Douglas–Rachford Splitting Method with Linearization for the Split Feasibility Problem

Ziyue Hu [1], Qiaoli Dong [1], Yuchao Tang [2] and Michael Th. Rassias [3,4,*]

[1] College of Science, Civil Aviation University of China, Tianjin 300300, China; 2020061009@cauc.edu.cn (Z.H.); dongql@lsec.cc.ac.cn (Q.D.)
[2] Department of Mathematics, NanChang University, Nanchang 330031, China; hhaaoo1331@163.com
[3] Department of Mathematics and Engineering Sciences, Hellenic Military Academy, 16673 Vari Attikis, Greece
[4] Program in Interdisciplinary Studies, Institute for Advanced Study, 1 Einstein Drive, Princeton, NJ 08540, USA
[*] Correspondence: michail.rassias@math.uzh.ch or mthrassias@yahoo.com

Abstract: The aim of this article is to introduce the Douglas–Rachford splitting method with linearization to solve the split feasibility problem (SFP). Our proposed method includes two existing methods in work of Tang et al. and Wang as special cases. The ranges of the parameters in work of Tang et al. are extended from (0,1) to (0,2). Under standard conditions, we prove the weak convergence of proposed algorithms. We also provide two numerical experiments to illustrate the effectiveness of the proposed algorithm by comparing the algorithms in work of Tang et al. and Wang.

Keywords: split feasibility problem; Douglas–Rachford splitting method with linearization; relaxed Douglas–Rachford splitting method with linearization; CQ algorithm

MSC: 47H05; 47H07; 47H10; 54H25

1. Introduction

In this article, we consider the split feasibility problem (SFP) in the form

$$\text{find} \quad x^* \in C \quad \text{and} \quad Ax^* \in Q, \tag{1}$$

where C and Q are the nonempty closed convex subsets of the real Hilbert spaces $\mathcal{H}_1$ and $\mathcal{H}_2$, respectively, and $A : \mathcal{H}_1 \to \mathcal{H}_2$ is a bounded linear operator. The SFP was introduced by Censor and Elfving [1] for inverse problems which arise from phase retrievals and medical image reconstruction [2] and were later generalized to the multiple-sets split feasibility problem [3].

Throughout this article, we assume that the SFP (1) is consistent, i.e., its solution set, denoted by

$$\Gamma = \{x \,|\, x \in C \quad \text{and} \quad Ax \in Q\},$$

is nonempty.

By transforming the SFP as an equivalent constrained optimization problem, Byrne [2,4] firstly introduced the well-known CQ algorithms, which have received a great deal of attention from many authors, who improved it in various ways; see, e.g., [5–12].

To solve the SFP (1), Wang [13] introduced the following gradient method which was called Polyak's gradient method:

$$x^{k+1} = x^k - \tau_k \left[(x^k - P_C x^k) + A^*(I - P_Q)Ax^k \right], \tag{2}$$

where

$$\tau_k = \rho_k \frac{\|x^k - P_C x^k\|^2 + \|(I - P_Q)Ax^k\|^2}{2\|(x^k - P_C x^k) + A^*(I - P_Q)Ax^k\|^2},$$

and $\rho_k \in (0,2)$. In fact, (2) is a special case of the so-called simple projection method in [14]. The authors [15] proposed an iterative method for solving the multiple-sets split feasibility problem with splitting self-adaptive stepsizes, which reduces to the following scheme when applied to the SFP (1):

$$x^{k+1} = x^k - \left[\mu_k (x^k - P_C x^k) + \nu_k A^* (I - P_Q) A x^k \right], \tag{3}$$

where $\mu_k \in (0,1)$ and

$$\nu_k = \sigma_k \frac{\|(I - P_Q) A x^k\|^2}{\|A^*(I - P_Q) A x^k\|^2},$$

and $\sigma_k \in (0,1)$. Note that the difference of the schemes (2) and (3) is the choices of the stepsizes.

The Douglas–Rachford splitting method introduced in [16] is a popular method to solve structured optimization problems. Let g and h be proper lower semicontinuous convex functions from a Hilbert space $\mathcal{H}$ to $(-\infty, +\infty)$. Consider the structured optimization problem:

$$\min_{x \in \mathcal{H}} g(x) + h(x).$$

Given the initial guess x^0, the Douglas–Rachford splitting method generates the iterative sequence via the following scheme:

$$\begin{cases} y^{k+1} = \underset{x \in \mathcal{H}}{\operatorname{argmin}} \left\{ g(y) + \dfrac{1}{2\lambda_k} \|y - x^k\|^2 \right\}, \\ z^{k+1} = \underset{z \in \mathcal{H}}{\operatorname{argmin}} \left\{ h(z) + \dfrac{1}{2\lambda_k} \|z - (2y^{k+1} - x^k)\|^2 \right\}, \\ x^{k+1} = x^k + \alpha_k (z^{k+1} - y^{k+1}), \end{cases} \tag{4}$$

where $\{\alpha_k\}_{k \in \mathbb{N}} \subset (0,2)$ is a parameter sequence and $\{\lambda_k\}_{k \in \mathbb{N}} \subset (0,\infty)$ is the proximal parameter sequence of the regularization terms. Note that the scheme (4) with $\{\alpha_k\}_{k \in \mathbb{N}} \subset (0,+\infty)$ was called the general splitting method in [17] and becomes the Peaceman–Rachford splitting method when $\{\alpha_k\}_{k \in \mathbb{N}} \subset [2,+\infty)$.

It is easy to see that the split feasibility problem (1) also equals the following unconstrained minimization problem:

$$\min_{x \in \mathcal{H}_1} \{ \iota_C(x) + f(x) \}, \tag{5}$$

where $f(x) = \frac{1}{2} \|(I - P_Q) A x\|^2$ and $\iota_C(x)$ is the indicator function of the set C; that is, $\iota_C(x) = 0$ if $x \in C$, otherwise $\iota_C(x) = +\infty$. Recall that the gradient of $f(x)$ is $\nabla f(x) = A^*(I - P_Q) A x$. Letting $g = f$ and $h = \iota_C$ in (4) and using the linearization technique, the authors [18] recently introduced the following general splitting methods with linearization for the SFP (1):

$$\begin{cases} y^{k+1} := x^k - \lambda_k \nabla f(x^k), \\ z^{k+1} := P_C(2y^{k+1} - x^k), \\ x^{k+1} := x^k + \alpha_k (z^{k+1} - y^{k+1}), \end{cases} \tag{6}$$

where $\{\alpha_k\}_{k \in \mathbb{N}}$ is a sequence of positive numbers and

$$\lambda_k := \begin{cases} \gamma \dfrac{f(x^k)}{\|\nabla f(x^k)\|^2}, & \text{if } \nabla f(x^k) \neq 0, \\ 0, & \text{otherwise,} \end{cases}$$

and $\gamma \in (0,2)$. The weak convergence of the algorithm (6) was established under the standard conditions, and the algorithm (6) has good numerical performance comparing the algorithm (2) and Algorithm 3.1 in [10].

In this article, we present an iterative scheme by letting $g = \iota_C$ and $h = f$ in (4) and using the linearization technique. The convergence of the corresponding scheme is analyzed.

The rest of the paper is constructed as follows. In Section 2, we recall some definitions and known results for further analysis. In Section 3, we present the Douglas–Rachford splitting method with linearization and its relaxed version. In Section 4, we show the weak convergence of the proposed algorithm, which converges weakly to a solution of the SFP. In Section 5, we give two numerical experiments to show the behavior of the algorithm. Finally, some concluding remarks are given in Section 6.

2. Preliminaries

Let $\mathcal{H}$ be a Hilbert space and K be a nonempty closed convex subset of $\mathcal{H}$. We use the notation:

- $\rightharpoonup$ for weak convergence and $\rightarrow$ for strong convergence;
- $\omega_w(x^k) = \{x : \exists x^{k_l} \rightharpoonup x\}$ denotes the weak ω-limit set of $\{x^k\}_{k\in\mathbb{N}}$.

The following identity will be used for the main results (see [19], Corollary 2.15):

$$\|\alpha x + (1-\alpha)y\|^2 = \alpha\|x\|^2 + (1-\alpha)\|y\|^2 - \alpha(1-\alpha)\|x-y\|^2 \tag{7}$$

for all $\alpha \in \mathbb{R}$ and $(x,y) \in \mathcal{H} \times \mathcal{H}$.

Definition 1 ([19], Definition 6.38). *Let K be a nonempty convex subset of $\mathcal{H}$ and let $x \in \mathcal{H}$. The normal cone to K at x is*

$$N_K(x) = \begin{cases} \{u \in \mathcal{H} \mid \sup\langle K - x, u\rangle \leq 0\}, & \text{if} \quad x \in K; \\ \varnothing, & \text{otherwise.} \end{cases}$$

For a point x, the the classical metric projection of x onto K, denoted by $P_K(x)$, is defined by

$$P_K(x) := \operatorname{argmin}\{\|x - y\| : \forall y \in K\}.$$

Lemma 1 ([19], Proposition 4.4). *For any $x, y \in \mathcal{H}$ and $z \in K$, the following hold:*

(i) $\|P_K(x) - P_K(y)\|^2 \leq \langle P_K(x) - P_K(y), x - y\rangle$;
(ii) $\|P_K(x) - z\|^2 \leq \|x - z\|^2 - \|P_K(x) - x\|^2$;
(iii) $\langle (I - P_K)x - (I - P_K)y, x - y\rangle \geq \|(I - P_K)x - (I - P_K)y\|^2$.

It follows from Lemma 1 (iii) that

$$\langle x - P_K(x), x - z\rangle \geq \|x - P_K(x)\|^2, \quad \forall x \in \mathcal{H}, \ \forall z \in K. \tag{8}$$

Recall that a mapping $T : \mathcal{H} \rightarrow \mathcal{H}$ is called to be *nonexpansive* if

$$\|Tx - Ty\| \leq \|x - y\|, \quad \forall x, y \in \mathcal{H}.$$

Lemma 1 (i) implies that P_K is firmly nonexpansive and consequently nonexpansive. The next lemma shows that the nonexpansive mappings are demiclosed at 0.

Lemma 2 ([19], Theorem 4.27). *Let K be a nonempty closed convex subset of $\mathcal{H}$ and $T : K \rightarrow \mathcal{H}$ be a nonexpansive mapping. Let $\{x^k\}_{k\in\mathbb{N}}$ be a sequence in K and $x \in \mathcal{H}$ such that $x^k \rightharpoonup x$ and $Tx^k - x^k \rightarrow 0$ as $k \rightarrow +\infty$. Then $x \in \mathrm{Fix}(T)$.*

Lemma 3 ([19], Lemma 2.47). *Let K be a nonempty closed convex subset of $\mathcal{H}$ and $\{x^k\}_{k\in\mathbb{N}}$ be a sequence in $\mathcal{H}$ such that the following two conditions hold:*

(i) *For all $x \in K$, $\lim_{n\to\infty} \|x^k - x\|$ exists;*

(ii) *Every sequential weak cluster point of $\{x^k\}_{k\in\mathbb{N}}$ is in K.*

Then the sequence $\{x^k\}_{k\in\mathbb{N}}$ converges weakly to a point in K.

3. Douglas–Rachford Splitting Method with Linearization

In this section, we introduce Douglas–Rachford splitting method with linearization and its relaxed variant.

Using symmetry, we can set $g = \iota_C$ and $h = f$ in (4). Now, we present a direct adaptation of the Douglas–Rachford splitting method (4), which can solve the equivalent problem (5) of the SFP (1). (see Algorithm 1)

Algorithm 1 Douglas–Rachford splitting method.

Step 0. Input $k := 0$, $x^0 \in \mathcal{H}_1$.
Step 1. Generate x^{k+1} by

$$\begin{cases} y^{k+1} := P_C(x^k), \\ z^{k+1} := \underset{z\in\mathcal{H}_1}{\mathrm{argmin}}\left\{ f(z) + \dfrac{1}{2\lambda_k}\|z - (2y^{k+1} - x^k)\|^2 \right\}, \\ x^{k+1} := x^k + \alpha_k\left(z^{k+1} - y^{k+1}\right), \end{cases} \tag{9}$$

where $\{\alpha_k\}_{k\in\mathbb{N}}$ and $\{\lambda_k\}_{k\in\mathbb{N}}$ are sequences of positive numbers.
Step 2. If a termination criterion is not met, then set $k := k + 1$ and go to Step 1.

By the first-order optimality condition of the second formula of (9) in Algorithm 1,

$$\begin{aligned} 0 =& \nabla f(z^{k+1}) + \frac{1}{\lambda_k}\left[z^{k+1} - (2y^{k+1} - x^k) \right] \\ =& A^*(I - P_Q)Az^{k+1} + \frac{1}{\lambda_k}\left[z^{k+1} - (2y^{k+1} - x^k) \right]. \end{aligned}$$

To calculate z^{k+1}, we need to get $(I + \lambda_k A^*(I - P_Q)A)^{-1}$, which is very difficult. In order to overcome this difficulty, we linearize $f(z)$ at x^k. Then, the second formula of (9) becomes

$$\begin{aligned} z^{k+1} :=& \underset{z\in\mathcal{H}_1}{\mathrm{argmin}}\left\{ f(x^k) + \langle \nabla f(x^k), z - x^k \rangle + \frac{1}{2\lambda_k}\|z - (2y^{k+1} - x^k)\|^2 \right\} \\ =& \underset{z\in\mathcal{H}_1}{\mathrm{argmin}}\left\{ \langle \nabla f(x^k), z - x^k \rangle + \frac{1}{2\lambda_k}\|z - (2y^{k+1} - x^k)\|^2 \right\}. \end{aligned}$$

Its first-order optimality condition is

$$\nabla f(x^k) + \frac{1}{\lambda_k}[z^{k+1} - (2y^{k+1} - x^k)] = 0.$$

Then we have

$$z^{k+1} = 2y^{k+1} - x^k - \lambda_k \nabla f(x^k).$$

Thus, we get the following linearized Douglas–Rachford splitting method:

$$\begin{cases} y^{k+1} := P_C(x^k), \\ z^{k+1} := 2y^{k+1} - x^k - \lambda_k \nabla f(x^k), \\ x^{k+1} := x^k + \alpha_k \left(z^{k+1} - y^{k+1} \right). \end{cases}$$

After a simple calculation, we get

$$x^{k+1} := x^k + \alpha_k \left[P_C(x^k) - x^k - \lambda_k \nabla f(x^k) \right].$$

In what follows, we present the Douglas–Rachford splitting method with linearization. (see Algorithm 2)

Algorithm 2 Douglas–Rachford splitting method with linearization.

Step 0. Input $k := 0$, $x^0 \in \mathcal{H}_1$.
Step 1. Given x^k, generate x^{k+1} by

$$x^{k+1} := x^k - \alpha_k \left[(x^k - P_C(x^k)) + \lambda_k A^*(I - P_Q)Ax^k \right],$$

where $\{\alpha_k\}_{k \in \mathbb{N}}$ and $\{\lambda_k\}_{k \in \mathbb{N}}$ are two sequences of positive numbers.
Step 2. If $\nabla f(x^k) = 0$ and $x^{k+1} = x^k$, then terminate. Otherwise, set $k := k + 1$ and go to Step 1.

Remark 1. *Algorithm* 2 *is a general scheme with two parameters sequences* $\{\alpha_k\}_{k \in \mathbb{N}}$ *and* $\{\lambda_k\}_{k \in \mathbb{N}}$, *which includes the algorithms* (2) *and* (3) *as special cases. Indeed,*

(i)   *Let* $\alpha_k = \tau_k$ *and* $\lambda_k \equiv 1$, $\forall k \in \mathbb{N}$; *then, Algorithm* 2 *becomes the algorithm* (2);
(ii) *Let* $\alpha_k = \mu_k$ *and* $\lambda_k = \frac{\nu_k}{\mu_k}$, $\forall k \in \mathbb{N}$; *then, Algorithm* 2 *becomes the algorithm* (3).

In the following, we extend the ranges of the parameter sequences $\{\mu_k\}_{k \in \mathbb{N}}$ and $\{\sigma_k\}_{k \in \mathbb{N}}$ in the algorithm (3). To this end, let $\alpha_k = \beta_k \theta$ and

$$\lambda_k = \frac{(1 - \beta_k)\rho}{\alpha_k} \eta_k,$$

where $\beta_k \in (0, 1)$, $\theta \in (0, 2)$, $\rho \in (0, 2)$ and

$$\eta_k = \frac{\|(I - P_Q)Ax^k\|^2}{\|A^*(I - P_Q)Ax^k\|^2}.$$

Then, Algorithm 2 becomes

$$x^{k+1} = x^k - \beta_k \theta (x^k - P_C(x^k)) - (1 - \beta_k)\rho \eta_k A^*(I - P_Q)Ax^k. \tag{10}$$

Note that $\beta_k \theta \in (0, 2)$ and $(1 - \beta_k)\rho \in (0, 2)$. Comparing (3) and (10), the ranges of the parameter sequences $\{\mu_k\}_{k \in \mathbb{N}}$ and $\{\sigma_k\}_{k \in \mathbb{N}}$ in the algorithm (3) are extended from (0,1) to (0,2). It is worth noting that $\{\mu_k\}_{k \in \mathbb{N}}$ and $\{\sigma_k\}_{k \in \mathbb{N}}$ cannot both be in (1,2). In fact, one is in (0,2), while the other is in (0,1). Due to $\alpha_k \in (0, 2)$, Algorithm 2 is the Douglas–Rachford splitting method with linearization.

We give two choices of β_k, as follows:

- **Constant choice:** $\beta_k := \beta \in (0, 1)$ for all $k \in \mathbb{N}$;

- **Adaptive choice**: Let $\underline{\beta}, \overline{\beta} \in (0, 1)$,

$$\beta_k^* = \frac{1}{2} + \frac{\theta(2 - \theta)\left\|x^k - P_C(x^k)\right\|^2 - \rho(2 - \rho)\eta_k\left\|(I - P_Q)Ax^k\right\|^2}{2\left\|\theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k\right\|^2}, \tag{11}$$

and $\beta_k = \max\{\underline{\beta}, \min\{\beta_k^*, \overline{\beta}\}\}$.

Note that the upper and lower bounds are imposed in the adaptive choice of β_k to guarantee $\beta_k \in (0, 1)$. Therefore, it is not fully adaptive.

Now, the following lemma shows the validity of the stopping criterion through Step 2 of the Douglas–Rachford splitting method with linearization.

Lemma 4. *If $\nabla f(x^k) = 0$ and $x^{k+1} = x^k$ for some k, then x^k generated by Algorithm 2 is a solution of the SFP (1).*

Proof. Using $\nabla f(x^k) = 0$ and $x^{k+1} = x^k$, we get $x^k = P_C(x^k)$, which implies $x^k \in C$. Thus, $0 \in N_C(x^k)$. We deduce that x^k is a solution of the SFP (1) because of the first-order optimality condition $0 \in N_C(x) + \nabla f(x)$ of problem (5). □

In Algorithm 2, we generally assume that the projections P_C and P_Q are easy to calculate. However, projection is sometimes impossible or difficult to calculate. In order to solve this problem, we consider a general situation of C and Q in SFP (1). $C = \{x \in \mathcal{H}_1 : c(x) \leq 0\}$ and $Q = \{y \in \mathcal{H}_2 : q(y) \leq 0\}$ are level sets, where $c : \mathcal{H}_1 \to \mathbb{R}$ and $q : \mathcal{H}_2 \to \mathbb{R}$ are convex functions. We assume that ∂c and ∂q are bounded operators and define the sets C_k and Q_k as follows:

$$C_k = \left\{x \in \mathcal{H}_1 : c(x^k) + \langle \xi^k, x - x^k \rangle \leq 0\right\},$$

where $\xi^k \in \partial c(x^k)$, and

$$Q_k = \left\{y \in \mathcal{H}_2 : q(Ax^k) + \langle \eta^k, y - Ax^k \rangle \leq 0\right\},$$

where $\eta^k \in \partial q(Ax^k)$.

Next, we define $f_k(x) := \frac{1}{2}\|(I - P_{Q_k})Ax\|^2$ and introduce the relaxed Douglas–Rachford splitting method with linearization. (see Algorithm 3)

Algorithm 3 Relaxed Douglas–Rachford splitting method with linearization.

Step 0. Input $k := 0$, $x^0 \in \mathcal{H}_1$.
Step 1. Given x^k, generate x^{k+1} by

$$x^{k+1} := x^k - \alpha_k\left[(x^k - P_{C_k}(x^k)) + \lambda_k A^*(I - P_{Q_k})Ax^k\right], \tag{12}$$

where $\{\alpha_k\}_{k\in\mathbb{N}}$ and $\{\lambda_k\}_{k\in\mathbb{N}}$ are two sequences of positive numbers.
Step 2. If $\nabla f_k(x^k) = 0$ and $x^{k+1} = x^k$, then terminate. Otherwise, set $k := k + 1$ and go to Step 1.

---

4. Convergence Analysis

In this section, we prove the weak convergence of Algorithm 2 under the standard conditions.

Firstly, we present two lemmas which are key for the convergence of Algorithm 2.

Lemma 5. *Let $\{x^k\}_{k\in\mathbb{N}}$ be the sequence generated by Algorithm 2 from any initial point $x^0 \in \mathcal{H}_1$ and $x^* \in \Gamma$. Let $\beta_k \equiv \beta \in (0,1), \forall k \in \mathbb{N}$. Then it holds*

$$
\begin{aligned}
\left\| x^{k+1} - x^* \right\|^2 \leq& \left\| x^k - x^* \right\|^2 - \beta\theta(2-\theta)\left\| x^k - P_C(x^k) \right\|^2 \\
& - (1-\beta)\rho(2-\rho)\eta_k \left\| (I - P_Q)Ax^k \right\|^2 \\
& - \beta(1-\beta)\left\| \theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k \right\|^2.
\end{aligned}
\tag{13}
$$

Proof. By (10) and (7), we have

$$
\begin{aligned}
&\left\| x^{k+1} - x^* \right\|^2 \\
=&\| \beta_k[x^k - \theta(x^k - P_C(x^k)) - x^*] + (1-\beta_k)[x^k - \rho\eta_k A^*(I - P_Q)Ax^k - x^*] \|^2 \\
=&\beta\left\| x^k - x^* - \theta(x^k - P_C(x^k)) \right\|^2 \\
&+ (1-\beta)\left\| x^k - x^* - \rho\eta_k A^*(I - P_Q)Ax^k \right\|^2 \\
&- \beta(1-\beta)\left\| \theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k \right\|^2.
\end{aligned}
\tag{14}
$$

Using (8), we get

$$
\left\langle x^k - x^*, x^k - P_C(x^k) \right\rangle \geq \left\| x^k - P_C(x^k) \right\|^2,
\tag{15}
$$

and

$$
\left\langle x^k - x^*, A^*(I - P_Q)Ax^k \right\rangle = \left\langle Ax^k - Ax^*, (I - P_Q)Ax^k \right\rangle \geq \left\| (I - P_Q)Ax^k \right\|^2.
\tag{16}
$$

By (15), we obtain

$$
\begin{aligned}
&\left\| x^k - x^* - \theta(x^k - P_C(x^k)) \right\|^2 \\
=&\left\| x^k - x^* \right\|^2 + \theta^2\left\| x^k - P_C(x^k) \right\|^2 - 2\theta\left\langle x^k - x^*, x^k - P_C(x^k) \right\rangle \\
\leq&\left\| x^k - x^* \right\|^2 - \theta(2-\theta)\left\| x^k - P_C(x^k) \right\|^2.
\end{aligned}
\tag{17}
$$

From (16) and the definition of η_k, it follows

$$
\begin{aligned}
&\left\| x^k - x^* - \rho\eta_k A^*(I - P_Q)Ax^k \right\|^2 \\
=&\left\| x^k - x^* \right\|^2 + \rho^2\eta_k^2\left\| A^*(I - P_Q)Ax^k \right\|^2 - 2\rho\eta_k\left\langle x^k - x^*, A^*(I - P_Q)Ax^k \right\rangle \\
\leq&\left\| x^k - x^* \right\|^2 + \rho^2\eta_k^2\left\| A^*(I - P_Q)Ax^k \right\|^2 - 2\rho\eta_k\left\| (I - P_Q)Ax^k \right\|^2 \\
=&\left\| x^k - x^* \right\|^2 - \rho(2-\rho)\eta_k\left\| (I - P_Q)Ax^k \right\|^2.
\end{aligned}
\tag{18}
$$

Combining (14), (17) and (18), we deduce (13). The proof is completed. □

Remark 2. *The inequality* (13) *with the ranges of the parameters* β, θ, ρ *shows the monotonically decreasing property of the sequence* $\{\|x^k - x^*\|\}_{k \in \mathbb{N}}$. *It is not the same as the inequality in* [15], *and the latter reduces to the following form for the SFP* (1):

$$
\left\|x^{k+1} - x^*\right\|^2 \leq \left\|x^k - x^*\right\|^2 - 2\mu_k(1 - \mu_k)\left\|x^k - P_C(x^k)\right\|^2
$$
$$
- 2\sigma_k(1 - \sigma_k)\eta_k\left\|(I - P_Q)Ax^k\right\|^2.
$$

It is difficult to compare them and show which is better in theory. The numerical experiments in Section 5 *illustrate that the optimal choices of the parameters of* $\{\mu_k\}_{k \in \mathbb{N}}$ *and* $\{\sigma_k\}_{k \in \mathbb{N}}$ *is that one is in (0,2) while the others are in (0,1).*

Lemma 6. *Let* $\{x^k\}_{k \in \mathbb{N}}$ *be the sequence generated by Algorithm* 2 *from any initial point* $x^0 \in \mathcal{H}_1$, $x^* \in \Gamma$. *Let* β_k *is given adaptively.*

(i)   *If* $\beta_k^* \in [\underline{\beta}, \overline{\beta}]$, *it holds*

$$
\left\|x^{k+1} - x^*\right\|^2
$$
$$
\leq \left\|x^k - x^*\right\|^2 - \frac{1}{2}\theta(2 - \theta)\left\|x^k - P_C(x^k)\right\|^2
$$
$$
- \frac{1}{2}\rho(2 - \rho)\eta_k\left\|(I - P_Q)Ax^k\right\|^2
$$
$$
- \frac{1}{4}\left\|\theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k\right\|^2
$$
$$
- \frac{\left[\theta(2 - \theta)\left\|x^k - P_C(x^k)\right\|^2 - \rho(2 - \rho)\eta_k\left\|(I - P_Q)Ax^k\right\|^2\right]^2}{4\left\|\theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k\right\|^2};
$$

(19)

(ii)   *If* $\beta_k^* \notin [\underline{\beta}, \overline{\beta}]$, *it holds*

$$
\left\|x^{k+1} - x^*\right\|^2 \leq \left\|x^k - x^*\right\|^2 - \underline{\beta}\theta(2 - \theta)\left\|x^k - P_C(x^k)\right\|^2
$$
$$
- (1 - \overline{\beta})\rho(2 - \rho)\eta_k\left\|(I - P_Q)Ax^k\right\|^2
$$
$$
- (\underline{\beta} - \overline{\beta}^2)\left\|\theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k\right\|^2.
$$

(20)

Proof. (i) In this case, $\beta_k = \beta_k^*$. Similar to Lemma 5, we have

$$
\left\|x^{k+1} - x^*\right\|^2 \leq \left\|x^k - x^*\right\|^2 - \beta_k\theta(2 - \theta)\left\|x^k - P_C(x^k)\right\|^2
$$
$$
- (1 - \beta_k)\rho(2 - \rho)\eta_k\left\|(I - P_Q)Ax^k\right\|^2
$$
$$
- \beta_k(1 - \beta_k)\left\|\theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k\right\|^2.
$$

(21)

Let

$$
d_k = \theta(2 - \theta)\left\|x^k - P_C(x^k)\right\|^2,
$$
$$
e_k = \rho(2 - \rho)\eta_k\left\|(I - P_Q)Ax^k\right\|^2,
$$
$$
m_k = \|\theta(x^k - P_C(x^k)) - \rho\eta_k A^*(I - P_Q)Ax^k\|^2.
$$

The inequality (21) becomes

$$
\|x^{k+1} - x^*\|^2 \leq \|x^k - x^*\|^2 - \beta_k d_k - (1 - \beta_k)e_k - \beta_k(1 - \beta_k)m_k.
$$

(22)

After simple calculations, we get

$$
\begin{aligned}
&- \beta_k d_k - (1 - \beta_k)e_k - \beta_k(1 - \beta_k)m_k \\
=& m_k \beta_k^2 - (d_k - e_k + m_k)\beta_k - e_k \\
=& m_k(\beta_k - \frac{d_k - e_k + m_k}{2m_k})^2 - m_k(\frac{1}{2} + \frac{d_k - e_k}{2m_k})^2 - e_k \\
=& -\frac{1}{2}d_k - \frac{1}{2}e_k - \frac{m_k}{4} - \frac{(d_k - e_k)^2}{4m_k},
\end{aligned}
\tag{23}
$$

where the last equality holds due to (11). Combining (22) and (23), we get (19);
(ii) In this case, $\beta_k = \underline{\beta}$ or $\beta_k = \overline{\beta}$. By Lemma 5, it is easy to obtain (20). $\quad\square$

Theorem 1. *Let $\{x^k\}_{k\in\mathbb{N}}$ be generated by Algorithm 2; then, $\{x^k\}_{k\in\mathbb{N}}$ converges weakly to a solution of the SFP (1).*

Proof. By (13), (19) and (20), we have

$$
\|x^{k+1} - x^*\| \le \|x^k - x^*\|,
$$

which implies that $\{x^k\}_{k\in\mathbb{N}}$ is bounded and $\lim_{k\to\infty}\|x^k - x^*\|$ exists. From the definition of η_k, it follows

$$
\eta_k = \frac{\|(I - P_Q)Ax^k\|^2}{\|A^*(I - P_Q)Ax^k\|^2} \ge \frac{\|Ax^k - P_Q(Ax^k)\|^2}{\|A\|^2 \|Ax^k - P_Q(Ax^k)\|^2} = \frac{1}{\|A\|^2}.
\tag{24}
$$

Combining (13), (19), (20) and (24), we deduce

$$
\lim_{k\to\infty} \|P_C(x^k) - x^k\| = 0,
\tag{25}
$$

and

$$
\lim_{k\to\infty} \|(I - P_Q)Ax^k\| = 0.
\tag{26}
$$

To use Lemma 3, we need to show $\omega_w(x^k) \subseteq \Gamma$. To this end, take arbitrarily $\hat{x} \in \omega_w(x^k)$ and let $x^{k_l} \rightharpoonup \hat{x}$ where $\{x^{k_l}\}_{l\in\mathbb{N}}$ is a subsequence of $\{x^k\}_{k\in\mathbb{N}}$. From Lemma 2 and (25), we get $\hat{x} \in C$. Combining (26) and the weak lower semicontinuity of f, we obtain

$$
0 \le f(\hat{x}) \le \liminf_{l\to\infty} f(x^{k_l}) = \lim_{k\to\infty} f(x^k) = 0.
$$

Hence, $f(\hat{x}) = \frac{1}{2}\|(I - P_Q)A\hat{x}\| = 0$, i.e., $A\hat{x} \in Q$. Thus, we have $\omega_w(x^k) \subseteq \Gamma$. By Lemma 3, we deduce that $\{x^k\}_{k\in\mathbb{N}}$ converges weakly to a point in Γ. The proof is completed. $\quad\square$

Combining Theorem 1 and [18] (Theorem 3.2), it is easy to verify the convergence of Algorithm 3.

Theorem 2. *Let $\{x^k\}_{k\in\mathbb{N}}$ be generated by Algorithm 3; then, $\{x^k\}_{k\in\mathbb{N}}$ converges weakly to a solution of the SFP (1).*

5. Numerical Results

In this section, we show the behavior of Algorithm 2 by comparing it with the algorithms (3) and (2) through two numerical examples.

For convenience, we denote the vector with all elements 0 by e_0 and the vector with all elements 1 by e_1 in what follows. In the next two numerical examples, we take the objective function value $p(x) = \frac{1}{2}\|x - P_C(x)\|^2 + \frac{1}{2}\|Ax - P_Q(Ax)\|^2 \le \epsilon$ as the stopping criterion.

Example 1 ([6])**.** *Consider the SFP, where* $A = (a_{ij})_{m \times n} \in \mathbb{R}^{m \times n}$ *(m = 200, n = 500) and* $a_{ij} \in (0, 1)$ *generated randomly and*

$$C = \{x \in \mathbb{R}^n : \|x - d\| \leq r\},$$
$$Q = \{y \in \mathbb{R}^m : L \leq y \leq U\},$$

where d is the center of the ball C, $e_0 \leq d \leq e_1$, $r \in (10, 20)$ *is the radius, d and r are both generated randomly. L and U are the boundary of the box Q and are also generated randomly, satisfying* $10e_1 \leq L \leq 20e_1$ *and* $20e_1 \leq U \leq 30e_1$, *respectively.*

In the numerical experiment, we take the objective function value $p(x^k) \leq \epsilon = 10^{-5}$ as the stopping criterion.

In Figure 1, the initial point $x^0 \in (0, 100e_1)$ is randomly chosen. For comparing Algorithm 2 with the algorithms (2) and (3), we take $\theta = 1.59$, $\rho = 1.86$ and $\beta_k \equiv \beta = 0.37$ in Algorithm 2, $\rho_k = 1.99$ in the algorithm (2) and $\mu_k = 0.72$ and $\nu_k = 0.88$ in the algorithm (3).
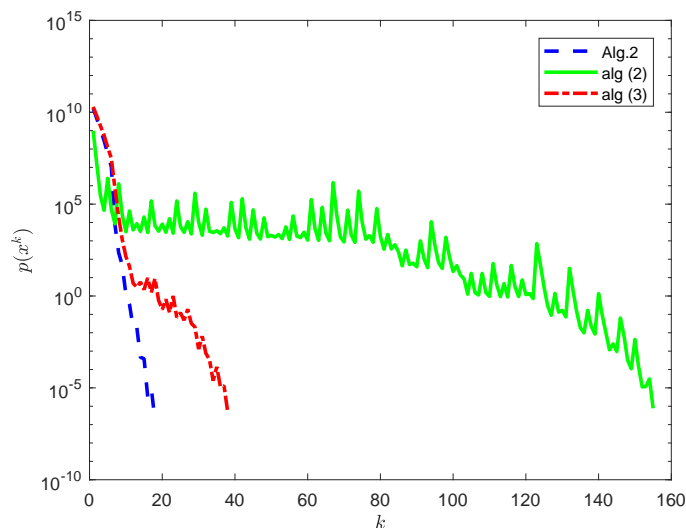
Figure 1. The value of $p(x^k)$ versus the iteration numbers for Example 1.

In Table 1, we show the iteration steps and CPU time of Algorithm 2, algorithm (2) and algorithm (3) for 3 initial points. For case 1, $x^0 = (100, \cdots, 100)$; for case 2, $x^0 = (100, -100, \cdots, 100, -100)$; and for case 3, $x^0 \in (-100e_1, 100e_1)$.

Table 1. Comparison of Algorithm 2, algorithms (2) and (3) under different initial points.

Cases	Algorithm 2		algorithm (2)		algorithm (3)	
	Iter.	CPU Time	Iter.	CPU Time	Iter.	CPU Time
Case 1	18	0.0313	155	0.0313	38	0.0469
Case 2	22	0.0469	372	0.0781	37	0.0469
Case 3	16	0.0313	256	0.0469	35	0.0313

As shown in Figure 1 and the iteration steps in Table 1, Algorithm 2 behaves better than the algorithms (2) and (3).

Example 2. *Suppose that* $\mathcal{H}_1 = \mathcal{H}_2 = L^2([0, 1])$ *with norm* $\|x\| = \left(\int_0^1 |x(t)|^2 dt \right)^{\frac{1}{2}}$ *and inner product*

$$\langle x, y \rangle = \int_0^1 x(t)y(t)dt.$$

Let

$$C := \{x \in \mathcal{H}_1 : \langle a, x \rangle = b\},$$

where $a = \sin(5t^3)$ *and* $b = 0$. *In this case, we have*

$$P_C(x) = x + \frac{b - \langle a, x \rangle}{\|a\|^2} a.$$

Additionally, let

$$Q := \{x \in \mathcal{H}_2 : \langle c, x \rangle \leq d\},$$

where $c = \sin(t/4)$ *and* $d = 1/5$; *then, we obtain*

$$P_Q(x) = \begin{cases} x & \text{if } \langle c, x \rangle \leq d \\ x + \frac{d - \langle c, x \rangle}{\|c\|^2} c & \text{if } \langle c, x \rangle > d. \end{cases}$$

Consider the matrix

$$A = \frac{1}{200} \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 2 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 2 & 2 & \cdots & 1 & 0 \\ 1 & 2 & 2 & \cdots & 2 & 1 \end{pmatrix}_{101 \times 101}$$

resulting from a discretization of the operator question of the first kind

$$y(t) = (Ax)(t) := \int_0^t x(s)ds, \quad s, t \in [0, 1]; \quad x, y : [0, 1] \longrightarrow [0, 1].$$

In the numerical experiment, the initial function is $x^0(t) = 10\sin(t^2 + 2t)$. We take $p(x^k) \leq \epsilon = 10^{-10}$ as the stopping criterion.

For comparing Algorithm 2 with the algorithms (2) and (3), we take $\theta = 1.99$, $\rho = 1.99$ and $\beta_k \equiv \beta = 0.41$ in Algorithm 2, $\rho_k = 1.89$ in the algorithm (2) and $\mu_k = 0.5$ and $\nu_k = 0.99$ in the algorithm (3). From Figure 2, we can see that the performance of Algorithm 2 is better than the algorithms (2) and (3).
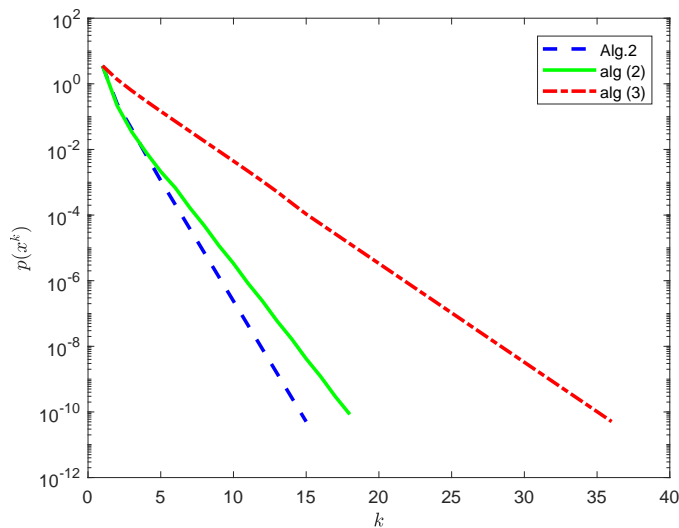

Figure 2. The value of $p(x^k)$ versus the iteration numbers for Example 2.

6. Some Concluding Remarks

In this article, we introduce the Douglas–Rachford splitting method with linearization for the split feasibility problem, which is a general method and includes the methods in [13,15] as special cases. The weak convergence of the proposed method is established under the standard conditions. Numerical experiments illustrate the effectiveness of our methods.

The methods proposed in this article can be generalized to solve the multiple-sets split feasibility problem. It is interesting to investigate the other possible choices of the parameters $\{\alpha_k\}_{k\in\mathbb{N}}$ and $\{\lambda_k\}_{k\in\mathbb{N}}$.

Recently, some authors have applied self-adaptive step sizes to split generalized equilibrium and fixed point problems [20] and pseudo-monotone equilibrium problems [21]. The numerical examples illustrate that the step sizes have excellent behaviour. Applying the self-adaptive step size to the split feasibility problem is worth investigating.

References

1. Censor, Y.; Elfving, T. A multiprojection algorithm using Bregman projections in a product space. *Numer. Algorithms* **1994**, *8*, 221–239. [CrossRef]
2. Byrne, C.L. Iterative oblique projection onto convex sets and the split feasibility problem. *Inverse Probl.* **2002**, *18*, 441–453. [CrossRef]
3. Censor, Y.; Elfving, T.; Kopf, N.; Bortfeld, T. The multiple-sets split feasibility problem and its applications for inverse problems. *Inverse Probl.* **2005**, *21*, 2071–2084. [CrossRef]
4. Byrne, C.L. A unified treatment of some iterative algorithms in signal processing and image reconstruction. *Inverse Probl.* **2004**, *20*, 103–120. [CrossRef]
5. Dang, Y.; Sun, J.; Zhang, S. Double projection algorithms for solving the split feasibility problems. *J. Ind. Manag. Optim.* **2019**, *15*, 2023–2034. [CrossRef]
6. Dong, Q.L.; Tang, Y.C.; Cho, Y.J.; Rassias, T.M. "Optimal" choice of the step length of the projection and contraction methods for solving the split feasibility problem. *J. Glob. Optim.* **2018**, *71*, 341–360. [CrossRef]
7. Dong, Q.L.; Yao, Y.; He, S. Weak convergence theorems of the modified relaxed projection algorithms for the split feasibility problem in hilbert spaces. *Optim. Lett.* **2014**, *8*, 1031–1046. [CrossRef]
8. He, S.; Xu, H.K. The selective projection method for convex feasibility and split feasibility problems. *J. Nonlinear Sci. Appl.* **2018**, *19*, 1199–1215.
9. Gibali, A.; Liu, L.; Tang, Y.C. Note on the modified relaxation CQ algorithm for the split feasibility problem. *Optim. Lett.* **2018**, *12*, 817–830. [CrossRef]
10. López, G.; Martín-Márquez, V.; Wang, F.; Xu, H.K. Solving the split feasibility problem without prior knowledge of matrix norms. *Inverse Probl.* **2012**, *27*, 085004. [CrossRef]
11. Qu, B.; Wang, C.; Xiu, N. Analysis on Newton projection method for the split feasibility problem. *Comput. Optim. Appl.* **2017**, *67*, 175–199. [CrossRef]
12. Wang, J.; Hu, Y.; Li, C.; Yao, J.C. Linear convergence of CQ algorithms and applications in gene regulatory network inference. *Inverse Probl.* **2017**, *33*, 055017. [CrossRef]
13. Wang, F. Polyak's gradient method for split feasibility problem constrained by level sets. *Numer. Algorithms* **2018**, *77*, 925–938. [CrossRef]

14. Zhao, J.; Yang, Q. A simple projection method for solving the multiple-sets split feasibility problem. *Inverse Prob. Sci. Eng.* **2013**, *21*, 537–546. [CrossRef]

15. Tang, Y.; Zhu, C.; Yu, H. Iterative methods for solving the multiple-sets split feasibility problem with splitting self-adaptive step size. *Fixed Point Theory Appl.* **2015**, *2015*, 178. [CrossRef]

16. Douglas, J.; Rachford, H.H. On the numerical solution of heat conduction problems in two or three space variables. *Trans. Am. Math. Soc.* **1956**, *82*, 421–439. [CrossRef]

17. Li, M.; Wu, Z. Convergence analysis of the generalized splitting methods for a class of nonconvex optimization problems. *J. Optimiz. Theory Appl.* **2019**, *183*, 535–565. [CrossRef]

18. Dong, Q.L.; He, S.; Rassias, M.T. General splitting methods with linearization for the split feasibility problem. *J. Glob. Optim.* **2021**, *79*, 813–836. [CrossRef]

19. Bauschke, H.H.; Combettes, P.L. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*; Springer: New York, NY, USA, 2017; Volume 2.

20. Olona, M.A.; Alakoya, T.O.; Owolabi, A.O.-E.; Mewomo, O.T. Inertial shrinking projection algorithm with self-adaptive step size for split generalized equilibrium and fixed point problems for a countable family of nonexpansive multivalued mappings. *Demonstr. Math.* **2021**, *54*, 47–67. [CrossRef]

21. Pakkaranang, N.; Rehman, H.; Kumam, W. Two strongly convergent self-adaptive iterative schemes for solving pseudo-monotone equilibrium problems with applications. *Demonstr. Math.* **2021**, *54*, 280–298. [CrossRef]