside results that depend only on the framework's abstract structure. Section 4 presents a general projective splitting framework with deviations and provides the proof of the main convergence result. Section 5 presents numerical experiments, in which we apply our algorithm to a robust proton therapy treatment planning problem on a head-and-neck cancer patient dataset. Finally, Section 6 contains the Conclusion and Further Research, while Section 7 contains statements and declarations.

2. Preliminaries

In this section, we present some basic definitions, lemmas and fundamental assumptions that would be used in the proof of our main results. To that end, throughout this work we denote by $\rightharpoonup$ the weak convergence in $\mathcal{H}$ and $Gr(T)$ the graph of T. Let $\mathcal{H}^n := \underbrace{\mathcal{H} \times \mathcal{H} \times \cdots \times \mathcal{H}}_{n\text{-times}}$ be endowed with the following norm and inner product:

$$\|(z, \mathbf{w})\|^2 := \|z\|^2 + \sum_{i=1}^{n-1} \|w_i\|^2, \text{ where } \mathbf{w} = (w_1, w_2, ..., w_{n-1}) \in \mathcal{H}^{n-1}, z \in \mathcal{H},$$

and

$$\langle (z, \mathbf{w}), (p, \mathbf{q}) \rangle := \langle z, p \rangle + \sum_{i=1}^{n-1} \langle w_i, q_i \rangle,$$

where $z, p \in \mathcal{H}$, $\mathbf{w}, \mathbf{q} \in \mathcal{H}^{n-1}$, respectively. Let $\mathcal{S}$ be the extended solution set (also known as Kuhn-Tucker set) of problem (1.1) defined by

$$\mathcal{S} := \left\{ (z, w_1, \ldots, w_{n-1}) \in \mathcal{H}^n : w_i \in T_i z \ \forall i \in \{1, \ldots, n-1\}, -\sum_{i=1}^{n-1} w_i \in T_n z \right\}, \tag{2.1}$$

it is easy to see that z^* solves (1.1) if and only if there exists $(w_1^*, w_2^*, ..., w_{n-1}^*) \in \mathcal{H}^{n-1}$ such that $(z^*, w_1^*, w_2^*, ..., w_{n-1}^*) \in \mathcal{S}$.

Since each T_i is maximally monotone, then the solution set $\mathcal{S}$ is closed and convex, see, e.g., [17]. In the sequel, we shall assume that $\mathcal{S} \neq \emptyset$.
At iteration k, let $(x_i^k, y_i^k) \in \mathcal{H}^2$, $i = 1, 2, ..., n$, and let $\varphi_k \colon \mathcal{H}^n \to \mathbb{R}$ be defined by

$$\begin{aligned} \varphi_k(z, w_1, ..., w_{n-1}) &:= \sum_{i=1}^{n-1} \langle z - x_i^k, y_i^k - w_i \rangle + \left\langle z - x_n^k, y_n^k + \sum_{i=1}^{n-1} w_i \right\rangle \\ &= \sum_{i=1}^{n} \langle z - x_i^k, y_i^k - w_i \rangle, \end{aligned} \tag{2.2}$$

where $w_n := -\sum_{i=1}^{n-1} w_i$, then φ_k defines a half space $H_k \subseteq \mathcal{H}^n$ which contains the set $\mathcal{S}$ if the points (x_i^k, y_i^k) are appropriately chosen for $i = 1, \ldots, n$. The new iterate, p^{k+1}, is obtained by projecting the previous iterate, p^k, onto the half space, H_k. The key important question in the above process is how to choose the points $(x_i^k, y_i^k) \in \mathcal{H}^2$ so that the function φ_k has the desired properties.

We denote the resolvent of T by $J_\rho^T := (I + \rho T)^{-1}$, where I is the identity map and $\rho > 0$. Clearly,

$$x = J_\rho^T(t) \iff x + \rho y = t, \text{ for some } y \in Tx.$$

Next we give some important properties of the separator map, φ_k, defined in (2.2).

Lemma 2.1. [14] *Let φ_k be defined as in (2.2). Then the following hold:*
1. φ_k is affine on $\mathcal{H}^n$ and

$$\varphi_k(z, w_1, \ldots, w_{n-1}) = \langle z, v^k \rangle + \sum_{i=1}^{n-1} \langle w_i, u_i^k \rangle - \sum_{i=1}^{n} \langle x_i^k, y_i^k \rangle,$$

where $v^k = \sum_{i=1}^n y_i^k$ and $u_i^k = x_i^k - x_n^k$, $i = 1, 2, ..., n-1$.
2. If $y_i^k \in T_i x_i^k$ for all $i = 1, \ldots, n$, then $\varphi_k(z, w_1, \ldots, w_{n-1}) \leq 0$ for all $(z, w_1, \ldots, w_{n-1}) \in \mathcal{S}$.
3. The gradient of φ_k with respect to the inner product $\langle \cdot, \cdot \rangle$ on $\mathcal{H}^n$ is:

$$\nabla \varphi_k = \left(\sum_{i=1}^{n} y_i^k, {x_1}^k - {x_n}^k, {x_2}^k - {x_n}^k, \ldots, {x_{n-1}}^k - {x_n}^k \right). \tag{2.3}$$

Lemma 2.2. *Let $\mathcal{H}$ be real Hilbert space and $x_i \in \mathcal{H}$, $i = 1, \cdots, n$. Then, the following equality holds:*

$$\left\| \sum_{i=1}^{n} x_i \right\|^2 \leq n \sum_{i=1}^{n} \|x_i\|^2$$

Lemma 2.3. *Let $\mathcal{H}$ be real Hilbert space and $x, y \in \mathcal{H}$. Then, the following equalities hold:*
1. $2\langle x, y \rangle = -\|x\|^2 - \|y\|^2 + \|x + y\|^2$.

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

Lemma 2.4. *Let $\mathcal{H}$ be real Hilbert space and $a, b, c, d \in \mathcal{H}$. Then the following inequality holds:*

$$2\langle a - b, c - d \rangle = \|a - d\|^2 + \|b - c\|^2 - \|a - c\|^2 - \|b - d\|^2$$

Lemma 2.5. [1] *Let $\mathcal{H}$ be a real Hilbert space, $A : \mathcal{H} \rightrightarrows \mathcal{H}$ and $B : \mathcal{H}^{n-1} \rightrightarrows \mathcal{H}^{n-1}$ be maximal monotone operators. Let $G : \mathcal{H} \to \mathcal{H}^{n-1}$ be a bounded linear operator. Let $a^k \in A(r^k)$ and $b^k \in B(s^k)$ be such that $r^k \rightharpoonup r^\infty$ and $b^k \rightharpoonup b^\infty$, for some $r^\infty \in \mathcal{H}$ and $b^\infty \in \mathcal{H}^{n-1}$. If $a^k + G^* b^k \to 0$ and $Gr^k - s^k \to 0$, then $b^\infty \in B(Gr^\infty)$ and $-G^* b^\infty \in A(r^\infty)$.*

Lemma 2.6. [27] *Let $\mathcal{H}$ be a real Hilbert space, $\emptyset \neq \mathcal{S} \subseteq \mathcal{H}$. Let $\{p^k\}$ be a sequence in $\mathcal{H}$ such that every weak cluster point of $\{p^k\}$ belongs to $\mathcal{S}$ and $\lim_{k\to\infty} \|p^k - p\|$ exists for every $p \in \mathcal{S}$. Then, $\{p^k\}$ converges weakly to a point in $\mathcal{S}$.*

3. An Abstract Projective Splitting Framework with Deviation

In this section, we first begin by presenting the main abstract projective splitting framework of this paper. This framework is described using an arbitrary affine separator φ_k without specifying how this separator is constructed. The motivation stems from the fact that the monotone inclusion in (1.1) can be reformulated as a problem of finding a point in the extended solution set, $\mathcal{S}$, given in (2.1). In the next section, we introduce Algorithm 2, which provides selection rule for the points (x_i^k, y_i^k) from the graph of T_i, $i = 1, \cdots, n$, in order to explicitly construct the affine separator φ_k.

Algorithm 1: Abstract Projective Splitting with Deviations

Step 0: Let $p^1, v^1 \in \mathcal{H}^n$, $0 < \epsilon < 1$, $0 \leq \xi_k < 1 - \epsilon$, $\epsilon \leq \beta_k \leq 2 - \frac{\epsilon}{2}$. Let $k \leftarrow 1$.

Step 1: Define

$$\hat{p}^k = p^k + v^k \tag{3.1}$$

Step 2: Define an affine function, φ_k, such that $\nabla\varphi_k \neq 0$, and $\varphi_k(p) \leq 0$ for all $p \in \mathcal{S}$. Let $\tilde{p}^{k+1}$ be the projection of $\hat{p}^k$ on to the halfspace $H_k = \{p \in \mathcal{H}^n : \varphi_k(p) \leq 0\}$, i.e.,

$$\tilde{p}^{k+1} = \hat{p}^k - \frac{\max\{0, \varphi_k(\hat{p}^k)\}}{\|\nabla\varphi_k\|^2}\nabla\varphi_k \tag{3.2}$$

Step 3: Choose some relaxation parameter β_k and set

$$p^{k+1} := p^k + \beta_k(\tilde{p}^{k+1} - \hat{p}^k) \tag{3.3}$$

Step 4: Choose v^{k+1} such that

$$\frac{\beta_{k+1}}{2 - \beta_{k+1}}\|v^{k+1}\|^2 \leq \xi_k l_k^2, \quad \text{where} \tag{3.4}$$

$$l_k := \beta_k(2 - \beta_k)\|\tilde{p}^{k+1} - p^k - \frac{(1 - \beta_k)}{2 - \beta_k}v^k\|^2 \tag{3.5}$$

Step 5: Let $k \leftarrow k + 1$ and go to step 1.

We make the following remarks:

Remark 3.1.

(*i*) Observe that using (3.2), the update in step 3 can be re-written as

$$p^{k+1} = p^k - \beta_k \frac{\max\{0, \varphi_k(\hat{p}^k)\}}{\|\nabla\varphi_k\|^2}\nabla\varphi_k.$$

(*ii*) If $v^k = 0$, then Algorithm 1 reduces to a well-known linear separator-projection method for finding a point in the solution set $\mathcal{S}$, see [14].

We now prove the following results, which solely rely on the abstract structure of Algorithm 1 and do not require a particular construction of φ_k.

Lemma 3.2. *Let $\{p^k\}$, $\{v^k\}$, $\{l_k^2\}$ be the sequences generated by Algorithm 1, and let $p^* \in \mathcal{S}$. Then, the following results hold:*

(a) $\|p^{k+1} - p^*\|^2 + l_k^2 \leq \|p^k - p^*\|^2 + \frac{\beta_k}{2-\beta_k}\|v^k\|^2$.

(b) $\|p^{k+1} - p^*\|^2 + l_k^2 \leq \|p^k - p^*\|^2 + \xi_{k-1}l_{k-1}^2$.

(c) The sequence $\{l_k^2\}$ is summable, i.e., $\sum_{k=1}^{+\infty} l_k^2 < \infty$ and $v^k \to 0$.

(d) $\lim_{k\to\infty}\|p^k - p^*\|^2$ *exists.*

(e) The sequences $\{p^k\}, \{\hat{p}^k\}$ and $\{v^k\}$ are bounded.

Proof. (*a*) Let $k \geq 1$, and let $p^* \in \mathcal{S}$. Since $\tilde{p}^{k+1}$ is the projection of $\hat{p}^k$ onto H_k, then it can be written as

$$\tilde{p}^{k+1} = (I + \gamma_k N_{H_k})^{-1}(\hat{p}^k), \tag{3.6}$$

where I denotes the identity map, N_{H_k} denotes the normal cone to H_k, and $\gamma_k > 0$. So, we have

$$\frac{\hat{p}^k - \tilde{p}^{k+1}}{\gamma_k} \in N_{H_k}(\tilde{p}^{k+1}) \tag{3.7}$$

From Step 2 of Algorithm 1, we have that $S \subseteq H_k \ \forall\ k \geq 1$. Since $p^* \in \mathcal{S} \subseteq H_k$, we have $N_{H_k}(p^*) \neq \emptyset$ for each $k \geq 1$. Consequently,

$$0 \in N_{H_k}(p^*) \ \ \forall k \geq 1. \tag{3.8}$$

Monotonicity of N_{H_k}, (3.7) and (3.8) imply

$$\left\langle \frac{\hat{p}^k - \tilde{p}^{k+1}}{\gamma_k}, \tilde{p}^{k+1} - p^* \right\rangle \geq 0. \tag{3.9}$$

Step 3 of Algorithm 1 implies

$$\hat{p}^k - \tilde{p}^{k+1} = \frac{1}{\beta_k}(p^k - p^{k+1}). \tag{3.10}$$

Substituting (3.10) into (3.9), we have

$$\frac{1}{\gamma_k\beta_k}\langle p^k - p^{k+1}, \tilde{p}^{k+1} - p^*\rangle \geq 0.$$

Using Lemma 2.4 and rearranging the resulting inequality gives

$$\|p^{k+1} - p^*\|^2 - \|p^k - p^*\|^2 \leq \|p^{k+1} - \tilde{p}^{k+1}\|^2 - \|p^k - \tilde{p}^{k+1}\|^2. \tag{3.11}$$

Again, Step 3 of Algorithm 1 implies

$$\|p^{k+1} - \tilde{p}^{k+1}\|^2 = \|p^k - \tilde{p}^{k+1}\|^2 + 2\beta_k\langle p^k - \tilde{p}^{k+1}, \tilde{p}^{k+1} - \hat{p}^k\rangle + \beta_k^2\|\tilde{p}^{k+1} - \hat{p}^k\|^2. \tag{3.12}$$

Substituting (3.12) into (3.11), we have

$$\begin{aligned}\|p^{k+1}-p^*\|^2-\|p^k-p^*\|^2 &\leq 2\beta_k\langle p^k-\tilde{p}^{k+1},\tilde{p}^{k+1}-\hat{p}^k\rangle+\beta_k^2\|\tilde{p}^{k+1}-\hat{p}^k\|^2\\ &=\beta_k^2\|\tilde{p}^{k+1}-\hat{p}^k\|^2+2\beta_k\langle\hat{p}^k-\tilde{p}^{k+1},\tilde{p}^{k+1}-\hat{p}^k\rangle\\ &\quad+2\beta_k\langle p^k-\hat{p}^k,\tilde{p}^{k+1}-\hat{p}^k\rangle\\ &=\beta_k^2\|\tilde{p}^{k+1}-\hat{p}^k\|^2-2\beta_k\|\tilde{p}^{k+1}-\hat{p}^k\|^2+2\beta_k\langle p^k-\hat{p}^k,\tilde{p}^{k+1}-\hat{p}^k\rangle\\ &=-\beta_k(2-\beta_k)\|\tilde{p}^{k+1}-\hat{p}^k\|^2+2\beta_k\langle p^k-\hat{p}^k,\tilde{p}^{k+1}-\hat{p}^k\rangle. \qquad (3.13)\end{aligned}$$

Now, using the definition of $\hat{p}^k$ from Step 1 of Algorithm 1, we have

$$\tilde{p}^{k+1}-p^k=\tilde{p}^{k+1}-\hat{p}^k+\hat{p}^k-p^k=\tilde{p}^{k+1}-\hat{p}^k+v^k. \qquad (3.14)$$

Using Step 4 of Algorithm 1 and (3.14), we have

$$\begin{aligned}l_k^2&=\beta_k(2-\beta_k)\|\tilde{p}^{k+1}-\hat{p}^k+v^k-\frac{(1-\beta_k)}{2-\beta_k}v^k\|^2\\ &=\beta_k(2-\beta_k)\|\tilde{p}^{k+1}-\hat{p}^k+\frac{v^k}{2-\beta_k}\|^2\\ &=\beta_k(2-\beta_k)\|\tilde{p}^{k+1}-\hat{p}^k\|^2+\beta_k(2-\beta_k)\|\frac{v^k}{2-\beta_k}\|^2+2\beta_k(2-\beta_k)\left\langle\tilde{p}^{k+1}-\hat{p}^k,\frac{v^k}{2-\beta_k}\right\rangle\\ &=\beta_k(2-\beta_k)\|\tilde{p}^{k+1}-\hat{p}^k\|^2+\frac{\beta_k}{2-\beta_k}\|v^k\|^2+2\beta_k\langle\tilde{p}^{k+1}-\hat{p}^k,v^k\rangle. \qquad (3.15)\end{aligned}$$

Now, adding l_k^2 to both sides of (3.13) and using (3.15), we have

$$\|p^{k+1}-p^*\|^2+l_k^2\leq\|p^k-p^*\|^2+\frac{\beta_k}{2-\beta_k}\|v^k\|^2, \qquad (3.16)$$

which is the desired result in (a).

(b) The result follows by applying (3.4) in Step 4 of Algorithm 1 to (3.16) obtained above.

(c) From (b) we have

$$l_k^2-\xi_{k-1}l_{k-1}^2\leq\|p^k-p^*\|^2-\|p^{k+1}-p^*\|^2. \qquad (3.17)$$

So, for fixed $N\in\mathbb{N}$, we have

$$\sum_{k=1}^{N}l_k^2-\sum_{k=1}^{N}\xi_{k-1}l_{k-1}^2\leq\|p^1-p^*\|^2-\|p^{N+1}-p^*\|^2. \qquad (3.18)$$

But

$$\begin{aligned}\sum_{k=1}^{N} l_k^2 - \sum_{k=1}^{N} \xi_{k-1} l_{k-1}^2 &= \sum_{k=1}^{N} l_k^2 - \sum_{k=0}^{N-1} \xi_k l_k^2 \\ &= \sum_{k=1}^{N} l_k^2 - \sum_{k=1}^{N} \xi_k l_k^2 - \xi_0 l_0^2 + \xi_N l_N^2 \\ &= \sum_{k=1}^{N} l_k^2 (1 - \xi_k) - \xi_0 l_0^2 + \xi_N l_N^2 \\ &\geq \epsilon \sum_{k=1}^{N} l_k^2 - \xi_0 l_0^2 + \xi_N l_N^2. \end{aligned} \tag{3.19}$$

In the last inequality, we have used the assumption that $1 - \xi_k \geq \epsilon$. Using (3.19) together with (3.18), we have

$$\epsilon \sum_{k=1}^{N} l_k^2 \leq \xi_0 l_0^2 + \|p^1 - p^*\|^2 - \|p^{N+1} - p^*\|^2 - \xi_N l_N^2 \leq \xi_0 l_0^2 + \|p^1 - p^*\|^2, \tag{3.20}$$

which implies that $\sum_{k=1}^{\infty} l_k^2 < +\infty$, and consequently, $l_k^2 \to 0$ as $k \to +\infty$. To prove the second part of (c), that is $v^k \to 0$, we first have from the assumption on β_k and ξ_k in Step 0 of Algorithm 1 that

$$\frac{\epsilon}{2-\epsilon} \leq \frac{\beta_k}{2-\beta_k} \text{ and } \xi_k \leq 1 - \epsilon.$$

Now, using Step 4 of Algorithm 1 we have

$$0 \leq \frac{\epsilon}{2-\epsilon} \|v^k\|^2 \leq \frac{\beta_k}{2-\beta_k} \|v^k\|^2 \leq \xi_{k-1} l_{k-1}^2 \leq (1-\epsilon) l_{k-1}^2. \tag{3.21}$$

Since $l_k^2 \to 0$ as $k \to +\infty$, we conclude that $v^k \to 0$ as $k \to +\infty$.

(d) Let $p^* \in \mathcal{S}$. Using the fact that $\xi_k < 1$ in (b), we have

$$\|p^{k+1} - p^*\|^2 + l_k^2 \leq \|p^k - p^*\|^2 + l_{k-1}^2.$$

This implies that the sequence $\{\|p^{k+1} - p^*\|^2 + l_k^2\}$ is non-increasing and bounded below. Therefore, it converges. But from the first part of (c) above we have that $l_k^2 \to 0$ as $k \to +\infty$. Therefore, $\lim_{n\to\infty} \|p^k - p^*\|^2$ exists.

(e) From (c) and (d) we have that the sequences $\{v^k\}$ and $\{p^k\}$ are bounded, respectively. Consequently, $\{\hat{p}^k\}$ is bounded. This completes the proof of the lemma. □

4. A General Projective Splitting Framework with Deviations

We now present Algorithm 2, which gives a detailed description of Algorithm 1, illustrating how the affine separator φ_k is constructed, through the choice of the points (x_i^k, y_i^k) from the graph of $T_i,\ i = 1, \cdots, n$.

Algorithm 2: Projective Splitting with Deviations for Solving (1.1)

Step 0: Let $p^1 = (z^1, w_1^1, \ldots, w_{n-1}^1)$, $v^1 = (v_0^1, v_1^1, v_2^1, \cdots, v_{n-1}^1)$, $0 < \epsilon < 1$, $0 \leq \xi_k < 1 - \epsilon$, $\epsilon \leq \beta_k \leq 2 - \frac{\epsilon}{2}$. Let $k \leftarrow 1$.

Step 1: Define

$$\hat{z}^k = z^k + v_0^k \tag{4.1}$$

$$\hat{w}_i^k = w_i^k + v_i^k,\ i = 1, \cdots, n-1, \tag{4.2}$$

$$\hat{w}_n^k = -\sum_{i=1}^{n-1} \hat{w}_i^k. \tag{4.3}$$

Step 2: Choose scalars $\rho_i^k > 0$ and compute $(x_i^k, y_i^k),\ i = 1, \cdots, n$ satisfying

$$\rho_i^k y_i^k + x_i^k = \hat{z}^k + \rho_i^k \hat{w}_i^k,\ y_i^k \in T_i(x_i^k) \tag{4.4}$$

Step 3: If $x_i^k = x_n^k,\ i = 1, \cdots, n$, and $\sum_{i=1}^n y_i^k = 0$, STOP. Otherwise, define

$$\varphi_k(\hat{p}^k) = \sum_{i=1}^{n} \langle \hat{z}^k - x_i^k, y_i^k - \hat{w}_i^k \rangle \tag{4.5}$$

$$\theta_k = \frac{\max\{0, \varphi_k(\hat{p}^k)\}}{\| \sum_{i=1}^{n} y_i^k \|^2 + \sum_{i=1}^{n-1} \| x_i^k - x_n^k \|^2} \tag{4.6}$$

Step 4: Choose some relaxation parameter $\beta_k \in [\underline{\beta}, \overline{\beta}]$ and define

$$z^{k+1} = z^k - \beta_k \theta_k \sum_{i=1}^{n} y_i^k \tag{4.7}$$

$$w_i^{k+1} = w_i^k - \beta_k \theta_k (x_i^k - x_n^k),\ i = 1, \cdots, n-1. \tag{4.8}$$

Step 5: Choose v^{k+1} such that

$$\frac{\beta_{k+1}}{2 - \beta_{k+1}} \| v^{k+1} \|^2 \leq \xi_k l_k^2, \quad \text{where} \tag{4.9}$$

$$l_k^2 := \beta_k (2 - \beta_k) \left\| \tilde{p}^{k+1} - p^k - \frac{(1 - \beta_k)}{2 - \beta_k} v^k \right\|^2 \tag{4.10}$$

Step 6: Let $k \leftarrow k + 1$ and go to step 1.

Remark 4.1.

(*i*) We remark that Step 2 of Algorithm 2, which specifies how to chose the points (x_i^k, y_i^k) from the graph of T_i, $i = 1, \cdots, n$, is the key distinction between Algorithm 2 and Algorithm 1.

(*ii*) Similar to how $\hat{w}_i^k$ is defined in (4.2), $i = 1, \cdots, n-1$, $\hat{w}_n^k$ can also be written as

$$\hat{w}_n^k = w_n^k + v_n^k,$$

where

$$w_n^k := -\sum_{i=1}^{n-1} w_i^k \quad \text{and} \quad v_n^k := -\sum_{i=1}^{n-1} v_i^k. \tag{4.11}$$

This can be easily seen by substituting (4.2) into (4.3).

(*iii*) If Algorithm 2 terminates at Step 3, then $\nabla\varphi_k = 0$, where $\nabla\varphi_k$ is defined as in (2.3). Moreover, since $y_i^k \in T_i x_i^k$ for $i = 1, \cdots, n$, we take

$$p^{k+1} = (z^{k+1}, w_1^{k+1}, \cdots, w_{n-1}^{k+1}) := (x_n^k, y_1^k, \cdots, y_{n-1}^k) \in \mathcal{S},$$

where $\mathcal{S}$ is as defined in (2.1).

Next, we present some key results used in the proof of convergence of Algorithm 2.

Lemma 4.2. *Let $\{p^k\}$, $\hat{p}^k$ and $\hat{w}_n^k$ be the sequences generated by Algorithm 2. Suppose for $i = 1, \cdots, n$,*

$$0 \leq \underline{\rho} \leq \rho_i^k \leq \overline{\rho} < \infty \ \forall\ k \geq 0.$$

Then,

$$\varphi_k(\hat{p}^k) \geq \frac{1}{2}\min\{\underline{\rho}, \overline{\rho}^{-1}\} \sum_{i=1}^{n} \left(\|y_i^k - \hat{w}_i^k\|^2 + \|x_i^k - \hat{z}^k\|^2\right).$$

Proof. Let $i \in \{1, \cdots, n\}$. Using Lemma 2.3 (1) and equation (4.4) of Algorithm 2, we have

$$\begin{aligned}\left\langle x_i^k - \hat{z}^k, \rho_i^k(y_i^k - \hat{w}_i^k)\right\rangle &= \frac{1}{2}\|x_i^k - \hat{z}^k + \rho_i^k(y_i^k - \hat{w}_i^k)\|^2 - \frac{1}{2}\|\rho_i^k(y_i^k - \hat{w}_i^k)\|^2 - \frac{1}{2}\|x_i^k - \hat{z}^k\|^2 \\ &= -\frac{1}{2}\left((\rho_i^k)^2\|y_i^k - \hat{w}_i^k\|^2 + \|x_i^k - \hat{z}^k\|^2\right). \end{aligned} \tag{4.12}$$

So that,

$$\begin{aligned}\left\langle \hat{z}^k - x_i^k, y_i^k - \hat{w}_i^k\right\rangle &= \frac{1}{2}\left(\rho_i^k\|y_i^k - \hat{w}_i^k\|^2 + (\rho_i^k)^{-1}\|x_i^k - \hat{z}^k\|^2\right) \\ &\geq \frac{1}{2}\left(\underline{\rho}\|y_i^k - \hat{w}_i^k\|^2 + (\overline{\rho})^{-1}\|x_i^k - \hat{z}^k\|^2\right) \\ &\geq \frac{1}{2}\min\{\underline{\rho}, \overline{\rho}^{-1}\}\left(\|y_i^k - \hat{w}_i^k\|^2 + \|x_i^k - \hat{z}^k\|^2\right). \end{aligned} \tag{4.13}$$

Summing up both sides of (4.13) from $i = 1, \cdots, n$, we have

$$\varphi_k(\hat{p}^k) \geq \frac{1}{2}\min\{\underline{\rho}, \overline{\rho}^{-1}\} \sum_{i=1}^{n} \left(\|y_i^k - \hat{w}_i^k\|^2 + \|x_i^k - \hat{z}^k\|^2\right).$$

□

Lemma 4.3. *Let $\{p^k\}$, $\hat{p}^k$ and $\hat{w}_n^k$ be the sequences generated by Algorithm 2. Then, the following inequality holds:*

$$\frac{(\varphi_k(\hat{p}^k))^2}{\alpha\|\nabla\varphi_k\|^2} \geq \varphi_k(\hat{p}^k) \geq \alpha\|\nabla\varphi_k\|^2, \tag{4.14}$$

for some $\alpha > 0$.

Proof. Let $k \geq 1$. Then,

$$\begin{aligned}
\left\|\sum_{i=1}^{n} y_i^k\right\|^2 &= \left\|\sum_{i=1}^{n}(\hat{w}_i^k - y_i^k - \hat{w}_i^k)\right\|^2 \\
&= \left\|\sum_{i=1}^{n}(\hat{w}_i^k - y_i^k)\right\|^2 \\
&\leq n\sum_{i=1}^{n}\|\hat{w}_i^k - y_i^k\|^2 \\
&\leq 2n\sum_{i=1}^{n}\|\hat{w}_i^k - y_i^k\|^2,
\end{aligned} \tag{4.15}$$

where we have used (4.3) in the second equality, and Lemma 2.2 in the first inequality. On the other hand, using Cauchy-Schwarz inequality, we have

$$\begin{aligned}
\sum_{i=1}^{n-1}\|x_i^k - x_n^k\|^2 &\leq 2\sum_{i=1}^{n-1}\|x_i^k - \hat{z}^k\|^2 + 2\sum_{i=1}^{n-1}\|\hat{z}^k - x_n^k\|^2 \\
&\leq 2\sum_{i=1}^{n-1}\|x_i^k - \hat{z}^k\|^2 + 2(n-1)\|\hat{z}^k - x_n^k\|^2 \\
&\leq 2n\sum_{i=1}^{n}\|x_i^k - \hat{z}^k\|^2.
\end{aligned} \tag{4.16}$$

Using (2.3), (4.15) and (4.16), we have

$$\begin{aligned}
\|\nabla\varphi_k\|^2 &= \left\|\sum_{i=1}^{n} y_i^k\right\|^2 + \sum_{i=1}^{n-1}\|x_i^k - x_n^k\|^2 \\
&\leq 2n\sum_{i=1}^{n}\left(\|\hat{w}_i^k - y_i^k\|^2 + \|x_i^k - \hat{z}^k\|^2\right).
\end{aligned} \tag{4.17}$$

Equation (4.17) implies

$$\sum_{i=1}^{n}\left(\|\hat{w}_i^k - y_i^k\|^2 + \|x_i^k - \hat{z}^k\|^2\right) \geq \frac{1}{2n}\|\nabla\varphi_k\|^2. \tag{4.18}$$

Lemma 4.2 together with (4.18) yields

$$\begin{aligned}\varphi_k(\hat{p}^k) &\geq \frac{1}{2}\min\{\underline{\rho}, \overline{\rho}^{-1}\}\sum_{i=1}^{n}\left(\|y_i^k - \hat{w}_i^k\|^2 + \|x_i^k - \hat{z}^k\|^2\right)\\ &\geq \frac{1}{2}\min\{\underline{\rho}, \overline{\rho}^{-1}\}\frac{1}{2n}\|\nabla\varphi_k\|^2\\ &= \frac{\min\{\underline{\rho}, \overline{\rho}^{-1}\}}{4n}\|\nabla\varphi_k\|^2. \end{aligned} \tag{4.19}$$

Define $\alpha := \frac{\min\{\underline{\rho}, \overline{\rho}^{-1}\}}{4n}$. Then, (4.19) becomes $\varphi_k(\hat{p}^k) \geq \alpha\|\nabla\varphi_k\|^2$, which is the second part of the inequality in (4.14). The first part of the inequality is obtained from the second. □

Lemma 4.4. *Let $\{p^k\}$, $\hat{p}^k$ and $\hat{w}_n^k$ be the sequences generated by Algorithm 2. Then, the following result holds:*

$$\frac{\max\{0, \varphi_k(\hat{p}^k)\}}{\|\nabla\varphi_k\|} \to 0.$$

Proof. We first observe that Lemma 3.2 (d) implies that the sequence $\{p^k - p^*\}$ is Cauchy. In particular, $\|p^{k+1} - p^k\| \to 0$. Also, using Step 3 of Algorithm 1 together with the lower bound on β_k given in Algorithm 1, we have

$$0 \leq \epsilon\|\tilde{p}^{k+1} - \hat{p}^k\| \leq \beta_k\|\tilde{p}^{k+1} - \hat{p}^k\| = \|p^{k+1} - p^k\| \to 0,$$

which implies that $\|\tilde{p}^{k+1} - \hat{p}^k\| \to 0$. Consequently, the result follows from Step 2 of Algorithm 1. □

Lemma 4.5. *Let $\{p^k\}$, $\hat{p}^k$ and $\hat{w}_n^k$ be the sequences generated by Algorithm 2. The following results hold:*

(a) $\varphi_k(\hat{p}^k) \to 0$ and $\|\nabla\varphi_k\|^2 \to 0$.

(b) $\|\hat{z}^k - x_i^k\|^2 \to 0$ and $\|y_i^k - \hat{w}_i^k\|^2 \to 0$, $i \in \{1, \cdots, n\}$.

(c) $\|z^k - x_i^k\|^2 \to 0$ and $\|y_i^k - w_i^k\|^2 \to 0$, $i \in \{1, \cdots, n\}$.

(d) Every weak cluster point of $\{p^k\}$ belongs to $\mathcal{S}$.

(e) If every weak cluster point of $\{p^k\}$ belongs to $\mathcal{S}$, then the sequence, $\{p^k\}$, converges weakly to some element in $\mathcal{S}$.

Proof. (a) Since $\varphi_k(\hat{p}^k) \geq 0$ from Lemma 4.2, then by Lemma 4.4, we obtain $\frac{\varphi_k(\hat{p}^k)}{\|\nabla\varphi_k\|} \to 0$. Consequently, Lemma 4.3 yields

$$0 \leq \varphi_k(\hat{p}^k) \leq \frac{(\varphi_k(\hat{p}^k))^2}{\alpha\|\nabla\varphi_k\|^2} \to 0.$$

Thus, $\varphi_k(\hat{p}^k) \to 0$. To obtain the second part of (a), we use Lemma 4.3 and the first part of (a) to obtain

$$0 \leq \alpha\|\nabla\varphi_k\|^2 \leq \varphi_k(\hat{p}^k) \to 0.$$

This implies that $\|\nabla\varphi_k\|^2 \to 0$. That is $\|\sum_{i=1}^{n} y_i^k\|^2$ and $\sum_{i=1}^{n-1}\|x_i^k - x_n^k\|^2$ both go to zero as k tends to infinity.

(b) This result follows directly from Lemma 4.2 and the first part of Lemma 4.5 (a).

(c) Let $i \in \{1, \cdots, n\}$. We observe that

$$\|z^k - x_i^k\| \leq \|v^k\| + \|\hat{z}^k - x_i^k\|. \tag{4.20}$$

Using the result in the second part of Lemma 3.2 (c) and the first part of Lemma 4.5 (b) in (4.20), we have the first part of the result. Similarly, since

$$\|y_i^k - w_i^k\| \leq \|y_i^k - \hat{w}_i^k\| + \|v^k\|, \tag{4.21}$$

using the result in the second part of Lemma 3.2 (c) and the second part of Lemma 4.5 (b) in (4.21), we have the second part of the result.

(d) Observe that $\{p^k\}$ is bounded by Lemma 3.2 (e). Let $\{p^{k_j}\}$ be a subsequence of $\{p^k\}$ such that $p^{k_j} \rightharpoonup p^\infty$, i.e.,

$$z^{k_j} \rightharpoonup z^\infty \text{ and } w_i^{k_j} \rightharpoonup w_i^\infty, \ i = 1, \cdots, n. \tag{4.22}$$

Lemma 4.5 (c) together with (4.22) imply

$$x_i^{k_j} \rightharpoonup z^\infty \text{ and } y_i^{k_j} \rightharpoonup w_i^\infty, \ i \in \{1, \cdots, n\}. \tag{4.23}$$

Now, define

$$A : \mathcal{H} \rightrightarrows \mathcal{H}, \quad B : \mathcal{H}^{n-1} \rightrightarrows \mathcal{H}^{n-1} \text{ and } G : \mathcal{H} \to \mathcal{H}^{n-1}$$

by

$$A := T_n, \quad B := T_1 \times \cdots \times T_{n-1} \quad \text{and } G := (G_1, \cdots, G_{n-1}), \tag{4.24}$$

where $G_i : \mathcal{H} \to \mathcal{H}$ and $G_i = I$ for all $i \in \{1, \cdots, n-1\}$, I denotes the identity map. The adjoint of G is $G^* : \mathcal{H}^{n-1} \to \mathcal{H}$ such that for any $y = (y_1, .., y_{n-1}) \in \mathcal{H}^{n-1}, x \in \mathcal{H}$, it satisfies

$$\langle G^* y, x\rangle = \langle y, Gx\rangle,$$

which as a result, gives us that $G^* y = \sum_{i=1}^{n-1} y_i$. Now, for any $i \in \{1, \cdots, n\}$, we have, from Step 2 of Algorithm 2 that $y_i^k \in T_i(x_i^k)$. This, in particular, implies that

$$y_n^{k_j} \in T_n x_n^{k_j}.$$

Again, using the notation of B in (4.24), we have that, for any $i \in \{1, \cdots, n\}$,

$$y_i^k \in T_i(x_i^k) \Longrightarrow (y_1^{k_j}, y_2^{k_j}, \cdots, \cdots y_n^{k_j}) \in B(x_1^{k_j}, \cdots, x_n^{k_j}).$$

Moreover, equation (4.23) implies that

$$x_n^{k_j} \rightharpoonup z^\infty \text{ and } (y_1^{k_j}, y_2^{k_j}, \cdots y_{n-1}^{k_j}) \rightharpoonup (w_1^\infty, w_2^\infty, \cdots, w_{n-1}^\infty).$$

Using the consequence of the second part of Lemma 4.5 (a) and the fact that $G^* y = \sum_{i=1}^n y_i$, we have

$$y_n^{k_j} + G^*(y_1^{k_j}, y_2^{k_j}, \cdots y_{n-1}^{k_j}) = y_n^{k_j} + \sum_{i=1}^{n-1} y_i^{k_j} = \sum_{i=1}^{n} y_i^{k_j} \to 0.$$

Again, using the consequence of the second part of Lemma 4.5 (a), we have

$$Gx_n^{k_j} - (x_1^{k_j}, x_2^{k_j}, \cdots, x_{n-1}^{k_j}) \to \mathbf{0}.$$

Now, take

$$r^{k_j} := x_n^{k_j} \text{ and } r^\infty := z^\infty \tag{4.25}$$

$$b^{k_j} := (y_1^{k_j}, \cdots, y_{n-1}^{k_j}) \text{ and } b^\infty := (w_1^\infty, w_2^\infty, \cdots, w_{n-1}^\infty) \tag{4.26}$$

$$a^{k_j} := y_n^{k_j} \text{ and } s^{k_j} := (x_1^{k_j}, \cdots, x_{n-1}^{k_j}). \tag{4.27}$$

Then, using Lemma 2.5, we conclude that

$$b^\infty \in B(Gr^\infty) \quad \text{and} \quad -G^*b^\infty \in Ar^\infty. \tag{4.28}$$

Using the notations in (4.25)-(4.27) together with those in (4.24), (4.28), we have

$$w_i^\infty \in T_i(z^\infty),\ i = 1, \cdots, n-1 \quad \text{and} \quad -\sum_{i=1}^{n-1} w_i^\infty \in T_n z^\infty,$$

which implies that $p^\infty = (z^\infty, w_1^\infty, \cdots, w_{n-1}^\infty) \in \mathcal{S}$.

(e). In view of Lemma 4.5 (d) and Lemma 3.2 (d), the result follows directly from Lemma 2.6. □

Theorem 4.6. *Let $\{p^k\} = \{(z^k, w_1^k, \cdots, w_{n-1}^k)\}$ be a sequence generated by Algorithm 2. If Algorithm 2 terminates via Step 3, then its final iterate is a member of the extended solution set $\mathcal{S}$. Otherwise, if Algorithm 2 produces an infinite sequence of iterations without terminating via Step 3, then there exists $\bar{p} = (\bar{z}, \bar{w}_1, \cdots, \bar{w}_{n-1}) \in \mathcal{S}$, such that $z^k \rightharpoonup \bar{z}$ and $y_i^k \rightharpoonup \bar{w}_i,\ i = 1, \cdots, n$. Furthermore, $x_i^k \rightharpoonup \bar{z}$ and $y_i^k \rightharpoonup \bar{w}_i,\ i = 1, \cdots, n$, where $w_n^k = -\sum_{i=1}^{n-1} w_i^k$.*

Proof. Let $\{p^k\}$ be a sequence generated by Algorithm 2. We consider the case where Algorithm 2 runs indefinitely and does not terminate at Step 3, since the case of termination at Step 3 has already been addressed in Remark 4.1. The fact that $\{p^k\}$ converges to some $\bar{p} = (\bar{z}, \bar{w}_1, \cdots, \bar{w}_{n-1}) \in \mathcal{S}$ follows from Lemma 4.5 (d) and (e). Consequently, Lemma 4.5 (c) gives

$$x_i^k \rightharpoonup \bar{z} \text{ and } y_i^k \rightharpoonup \bar{w}_i,\ i = 1, \cdots, n-1.$$

□

We now present a special case of Algorithm 2 in which the deviation vector is chosen as the inertial term $v^k = t_k(p^k - p^{k-1})$, where the inertial parameter t_k is adaptively controlled via the condition in Step 5 of the Algorithm below:

Algorithm 3: Projective Splitting with Controlled Inertia for Solving (1.1)

Step 0: Let $p^1 = (z^1, w_1^1, \ldots, w_{n-1}^1)$, $v^1 = (v_0^1, v_1^1, v_2^1, \cdots, v_{n-1}^1)$, $0 < \epsilon < 1$, $0 \leq \xi_k < 1 - \epsilon$, $\epsilon \leq \beta_k \leq 2 - \frac{\epsilon}{2}$. Let $t_k \in [0, 1]$ and $k \leftarrow 1$.

Step 1: Define

$$\hat{z}^k = z^k + t_k(z^k - z^{k-1}) \tag{4.29}$$

$$\hat{w}_i^k = w_i^k + t_k(w_i^k - w_i^{k-1}),\ i = 1, \cdots, n-1, \tag{4.30}$$

$$\hat{w}_n^k = -\sum_{i=1}^{n-1} \hat{w}_i^k. \tag{4.31}$$

Step 2: Choose scalars $\rho_i^k > 0$ and compute $(x_i^k, y_i^k),\ i = 1, \cdots, n$ satisfying

$$\rho_i^k y_i^k + x_i^k = \hat{z}^k + \rho_i^k \hat{w}_i^k,\ y_i^k \in T_i(x_i^k) \tag{4.32}$$

Step 3: If $x_i^k = x_n^k,\ i = 1, \cdots, n$, and $\sum_{i=1}^n y_i^k = 0$, STOP. Otherwise, define

$$\varphi_k(\hat{p}^k) = \sum_{i=1}^{n} \langle \hat{z}^k - x_i^k, y_i^k - \hat{w}_i^k \rangle \tag{4.33}$$

$$\theta_k = \frac{\max\{0, \varphi_k(\hat{p}^k)\}}{\| \sum_{i=1}^{n} y_i^k \|^2 + \sum_{i=1}^{n-1} \| x_i^k - x_n^k \|^2} \tag{4.34}$$

Step 4: Chose some relaxation parameter $\beta_k \in [\underline{\beta}, \overline{\beta}]$ and define

$$z^{k+1} = z^k - \beta_k \theta_k \sum_{i=1}^{n} y_i^k \tag{4.35}$$

$$w_i^{k+1} = w_i^k - \beta_k \theta_k (x_i^k - x_n^k),\ i = 1, \cdots, n-1. \tag{4.36}$$

Step 5: Choose t_{k+1} such that

$$t_{k+1}^2 \| p^{k+1} - p^k \|^2 \leq \frac{2 - \beta_{k+1}}{\beta_{k+1}} \xi_k l_k^2 \ \text{ where} \tag{4.37}$$

$$l_k^2 := \beta_k (2 - \beta_k) \left\| \tilde{p}^{k+1} - p^k - \frac{(1 - \beta_k)}{2 - \beta_k} t_k (p^k - p^{k-1}) \right\|^2 \tag{4.38}$$

Step 6: Let $k \leftarrow k + 1$ and go to step 1.

Therefore, the following corollary follows.

Corollary 1. *Let $\{p^k\} = \{(z^k, w_1^k, \cdots, w_{n-1}^k)\}$ be a sequence generated by Algorithm 3 with deviation vector $v^k = t_k(p^k - p^{k-1})$, where t_k satisfies the condition in Step 5 of Algorithm 3. If Algorithm 3 terminates via Step 3, then its final iterate is a member of the extended solution set $\mathcal{S}$. Otherwise, if Algorithm 3 produces an infinite sequence without terminating via Step 3, then there exists $\bar{p} = (\bar{z}, \bar{w}_1, \cdots, \bar{w}_{n-1}) \in \mathcal{S}$ such that $z^k \rightharpoonup \bar{z}$ and $y_i^k \rightharpoonup \bar{w}_i$, $i = 1, \cdots, n$. Furthermore, $x_i^k \rightharpoonup \bar{z}$ and $y_i^k \rightharpoonup \bar{w}_i$, $i = 1, \cdots, n$, where $\bar{w}_n^k = -\sum_{i=1}^{n-1} \bar{w}_i^k$.*

5. Numerical Experiments

In this section, we apply our result to robust proton therapy treatment planning problem. We evaluate the performance of our algorithm on a head-and-neck patient dataset, considering 13 uncertainty scenarios that account for $3mm$ setup and 3.5% range uncertainties under different beam configurations. We demonstrate how our Algorithm can be utilized for this class of problems to determine optimal beam intensities for an effective treatment plan.

Although our main result is formulated in terms of deviation, Algorithm 3, as a special case, reduces to an inertial-type algorithm, but with a notable distinction. In classical inertial algorithms, the inertial parameter is chosen within the interval $[0, 1)$. In contrast, for Algorithm 3, the inertial parameter must satisfy the specific condition outlined in Step 5 of the Algorithm. This additional requirement differentiates Algorithm 3 from classical inertial methods.

All numerical experiments were carried out on MATLAB R2020a version. All programs were run on a 64-bit OS PC with Intel(R) Core(TM) i7-7600U CPU @ 2.80GHz 2.90 GHz and 16GB RAM. All figures were plotted using the log log plot command.

Example 5.1. Robust Proton Treatment Planning Problem.
Proton treatment planning has been widely studied for many years, see, e.g., [15, 16, 33, 36, 40]. The sharp dose fall-off of protons allows precise tumor targeting while sparing surrounding healthy tissues. However, this same advantage makes proton therapy sensitive to uncertainties such as patient setup errors, anatomical changes, and proton range estimation inaccuracies. Consequently, the proton treatment planning problem is formulated as a robust optimization problem, which takes the form of problem (1.1), and the plan is optimized to remain effective across all possible scenarios [40, 42].

Mathematically, we consider r uncertainty scenarios, including the nominal scenario, labeled $s = 1, \ldots, r$. Each scenario has a dose influence matrix $A_s \in \mathbb{R}_+^{m\times n}$, where m is the number of voxels and n is the number of beamlets. Let $a^{si} := (a_{i1}^{si}, a_{i2}^{si}, a_{i3}^{si}, ..., a_{in}^{si}) \in \mathbb{R}_+^n$ be row i of A_s, $i = 1, \ldots, m$. Then $A_s = [a^{s1}, a^{s2}, \ldots, a^{sm}]^\top$. Each voxel has a prescribed dose, which we collect in a vector $p \in \mathbb{R}_+^m$. For voxels belonging to the target, i.e., the tumor, the prescribed dose is a fixed constant $D_{\text{pres}} > 0$, while voxels outside the target have a prescribed dose of 0.

Our goal is to determine spot intensities $z \in \mathbb{R}_+^n$ that minimize the total deviation between the prescribed dose and the dose delivered in all uncertainty scenarios. This leads to the following constrained least squares optimization problem:

$$\min_{z\in\mathbb{R}_+^n} \sum_{s=1}^{r} \|A_s z - p\|^2 \;=\; \min_{z\in\mathbb{R}_+^n} \sum_{s=1}^{r}\sum_{i=1}^{m} ({a^{si}}^\top z - p_i)^2 \;=\; \min_{z\in\mathbb{R}_+^n} \sum_{s=1}^{r} f_s(z), \tag{5.1}$$

where $f_s(z) := \sum_{i=1}^{m} ({a^{si}}^\top z - p_i)^2$. To determine the optimal spot intensities for problem (5.1) using Algorithm 3, we first observe that problem (5.1) can be expressed as the following unconstrained optimization problem

$$\min_{z\in\mathbb{R}_+^n} \sum_{s=1}^{r} \|A_s z - p\|^2 = \min_{z\in\mathbb{R}^n} \sum_{s=1}^{r} \|A_s z - p\|^2 + \mathbb{I}_{\mathbb{R}_+^n}(z), \tag{5.2}$$

where $\mathbb{I}_{\mathbb{R}^n_+}$ is the indicator function of the positive orthant $\mathbb{R}^n_+$. The first-order optimality condition of (5.2) gives the following monotone inclusion problem :

$$0 \in \sum_{s=1}^{r} 2A_s^\top (A_s z - p) \; + N_{\mathbb{R}^n_+}(z), \tag{5.3}$$

which fits the general framework (1.1). Here, $N_{\mathbb{R}^n_+}$ denotes the normal cone to the non-negative orthant $\mathbb{R}^n_+$, which is known to be the subdifferential of the indicator function $\mathbb{I}_{\mathbb{R}^n_+}(z)$. Comparing (5.3) with our problem of interest (1.1), we have that, $n = r + 1$ and

$$T_s(z) = 2A_s^\top (A_s z - p), \;\; s = 1, \cdots, r, \qquad \text{and} \qquad T_{r+1}(z) = N_{\mathbb{R}^n_+}.$$

We consider three influence matrices, whose dimensions and sparsity are reported in Table 1. The first two are obtained from the CORT dataset [9] and correspond to a single head-and-neck cancer patient. Each of these two dataset corresponds to a different beam configuration. The third is a randomly generated matrix of comparable size.

TABLE 1. Dimensions and sparsity of influence matrices used in the experiments

Dataset	**Beam configuration**	m **(voxels)**	n **(beamlets)**	**Sparsity**
Clinical 1	Gantry 138°, Couch 0°	1,715,200	1,168	0.31%
Clinical 2	Gantry 135°, Couch 45°	1,715,200	1,083	0.33%
Synthetic	Randomly generated (Matlab)	1,715,200	1,083	2%

We simulate proton-like uncertainty scenarios following the framework of [28, 37]. Specifically, range uncertainty was approximated by rescaling the dose influence matrix by 3.5%, analogous to the stopping power ratio uncertainty in proton therapy. The setup errors were simulated by shifting the isocenter by ±3 mm in the x, y, and z directions. This yields 13 scenarios: one nominal and twelve perturbed scenarios. For clarity, we refer to Algorithm 3 of this paper, in which the inertial parameter t_k is chosen to satisfy the condition in Step 5, as Algorithm A. We also refer to the classical inertial version of Algorithm 3, in which the inertial parameter is chosen freely in [0,1) without satisfying the condition in Step 5, as Algorithm B. The results in Figures 1–6 demonstrate a potential improvement in the convergence and stability of Algorithm A over Algorithm B.

Both algorithms are implemented using the same relaxation parameter $\beta_k = \beta = 1.5519$. For Algorithm A, we set $\epsilon = 0.05\beta$, $\xi_k = 0.051(1 - \epsilon)$, and adaptively select the inertial parameter t_k at each iteration to satisfy the safeguarding condition. For Algorithm B, we fix $t_k = t = 0.1$, as this value appears to give the best performance.

For each of the three datasets described in Table 1, i.e., Clinical dataset 1, Clinical dataset 2, and the synthetic dataset, the convergence graphs are presented in Figures 1–2, Figures 3–4, and Figures 5–6, respectively.

6. Conclusion and Further Research

In this work, we have proved weak convergence of projective splitting methods with deviations for finding zeros of a finite sum of maximal monotone operators. The Algorithm allows the introduction of a deviation vector at each iteration, thereby providing additional flexibility in the update steps while preserving the fundamental structure of the projective splitting framework. We proved that the generated sequence converges weakly to a solution of the inclusion

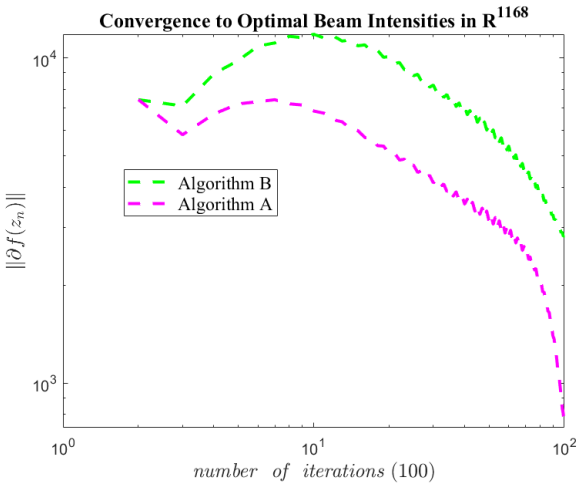


Figure 1. Example (5.1) In $\mathbb{R}^{1168}$. The figure corresponds to Clinical dataset 1 (Gantry 138°, Couch 0°) of dimension 1,715,200 × 1,168 from which 13 scenarios are generated. Both Algorithms exhibit an initial transient rise before decreasing, with Algorithm A maintaining a consistently lower subgradient norm than Algorithm B throughout, reaching a significantly lower final value by iteration 100

.

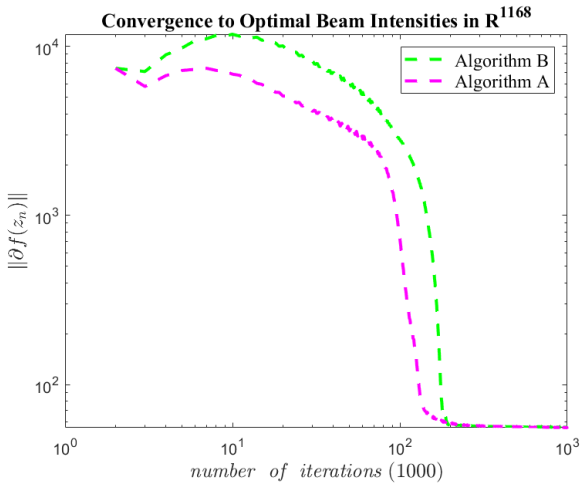


Figure 2. Example (5.1) in $\mathbb{R}^{1168}$. The figure corresponds to Clinical dataset 1 (Gantry 138°, Couch 0°) with a dose influence matrix of dimension 1,715,200 × 1,168 from which 13 scenarios are generated. Algorithm B peaks at a substantially higher value than Algorithm A during the initial transient phase.

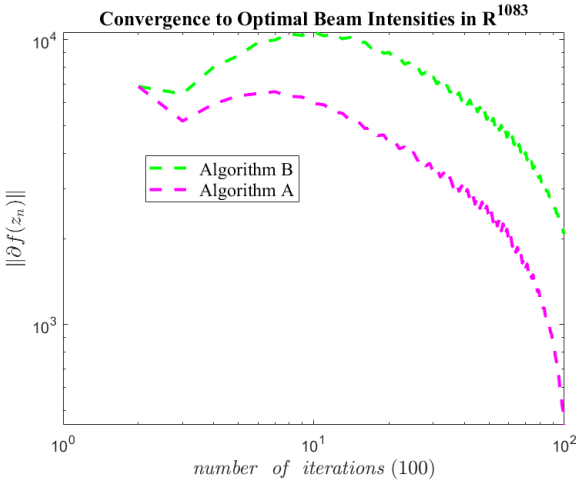


Figure 3. Example (5.1) in $\mathbb{R}^{1083}$. The figure corresponds to Clinical dataset 2 (Gantry 135°, Couch 45°) with a dose influence matrix of dimension 1,715,200 × 1,083, from which 13 scenarios are generated. Algorithm A consistently maintains a lower subgradient norm than Algorithm B across all iterations, reaching approximately ten times lower final value than Algorithm B by iteration 100.

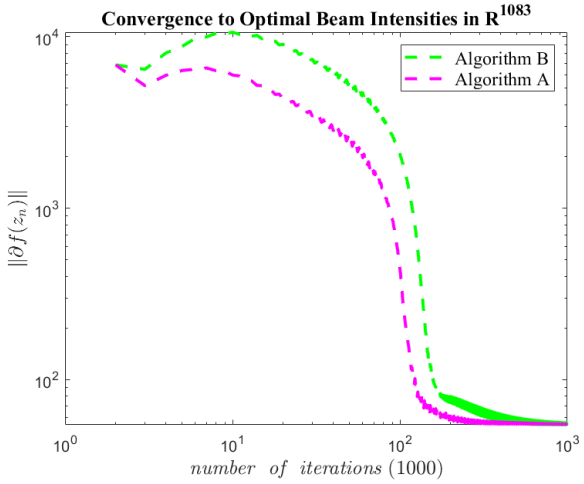


Figure 4. Example (5.1) in $\mathbb{R}^{1083}$. The figure corresponds to Clinical dataset 2 (Gantry 135°, Couch 45°) with a dose influence matrix of dimension 1,715,200 × 1,083, from which 13 scenarios are generated. Algorithm A converges faster than Algorithm B, reaching the minimum subgradient norm approximately 100 iterations earlier. Both Algorithms eventually stabilize at comparable final values.

problem, provided that the norm of the deviation vector is bounded by a quantity that can be

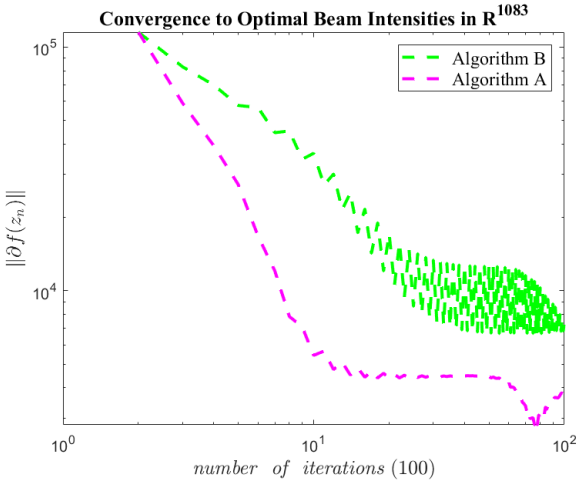


Figure 5. Example (5.1) in $\mathbb{R}^{1083}$. The figure corresponds to the synthetic dataset of dimension $1{,}715{,}200 \times 1{,}083$, from which 13 scenarios are generated. Algorithm B exhibits persistent oscillations over approximately iterations 10 to 100, failing to stabilize within the given number of iterations. Algorithm A, however, decreases smoothly with only a single transient dip before recovering and continuing its stable convergence to a significantly lower value.

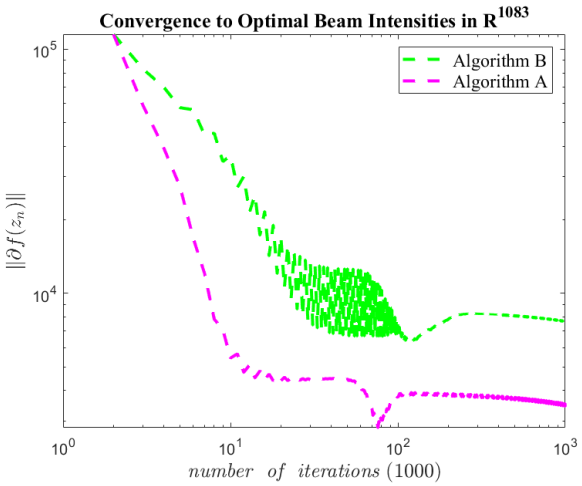


Figure 6. Example (5.1) in $\mathbb{R}^{1083}$. This figure corresponds to the synthetic dataset of dimension $1{,}715{,}200 \times 1{,}083$, from which 13 scenarios are generated. Algorithm B oscillates persistently over approximately iterations 10 to 300 before eventually stabilising, but at a substantially higher subgradient norm than Algorithm A. Algorithm A, by contrast, decreases smoothly with only a single transient dip near iteration 100, before recovering and continuing its stable convergence to a significantly lower final value.

computed online. We applied Algorithm 3 to a robust proton therapy treatment planning problem. Numerical experiments conducted on a head-and-neck cancer patient dataset confirmed the theoretical convergence behavior of the algorithm. Moreover, the results demonstrated a significant improvement of Algorithm 3 over the classical inertial algorithm, both in terms of convergence speed and stability.

Possible future work will concentrate on the use of some neural networks to select deviation vectors for solving large scale problems in inverse problems, see, e.g., [4].

7. Statements and Declarations

Acknowledgments
The authors appreciate the support of the Charles Chidume Mathematics Institute, African University of Science and Technology, Abuja, Nigeria.

Funding
The first author's research was supported by the Schmidt Sciences AI in Science Faculty Fellowship, Data Science Institute, University of Chicago, USA.

The second author's research was supported by the International Mathematical Union Breakout Graduate Fellowship (IMU-BGF) [Award No.: IMU-BGF-2024-46].

Part of this research was performed while the first author was visiting the Institute for Mathematical and Statistical Innovation (IMSI), which is supported by the National Science Foundation (Grant No. DMS-2425650)

Authors' Contributions
All authors contributed equally.

Competing interests
The authors declare that they have no conflict of interest.

Data Availability
The clinical dose-influence matrices used in this study are derived from the publicly available CORT dataset [9], deposited in the GigaScience GigaDB repository and accessible via https://doi.org/10.1186/2047-217X-3-37. The synthetic dose-influence matrix was randomly generated in MATLAB.

Code availability: The matlab code that generated the plots is available upon request.

References

[1] Alotaibi, A., Combettes, P. L., Shahzad, N.: Solving coupled composite monotone inclusions by successive Fejér approximations of their Kuhn-Tucker set. SIAM J. Optim. 24(4), 2076-2095 (2014). https://doi.org/10.1137/130950616

[2] Aragón-Artacho, F. J., Malitsky, Y., Tam, M. K., and Torregrosa-Belén, D.: Distributed forward-backward methods for ring networks. Comput. Optim. Appl. 86(3), 845-870 (2023). https://doi.org/10.1007/s10589-022-00400-z

[3] Attouch, H., Peypouquet, J., Redont, P.: Backward-forward algorithms for structured monotone inclusions in Hilbert spaces. J. Math. Anal. Appl. 457, 1095-1117 (2018). https://doi.org/10.1016/j.jmaa.2016.06.025

[4] Banert S., Rudzusika J., Oktem O., Adler J.: Accelerated Forward-Backward Optimization using Deep Learning. SIAM J. Optim. 34(2), 1236–1263 (2024). https://doi.org/10.1137/22M1532548.

[5] Bello, A. U., Okeke, C. C., Isyaku, M., Omojola, M. T.: Forward-reflected-backward splitting method without cocoercivity for the sum of maximal monotone operators in Banach spaces. Optimization, 72(9), 2201–2222 (2022). https://doi.org/10.1080/02331934.2022.2057308

[6] Bello, A. U., Yusuf, H., Djitte, N.: Single-step algorithm for variational inequality problems in 2-uniformly convex Banach spaces. Rend. Circ. Mat. Palermo, II. Ser 72, 1463-1481 (2023). https://doi.org/10.1007/s12215-022-00746-7

[7] Bruck, R.: On the weak convergence of an ergodic iteration for the solution of variational inequalities for monotone operators in Hilbert space. J. Math. Anal. Appl. 61(1), 159-164 (1977). https://doi.org/10.1016/0022-247X(77)90152-4

[8] Combettes, P.L., Pesquet, J. C.: Proximal Splitting Methods in Signal Processing. In: Bauschke, H., Burachik, R., Combettes, P., Elser, V., Luke, D. and Wolkowicz, H., Ed., Fixed-Point Algorithms for Inverse Problems in Science and Engineering, Springer, 185-212 (2011). https://doi.org/10.1007/978-1-4419-9569-8_10

[9] Craft, D., Bangert, M., Long, T., Papp, D., Unkelbach, J.: Shared data for intensity modulated radiation therapy (IMRT) optimization research: the CORT dataset. GigaScience, 3(1), 37. https://doi.org/10.1186/2047-217X-3-37

[10] Davis, D., Yin, W.: A Three-operator splitting scheme and its optimization applications. Set-Valued Var. Anal. 25, 829-858 (2017). https://doi.org/10.1007/s11228-017-0421-z

[11] Dong, Q. L., Su, M., Shehu, Y.: Three-operator reflected forward-backward splitting algorithm with double inertial effects. Optim. Methods Softw. 39(2), 431-456 (2024). https://doi.org/10.1080/10556788.2024.2307470

[12] Eckstein, J.: Nonlinear Proximal Point Algorithms Using Bregman Functions, with Applications to Convex Programming. Math. Oper. Res. 18(1), 202-226 (1993). http://www.jstor.org/stable/3690161

[13] Eckstein, J., Svaiter, B. F.: A family of projective splitting methods for the sum of two maximal monotone operators. Math. Program. 111(1), 173-199 (2008). https://doi.org/10.1007/s10107-006-0070-8

[14] Eckstein, J., Svaiter, B. F.: General projective splitting methods for sums of maximal monotone operators. SIAM J. Control Optim. 48(2), 787-811 (2009). https://doi.org/10.1137/070698816

[15] Fu, A., Taasti, V.T., & Zarepisheh, M.:Distributed and Scalable Optimization for Robust Proton Treatment Planning. Med. Phys. 50(1), 633-642 (2023). https://doi.org/10.1002/mp.15897

[16] Gnacadja, E., Hernalsteens, C., Boogert, S., et al.: Optimization of proton therapy eye-treatment systems toward improved clinical performances.Phys. Rev. Res. 4, 013114 (2022). https://doi.org/10.1103/PhysRevResearch.4.013114

[17] Johnstone, P. R., Eckstein, J.: Projective splitting with forward steps. Math. Program. 191, 631-670 (2022). https://doi.org/10.1007/s10107-020-01565-3

[18] Johnstone, P. R., Eckstein, J.: Projective splitting with forward steps only requires continuity. Optim. Lett. 14, 229-247 (2020). https://doi.org/10.1007/s11590-019-01509-7

[19] Johnstone, P. R., Eckstein, J.: Single-forward-step projective splitting: exploiting cocoercivity. Comput. Optim. Appl. 78(1), 125-166 (2021). https://doi.org/10.1007/s10589-020-00238-3

[20] Johnstone, P. R., Eckstein, J.: Convergence rates for projective splitting. SIAM J. Optim. 29(3), 1931-195 (2019). https://doi.org/10.1137/18M1203523

[21] Johnstone, P. R., Eckstein, J., Flynn, T., Yoo, S.: Stochastic projective splitting. Comput. Optim. Appl. 87, 397-437 (2024). https://doi.org/10.1007/s10589-023-00528-6

[22] Kitkuan, D., Kumam, P., Padcharoen, A., Kumam, W., Thounthong, P.: Algorithms for zeros of two accretive operators for solving convex minimization problems and its application to image restoration problems. J. Comp. Appl. Math. 354, 471-495 (2019)

[23] Lions, P. L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Anal. 16, 964-979 (1979). https://doi.org/10.1137/0716071

[24] Machado, M. P.: On the Complexity of the Projective Splitting and Spingarn's Methods for the Sum of Two Maximal Monotone Operators. J. Optim. Theory. Appl. 178, 153-190 (2018). https://doi.org/10.1007/s10957-018-1310-9

[25] Malitsky, Y., Tam, M. K.: A forward-backward splitting method for monotone inclusions without cocoercivity. SIAM J. Optim. 30(2), 1451-1472 (2020). https://doi.org/10.1137/18M1207260

[26] Marques Alves, M., Geremia, M., Marcavillaca, R. T.: A relative-error inertial-relaxed inexact projective splitting algorithm. J. Convex Anal. 31(1), 001-024 (2024)

[27] Opial, Z.: Weak convergence of the sequence of successive approximations for nonexpansive mappings. Bull. Amer. Math. Soc., 73:591–597, 1967

[28] Paganetti, H.: Range uncertainties in proton therapy and the role of Monte Carlo simulations. Phys. Med. Biol., 57(11), R99-R117 (2012). https://doi.org/10.1088/0031-9155/57/11/R99

[29] Passty, G. B.: Ergodic convergence to a zero of the sum of monotone operators in Hilbert spaces. J. Math. Anal. Appl. 72, 383-390 (1979). https://doi.org/10.1016/0022-247X(79)90234-8

[30] Peaceman, D. H., Rachford, H. H.: The numerical solutions of parabolic and elliptic differential equations. J. Soc. Ind. Appl. Math. 3, 28-41 (1955). https://doi.org/10.1137/0103003

[31] Rieger, J., Tam, M. K.: Backward-forward-reflected-backward splitting for three operator monotone inclusions. Appl. Math. Comput. 381, 125-248 (2020). https://doi.org/10.1016/j.amc.2020.125248

[32] Sadeghi, H., Banert, S., Giselsson, P. : *Forward-backward splitting with deviations for monotone inclusions,* Appl. Set-Valued Anal. Optim. 6 (2024), 113-135

[33] Schwarz, M.: Treatment planning in proton therapy. Eur. Phys. J. Plus 126, 67 (2011). https://doi.org/10.1140/epjp/i2011-11067-y

[34] Spingarn, J. E.: Partial inverse of a monotone operator. Appl. Math. Optim. 10, 247-265 (1983). https://doi.org/10.1007/BF01448388

[35] Svaiter, B. F.: On weak convergence of the Douglas-Rachford method. SIAM J. Control Optim. 49, 280-287 (2011). https://doi.org/10.1137/100788100

[36] Taasti, V.T., Hong, L., Deasy, J. O., Zarepisheh, M.: Automated proton treatment planning with robust optimization using constrained hierarchical optimization. Med. Phys., 47, 2779-2790 (2020). https://doi.org/10.1002/mp.14148

[37] Taasti, V. T., Bäumer, C., Dahlgren, CV., et al.: Inter-centre variability of CT-based stopping-power prediction in particle therapy: Survey-based evaluation. Phys. Imaging Radiat. Oncol., 6(1), 25-30 (2018). https://doi.org/10.1016/j.phro.2018.04.006

[38] Thekumparampil, K. K., He, N., Oh, S.: Lifted primal-dual method for bilinearly coupled smooth minimax optimization. In: Camps-Valls, G., Ruiz, F. J. R., Valera, I. (Eds.), Proceedings of The 25th International Conference on Artificial Intelligence and Statistics, PMLR 151:4281-4308, (2022)

[39] Tseng, P.: A modified forward-backward splitting method for maximal monotone mappings. SIAM J. Control Optim. 38(2), 431-446 (2000). https://doi.org/10.1137/S0363012998338806

[40] Unkelbach, J., Paganetti, H.: Robust Proton Treatment Planning: Physical and Biological Optimization.Semin. Radiat. Oncol. 28(2), 88-96 (2018). https://doi.org/10.1016/j.semradonc.2017.11.005

[41] Worapitponga, W., Chaipunyaa, P., Kumamb, P., Salisub, S.: Approximating fixed points of nonexpansive-type mappings in Hadamard spaces. J. Math. Computer Sci., 39 (2025), 522–540

[42] Zhang, X.: A Review of the Robust Optimization Process and Advances with Monte Carlo in the Proton Therapy Management of Head and Neck Tumors.Int. J. Part. Ther., 8(1), 14-24 (2021), https://doi.org/10.14338/IJPT-20-00078.1