CONVERGENCE ANALYSIS OF A PARALLEL PROJECTION ALGORITHM FOR SOLVING CONVEX FEASIBILITY PROBLEMS

YAZHENG DANG*
Business School
University of Shanghai for Science and Technology
Shanghai, China

FANWEN MENG
Department of Health Services and Outcomes Research
National Healthcare Group, Singapore

JIE SUN
Curtin University, Australia
and National University of Singapore, Singapore

(Communicated by Kok Lay Teo)

Abstract. The convex feasibility problem (CFP) is a classical problem in nonlinear analysis. In this paper, we propose an inertial parallel projection algorithm for solving CFP. Different from the previous algorithms, the proposed method introduces a sequence of parameters and uses the information of last two iterations at each step. To prove its convergence in a simple way, we transform the parallel algorithm to a sequential one in a constructed product space. Preliminary experiments are conducted to demonstrate that the proposed approach converges faster than the general extrapolated algorithms.

1. Introduction. In this paper, we are interested in designing new algorithms for solving the convex feasibility problem (CFP) of finding a point in the intersection of finitely many closed convex sets of a Euclid space. The CFP is a classical problem in nonlinear analysis with applications in a variety of areas including economics, physics, and applied sciences, for example, optimization [6], approximation theory [4,9], image reconstruction from projections and computerized tomography [5,15], control problem [3,10], to name only a few.

A number of algorithms were presented based on projection method for solving the CFP [1,12,14,17]. According to the number of projections adopted at each step, the sequential projection algorithm, the parallel projection algorithm and the block-iterative projection algorithm had been intensively investigated. The reader is referred to [2,11,16] for a detailed discussion in this regard.

2010 Mathematics Subject Classification. 49M37, 90C25, 90C90.

Key words and phrases. Convex feasible problem, inertial parallel projection algorithm, asymptotic convergence.

This work was supported by Natural Science Foundation of Shanghai (14ZR1429200), Innovation Program of Shanghai Municipal Education Commission (15ZZ074) and Australian Research Council (DP160102819).

* Corresponding author.
Convergence of algorithms is an important issue in numerical computation. In the past decade, a great deal of efforts have been paid to study the convergence of algorithms developed for CFP. In particular, to achieve faster convergence, an over-relaxed projection method [1], a non-monotonous projection method [8] and extrapolation projection methods [7, 18] were presented for solving CFP. In general, these methods basically used the current iterative point to obtain the next iteration point. However, this kind of design appears to underutilize available information. Certain research [24] has pointed out that algorithms using the information of two proceeding iterations at each step could converge faster than only using previous iteration for certain type of problems. However, very few research has been conducted along this direction. Hence, it is interesting and meaningful to explore the possibility of solving CFP by developing algorithms with more information from the previous steps.

In this paper, we shall investigate a new approach to solving convex feasibility problem following the above idea. Specifically, we shall develop a new accelerated method by applying the inertial technique [19–21] to parallel projection algorithm. In contrast to previous algorithms [1, 7, 18], the proposed method employs a sequence of parameters and two preceding iteration points to obtain the new iteration point. The algorithm utilizes more information available at each step, which enables us to find improved performance in comparison with previous algorithms. This feature is illustrated by preliminary experiments conducted in Section 5. Furthermore, to prove its convergence in a simple way, we transform the parallel algorithm to a sequential one in a constructed product space. We believe that the obtained results could complement and enrich the field of nonlinear analysis in both theory and applications.

The remainder of the paper is organized as follows. Section 2 summarizes some of the related notions and results. Section 3 describes the methodology adopted in the research in which an inertial parallel projection algorithm is presented for solving CFP. Section 4 presents the theoretical results in terms of convergence analysis. Some numerical tests are given in Section 5.

2. Preliminaries. We are concerned with the following convex feasibility problem

\[
\text{Find a point } x \in C = \bigcap_{i=1}^{m} C_i, \tag{1}
\]

where \( C_i \subset \mathbb{R}^n, i = 1, 2, \ldots, m, \) are closed convex sets. In the section, we summarize some notations and relevant results which will be used in the subsequent analysis. Let \( X \) be a finite dimensional vector space equipped with a scalar product denoted by \( \langle \cdot, \cdot \rangle \) and \( \| \cdot \| \) be the norm induced by \( \langle \cdot, \cdot \rangle \). Let \( P_C(\cdot) \) denote the projection mapping from \( X \) onto a nonempty closed convex set \( C \) of \( X \), that is,

\[
P_C(x) := \arg \min_{y \in C} \|x - y\|, \quad x \in X. \tag{2}
\]

We denote by \( I \) the identity operator in an arbitrary vector space. Throughout this paper, our focus is on \( X \) being the usual Euclidean space, namely, \( X = \mathbb{R}^n \). We denote by \( \langle \cdot, \cdot \rangle \) and \( \| \cdot \| \) the inner product and norm in \( (\mathbb{R}^n)^m \), respectively.

Let \( E \) be a real Banach space and \( S \) be a nonempty closed convex subset of \( E \). Recall that an operator \( T : S \to S \) is said to be \( \textit{non-expansive} \), if

\[
\|T(x) - T(y)\| \leq \|x - y\|, \quad x, y \in S, \tag{3}
\]
and firmly non-expansive, if
\[ \|T(x) - T(y)\|^2 \leq \langle x - y, T(x) - T(y) \rangle, \quad x, y \in S. \] \tag{4}

It is easy to see that (4) \Rightarrow (3).

Note that \( P_C(x) \) can be characterized by the following inequality
\[ \langle x - P_C(x), z - P_C(x) \rangle \leq 0, \quad \forall z \in C, \quad x \in \mathbb{R}^n. \] \tag{5}

The inequality (5) implies that for any \( x, y \in \mathbb{R}^n \) and a closed convex set \( C \subset \mathbb{R}^n \), the following inequalities hold
\[ \langle P_C(y) - P_C(x), P_C(x) - x \rangle \geq 0 \]
and
\[ \langle P_C(y) - P_C(x), y - P_C(y) \rangle \geq 0. \]

Taking the sum of the above two inequalities, we obtain that
\[ \langle P_C(y) - P_C(x), P_C(x) - P_C(y) - x + y \rangle \geq 0, \]
which can be further written as
\[ \langle P_C(y) - P_C(x), y - x \rangle \geq \|P_C(y) - P_C(x)\|^2. \]

Therefore, the operator \( P_C \) is firmly non-expansive. By virtue of Cauchy inequality, it yields that
\[ \|P_C(x) - P_C(y)\| \leq \|x - y\|, \]
that is, the operator \( P_C \) is non-expansive. From (5), it is not hard to derive that
\[ \|x - P_C(x)\|^2 \leq \langle x - z, x - P_C(x) \rangle, \quad \forall z \in C, \quad x \in \mathbb{R}^n. \] \tag{6}

The following results will be needed in convergence analysis of the proposed algorithm.

**Lemma 2.1** ([19]). For sequences \( \{\psi_k\}_{k=0}^\infty, \{\delta_k\}_{k=0}^\infty \) with \( \psi_k, \delta_k \in [0, \infty) \) for \( k = 0, 1, \ldots \), assume that
(a) for \( k = 1, 2, 3, \ldots \), \( \psi_{k+1} - \psi_k \leq \theta_k(\psi_k - \psi_{k-1}) + \delta_k \),
(b) \( \sum_{k=0}^{\infty} \delta_k < \infty \),
(c) for \( k = 0, 1, 2, \ldots \), \( \theta_k \in [0, \theta] \) where \( \theta \in [0, 1) \).

Then, the sequence \( \{\psi_k\} \) is convergent and \( \sum_{k=0}^{\infty} |\psi_{k+1} - \psi_k| < \infty \), where \( |t|_+ := \max\{t, 0\} \) for any \( t \in \mathbb{R} \).

**Lemma 2.2** ([19]). Assume that the parameters \( \delta_k, \gamma_k, k = 0, 1, 2, \ldots \), are positive and satisfy
\[ \sum_{k=0}^{\infty} \delta_k < +\infty, \quad \gamma_{k+1} \leq \gamma_k + \delta_k, \quad k = 0, 1, 2, \ldots \]

Then, the sequence \( \{\gamma_k\}_{k=0}^\infty \) is convergent.

Let \( T : C \to C \) be a non-expansive mapping, where \( C \) is a subset of a Hilbert space. We denote by \( \text{Fix}(T) \) the set of fixed points of operator \( T \), which is defined as \( \text{Fix}(T) := \{x \mid T(x) = x, \forall x \in C\} \).

**Lemma 2.3** ([23]). Let \( H \) be a Hilbert space, \( C \) a closed convex subset of \( H \), and \( T : C \to C \) a non-expansive mapping with \( \text{Fix}(T) \neq \emptyset \). If \( \{x^k\} \) is a sequence in \( C \) weakly converging to \( x \) and \( \{(I - T)(x^k)\} \) converges strongly to \( y \), then \( (I - T)(x) = y \).
Lemma 2.4 ([22]). Let $H$ be a Hilbert space, $\{x^k\}$ be a sequence in $H$. Suppose that there exists a nonempty set $S \subset H$ satisfying

(i) for every $x^* \in S$, $\lim_{k \to \infty} \|x^k - x^*\|$ exists,

(ii) any weak cluster point of $\{x^k\}$ belongs to $S$.

Then, there exists $\bar{x}^* \in S$ such that $\{x^k\}$ weakly converges to $\bar{x}^*$.

3. Algorithm description. In this section, we propose an algorithm called inertial parallel projection algorithm for solving problem (1). To facilitate convergence analysis, we further convert the proposed algorithm to a sequential one by constructing a product space.

3.1. Inertial parallel projection algorithm. We state the inertial parallel projection algorithm as follows.

Algorithm 3.1 Inertial parallel projection algorithm.

Initialization: Take $x^0, x^1$ in $\mathbb{R}^n$. Select relaxed factor $\lambda_k \in (0, 2)$ and inertial parameter $\theta_k \in [0, \theta]$, $\theta \in (0, 1)$ for all $k > 1, 2, \cdots$.

Iterative step:

$$
\begin{align*}
x^{k+1} &= y^k + \lambda_k \left( \sum_{i=1}^{m} \frac{1}{m} P_{C_i}(y^k) - y^k \right), \quad (7) \\
y^k &= x^k + \theta_k(x^k - x^{k-1}). \quad (8)
\end{align*}
$$

where $P_{C_i}()$ denotes the projection mapping onto the convex set $C_i$, $i = 1, \cdots, m$.

Note that, if $\theta_k \equiv 0$ for all $k$, Algorithm 3.1 then reduces to the general parallel projection algorithm, which has been widely investigated in the past.

3.2. Construction of a product space. To investigate the convergence of the algorithm, we transform the parallel projection algorithm (7)-(8) as a sequential one by introducing a product space as follows. Define

$$
\langle \langle V, W \rangle \rangle = (v_1, w_1) + \cdots + (v_m, w_m),
$$

$$
\|\|V\|\|^2 = \langle \langle V, V \rangle \rangle = \sum_{i=1}^{m} \|v_i\|^2,
$$

where $V = (v_1, v_2, \cdots, v_m) \in (\mathbb{R}^n)^m, W = (w_1, w_2, \cdots, w_m) \in (\mathbb{R}^n)^m$. We denote by $H = ((\mathbb{R}^n)^m, \langle \langle \cdot, \cdot \rangle \rangle, \|\| \cdot \|| \|)$ the product space equipped with the norm $\|\| \cdot \|| \|$ induced by the inner product $\langle \langle \cdot, \cdot \rangle \rangle$.

Next, we define two important subsets of the product space $H$, which will be used in the analysis. We first define $\mathcal{N}$ the Cartesian product of the convex sets $C_i, i = 1, 2, \cdots, m$, i.e., $\mathcal{N} := C_1 \times C_2 \times \cdots \times C_m$. Another set, $D$, is the image of a mapping $q : \mathbb{R}^n \to H$ on $H$ defined by $q(v) \equiv (v, v, \cdots, v)$ for any $v \in \mathbb{R}^n$. That is, $D := \{q(v) : \forall v \in \mathbb{R}^n\}$. It is evident that $D$ is a closed linear subspace of $H$.

In addition, if $C \neq \emptyset$, then $\mathcal{N} \cap D \neq \emptyset$. Moreover, $q(C) = \mathcal{N} \cap D$. Hence, for the underlying CFP (1), finding a point in $C \subset \mathbb{R}^n$ is equivalent to finding a point in $\mathcal{N} \cap D \subset H$. As a result, our subsequent analysis will focus on the latter problem.

3.3. A sequential parallel projection algorithm. In the constructed product space $H$, we next develop an alternative algorithm of Algorithm 3.1 for (1), which is generally a sequential projection approach. Before stating the algorithm, we investigate some interesting properties concerning the projections on two sets $\mathcal{N}$ and $D$. First, the following result is taken from [22].
Lemma 3.1 ([22]). Let \( V = (v_1, v_2, \ldots, v_m) \in H \). Denote \( P_R(\cdot) \) and \( P_D(\cdot) \) by the projections onto \( R \) and \( D \), respectively. Then

1. \( P_R(V) = (P_{C_1}(v_1), P_{C_2}(v_2), \ldots, P_{C_m}(v_m)) \),
2. \( P_D(V) = q(\sum_{i=1}^{m} \frac{1}{m} v_i) \).

In order to show the convergence results stated in Section 4, we first derive the following results as Propositions 3.1 and 3.2.

**Proposition 3.1.** Let \( P_D, P_R \) be as in Lemma 3.1. Then

\[
\langle \langle X - Z, X - P_D P_R(X) \rangle \rangle \geq \| X - P_D P_R(X) \|^2, \quad \forall Z \in D \cap R, \ X \in D.
\]

**Proof.** For any \( X \in D \), it follows from Lemma 3.1 that

\[
P_R(X) = (P_{C_1}(x), P_{C_2}(x), \ldots, P_{C_m}(x)),
\]

\[
P_D P_R(X) = \left( \sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x), \sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x), \ldots, \sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x) \right).
\]

By the definition of norm \( \| \cdot \| \) together with its convexity, we have

\[
\| X - P_D P_R(X) \|^2 = m \| x - \sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x) \|^2
\]

\[
= \frac{1}{m} \| \sum_{i=1}^{m} (x - P_{C_i}(x)) \|^2 \leq \frac{1}{m} \sum_{i=1}^{m} \| x - P_{C_i}(x) \|^2
\]

\[
= \frac{1}{m} \| X - P_R(X) \|^2 \leq \| X - P_R(X) \|^2.
\]

In addition, by the definition of \( \langle \langle \cdot, \cdot \rangle \rangle \), it yields that

\[
\langle X - Z, X - P_D P_R(X) \rangle
\]

\[
= m \langle x - z, x - \sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x) \rangle = \langle x - z, \sum_{i=1}^{m} (x - P_{C_i}(x)) \rangle
\]

\[
= \sum_{i=1}^{m} \langle x - z, x - P_{C_i}(x) \rangle = \langle \langle X - Z, X - P_R(X) \rangle \rangle
\]

\[
= \langle \langle X - P_R(X), X - P_R(X) \rangle \rangle + \langle \langle P_R(X) - Z, X - P_R(X) \rangle \rangle
\]

\[
= \| X - P_R(X) \|^2 + \langle \langle P_R(X) - Z, X - P_R(X) \rangle \rangle.
\]

Hence, according to the property of projection mapping and (9), it follows that

\[
\langle \langle X - Z, X - P_D P_R(X) \rangle \rangle \geq \| X - P_R(X) \|^2 \geq \| X - P_D P_R(X) \|^2.
\]

This completes the proof. \( \Box \)

**Proposition 3.2.** Let \( P_D, P_R \) be as in Lemma 3.1. Then \( \text{Fix} (P_D P_R) = \text{Fix} (P_D) \cap \text{Fix} (P_R) = D \cap R \).

**Proof.** We first show that “\( x \in \text{Fix} (P_D P_R) \Rightarrow x \in \text{Fix}(P_D) \cap \text{Fix}(P_R) \)”.

Let \( x \in \text{Fix}(P_D P_R) \). Then \( x = P_D P_R(x) \), and \( x = P_R(x) \), \( x = P_D(x) \). Hence \( x \in \text{Fix}(P_D) \cap \text{Fix}(P_R) \). Next, we shall prove the following statement: “\( \text{Fix}(P_D) \cap \text{Fix}(P_R) \subset \text{Fix}(P_D P_R) \)”.

Suppose \( x \in \text{Fix}(P_D) \cap \text{Fix}(P_R) \), we have \( x = P_R(x) \), \( x = P_D(x) \), and \( x = P_D P_R(x) \), that is, \( x \in \text{Fix}(P_D P_R) \). Evidently, \( \text{Fix}(P_D) \cap \text{Fix}(P_R) \subset \text{Fix}(P_D P_R) \). Therefore, \( \text{Fix} (P_D P_R) = \text{Fix}(P_D) \cap \text{Fix} (P_R) = D \cap R \). This completes the proof. \( \Box \)
We now present the inertial parallel projection Algorithm 3.1 as a sequential one.

Algorithm 3.2

Initialization: Take $x^0, x^1 \in \mathbb{R}^n$, the points $q(x^0), q(x^1)$ belong to $D$. Then, $X^0 = q(x^0), X^1 = q(x^1)$ are the initiative points for a sequential projection algorithm in $H$, select $\lambda_k \in (0, 2)$ and $\theta_k \in [0, \theta], \theta \in [0, 1)$, for $k = 1, 2, \ldots$.

Step 1: Let $y^1 = x^1 + \theta_1(x^1 - x^0)$. Then

$$Z^2 = q(y^1) + \lambda_1(P_D(q(y^1)) - q(y^1)), \quad X^2 = P_D(Z^2).$$

Iterative step:

\begin{align}
Y^k &= X^k + \theta_k(X^k - X^{k-1}), \quad (10) \\
X^{k+1} &= Y^k + \lambda_k(P_D P_H(Y^k) - Y^k). \quad (11)
\end{align}

4. Convergence analysis. In this section, we shall study the convergence of the proposed algorithms. Theorem 4.1 considers the case where $\inf_k \lambda_k > 0$, and $\sup_k \lambda_k < 2$. Proposition 3.1 shows that the choice of $\theta_k$ in Theorem 4.1 can be simplified in some special cases. Theorem 4.2 considers the case where $\sum_{k=0}^{\infty} \lambda_k(2 - \lambda_k) = \infty$. Analysis on Theorem 4.1, Proposition 4.1, and Theorem 4.2 are under the product space $H$ while Theorem 4.3, Proposition 4.2, and Theorem 4.4 are the respective counterparts of Theorem 4.1, Proposition 4.1, and Theorem 4.2 in the original space $\mathbb{R}^m$.

Theorem 4.1. Let $D \cap \mathcal{H} \neq \emptyset$, $\inf_k \lambda_k > 0$, $\sup_k \lambda_k < 2$. For each $k$, choose $\theta_k \in [0, \tilde{\theta}]$ with $\tilde{\theta} := \min\{\theta, 1/(k||X^k - X^{k-1}||)^2\}, \theta \in [0, 1)$. Then, for arbitrary $X^0, X^1$ in $D$, the sequence $\{X^k\}$ generated by Algorithm 3.2 converges to an element in $D \cap \mathcal{H}$.

Proof. For a given $Z \in D \cap \mathcal{H}$, define $\varphi_k := \frac{1}{2}||X^k - Z||^2$. The proof of this theorem is divided into three parts as follows.

Part I. We claim that

$$\varphi_{k+1} \leq \frac{1}{2}||Y^k - Z||^2 - \lambda_k(1 - \frac{\lambda_k}{2})||Y^k - P_D P_H(Y^k)||^2. \quad (12)$$

Observing, for arbitrary $A, B \in H$,

$$\langle A, B \rangle = \frac{1}{2}||A - B||^2 + \frac{1}{2}||A||^2 + \frac{1}{2}||B||^2.$$

We thus obtain

$$\langle X^k - Z, X^k - X^{k-1} \rangle = \frac{1}{2}||X^{k-1} - Z||^2 + \frac{1}{2}||X^k - Z||^2 + \frac{1}{2}||X^k - X^{k-1}||^2,$$

namely

$$\langle X^k - Z, X^k - X^{k-1} \rangle = \varphi_k - \varphi_{k-1} + \frac{1}{2}||X^k - X^{k-1}||^2. \quad (13)$$
By Proposition 3.1, we deduce

\[ \varphi_{k+1} = \frac{1}{2} \|X^{k+1} - Z\|^2 \]  
\[ = \frac{1}{2} \|(1 - \lambda_k)Y^k + \lambda_k P_D P_R(Y^k) - Z\|^2 \]  
\[ = \frac{1}{2} \|Y^k - Z\|^2 - \lambda_k \langle Y^k - Z, Y^k - P_D P_R(Y^k) \rangle + \frac{1}{2} \lambda_k^2 \|Y^k - P_D P_R(Y^k)\|^2 \]  
\[ \leq \frac{1}{2} \|Y^k - Z\|^2 - \lambda_k (1 - \frac{\lambda_k}{2}) \|Y^k - P_D P_R(Y^k)\|^2. \]  

Hence,

\[ \varphi_{k+1} \leq \frac{1}{2} \|Y^k - Z\|^2 - \lambda_k (1 - \frac{\lambda_k}{2}) \|Y^k - P_D P_R(Y^k)\|^2. \]  

Part II. We show that the sequence \(\{\|X^k - Z\|\}\) is convergent. By virtue of (10), we have

\[ \frac{1}{2} \|Y^k - Z\|^2 = \frac{1}{2} \|(X^k - Z) - \theta_k (X^{k-1} - X^k)\|^2 \]  
\[ = \frac{1}{2} \|(X^k - Z)\|^2 - \theta_k \langle X^k - Z, X^{k-1} - X^k \rangle \]  
\[ + \frac{\theta_k^2}{2} \|X^{k-1} - X^k\|^2 \]  
\[ = \varphi_k - \theta_k \langle X^k - Z, X^{k-1} - X^k \rangle + \frac{\theta_k^2}{2} \|X^{k-1} - X^k\|^2. \]  

Putting (16) into (12), it turns out that

\[ \varphi_{k+1} \leq \varphi_k + \theta_k (\varphi_k - \varphi_{k-1}) + \frac{\theta_k + \theta_k^2}{2} \|X^{k-1} - X^k\|^2. \]  

It follows from (13) that

\[ \frac{1}{2} \|Y^k - Z\|^2 = \varphi_k + \theta_k (\varphi_k - \varphi_{k-1}) + \theta_k + \frac{\theta_k^2}{2} \|X^{k-1} - X^k\|^2. \]  

Then, by \(\theta_k^2 \leq \theta_k\) (since \(\theta_k \in [0, 1]\)), (17) implies

\[ \varphi_{k+1} \leq \theta_k (\varphi_k - \varphi_{k-1}) + \theta_k \|X^{k-1} - X^k\|^2. \]  

From the selection of the parameter \(\theta_k\), we have \(\sum_{k \geq 0} \theta_k \|X^k - X^{k-1}\|^2 < \infty\), applying Lemma 2.1 to (18), we get that the sequence \(\{\|X^k - Z\|\}\) is convergent.

Part III. We prove that the sequence \(\{X^k\}\) generated by Algorithm 3.2 converges to an element in \(D \cap \mathcal{R}\). By (18) and Lemma 2.1, we get \(\sum_{k \geq 0} \theta_k \|X^k - Z\|^2 - \|X^{k-1} - \|\|^2 < \infty\), and from (17), we arrive at

\[ \lambda_k (2 - \lambda_k) \|Y^k - P_D P_R(Y^k)\|^2 \leq \|X^k - Z\|^2 - \|X^{k+1} - Z\|^2 \]  
\[ \quad + \theta_k (\|X^k - Z\|^2 - \|X^{k-1} - Z\|^2) \]  
\[ + 2 \theta_k \|X^{k-1} - X^k\|^2. \]  

This yields

\[ \sum_k \lambda_k (2 - \lambda_k) \|Y^k - P_D P_R(Y^k)\|^2 < \infty. \]
Proposition 4.1. Since $\inf\lambda_k > 0, \sup\lambda_k < 2$, we obtain
\[
\lim_k \|Y^k - PDP_k(Y^k)\| = 0.
\]
Let $\{X^{k_l}\}$ be a subsequence of $\{X^k\}$ such that $\{X^{k_l}\}$ converges to $X^*$. Evidently, $\lim_k \|Y^k - PDP_k(Y^k)\| = 0$, and $\{X^{k_l}\}$ converges to $X^*$ too. From Lemma 2.3 and Proposition 3.2, we obtain $X^* \in D \cap R$. Since $\{\|X^k - X^*\|\}$ is convergent, from Lemma 2.4, it is easy to get that $\lim_{k \to \infty} \|X^k - X^*\| = \lim_{1 \to \infty} \|X^{k_l} - X^*\| = 0$. This completes the proof. \hfill \Box

Proposition 4.1. Let $D \cap R \neq \emptyset$. Let $R_1 := \inf_{k \geq 0} \lambda_k > 0$ and $R_2 := \sup_{k \geq 0} \lambda_k < 2$. Assume that $\theta_k$ is nondecreasing, satisfying $0 \leq \theta_k \leq \theta$ for some $\theta \in [0, 1)$ such that
\[
0 < p(\theta) := \frac{1}{R_2 - 1/2} - \frac{1}{R_1 - 1/2} \theta - \frac{1}{R_2} \theta^2.
\]
Then
\[
\sum_{k=1}^{\infty} \|X^{k+1} - X^k\|^2 < \infty,
\]
and the sequence $\{X^k\}$ generated by Algorithm 3.2 converges to an element in $D \cap R$.

Proof. For any given $Z \in D \cap R$, define $\varphi_k := \frac{1}{2} \|X^k - Z\|^2$. From (10), we get that
\[
\lambda_k^2 \|PDP_k(Y^k) - Y^k\|^2 = \|X^{k+1} - Y^k\|^2 = \|X^{k+1} - X^k\|^2
\]
\[
2 \theta_k \langle(X^{k+1} - X^k, X^k - X^{k-1})\rangle + \theta_k^2 \|X^k - X^{k-1}\|^2
\]
\[
\geq (1 - \theta_k) \|X^{k+1} - X^k\|^2 - \theta_k (1 - \theta_k) \|X^k - X^{k-1}\|^2.
\]
it follows from (17) that
\[
\varphi_{k+1} \leq \varphi_k + \theta_k (\varphi_k - \varphi_{k-1}) + \left(\theta_k + \theta_k^2\right)/2 + (1/\lambda_k - 1/2) \theta_k (1 - \theta_k) \|X^k - X^{k-1}\|^2
\]
\[
- (1/\lambda_k - 1/2)(1 - \theta_k) \|X^{k+1} - X^k\|^2.
\]
It leads to
\[
\varphi_{k+1} - \theta_k \varphi_k \leq \varphi_k - \theta_k \varphi_{k-1} + \theta_k [1/\lambda_k + (1 - 1/\lambda_k) \theta_k] \|X^k - X^{k-1}\|^2
\]
\[
- (1/\lambda_k - 1/2)(1 - \theta_k) \|X^{k+1} - X^k\|^2.
\]
Setting
\[
\mu_k := \varphi_k - \theta_k \varphi_{k-1} + \theta_k [1/\lambda_k + (1 - 1/\lambda_k) \theta_k] \|X^k - X^{k-1}\|^2,
\]
since $\theta_{k+1} \geq \theta_k$, we obtain
\[
\mu_{k+1} \leq \mu_k + \left[\theta_{k+1}/\lambda_{k+1} + (1 - 1/\lambda_{k+1}) \theta_{k+1}^2 + (1/\lambda_k - 1/2) (\theta_k - 1)\right] \|X^{k+1} - X^k\|^2.
\]
On the other hand,
\[
\theta_{k+1}/\lambda_{k+1} + (1 - 1/\lambda_{k+1}) \theta_{k+1}^2 \leq \theta/R_1 + (1 - 1/R_2) \theta^2
\]
and
\[
(1/\lambda_k - 1/2)(\theta_k - 1) \leq (1/R_1 - 1/2) \theta - 1/R_2 + 1/2.
\]
Therefore,
\[
\mu_{k+1} \leq \mu_k - p(\theta) \|X^{k+1} - X^k\|^2,
\]
where $p(\theta)$ is given by (21). Since $p(\theta) > 0$, we have that $\{\mu_k\}$ is a non-increasing sequence. Obviously,
\[
\mu_k := \varphi_k - \theta_k \varphi_{k-1} + \theta_k [1/\lambda_k + (1 - 1/\lambda_k) \theta_k] \|X^k - X^{k-1}\|^2 \geq \varphi_k - \theta_k \varphi_{k-1},
\]
which implies
\[ \varphi_k \leq \theta \varphi_{k-1} + \mu_k \leq \theta \varphi_{k-1} + \mu_1. \]

Thus we have
\[ \varphi_k \leq \theta[\theta \varphi_{k-2} + \mu_1] + \mu_1 \leq \cdots \leq \theta^k \varphi_0 + \mu_1 \sum_{j=0}^{k-1} \theta^j \leq \theta^k \varphi_0 + \mu_1/(1 - \theta). \]

Furthermore, it follows that
\[ p(\theta) \sum_{j=0}^{k} \|X^j - X^j P_k\|^2 \leq \mu_1 - \mu_{k+1} \leq \mu_1 + \theta \varphi_k \leq \theta^{k+1} \varphi_0 + \mu_1/(1 - \theta). \]

This means that
\[ \sum_k \|X^k - X^{k-1}\| \leq 2\mu_1/((1 - \theta)p(\theta)). \]

The rest part of the proof is similar to that of Theorem 4.1, it is omitted for brevity. This completes the proof. \( \square \)

**Remark 1.** Suppose \( 1 \leq R_2 \leq 2 \). By (20), we know that \( p(0) = 1/R_2 - 1/2 > 0 \), then the quadratic polynomial \( p(\theta) \) has an unique positive root \( \theta^* > 0 \), and for all \( \theta \in [0, \theta^*), p(\theta) > 0 \). For instance, when \( \lambda_k \equiv 1 \), we get \( p(\theta) = 1/2 - (3/2)\theta \) and \( \theta^* = 1/3 \).

In the next theorem, we consider the convergence in the case of \( \sum_k \lambda_k(2 - \lambda_k) = \infty \).

**Theorem 4.2.** Let \( D \cap \mathbb{N} \neq \emptyset \), \( \sum_k \lambda_k(2 - \lambda_k) = \infty \). Choose \( \theta_k \), satisfying \( \theta_k \in [0, \bar{\theta}] \) with \( \bar{\theta} := \min\{\theta, 1/(k_1k_2)\} \) \( \in [0, 1) \). Then, for arbitrary \( X_0, X^1 \) in \( D \), the sequence \( \{X^k\} \) generated by Algorithm 3.2 converges to an element in \( D \cap \mathbb{N} \).

**Proof.** Following the same notations as in Theorem 4.1, by \( \sum_k \lambda_k(2 - \lambda_k) = \infty \) and by (19), we have
\[ \liminf_{k \to \infty} \|Y^k - P_D P_R(Y^k)\| = 0. \]

Let \( U = 2P_D P_R - I \). Evidently, \( U \) is non-expansive, \( I - P_D P_R = \frac{1}{2}(I - U) \). Since, \( X^{k+1} - Y^k = \lambda_k(P_D P_R(Y^k) - Y^k) \) and \( X^{k+1} = (1 - \lambda_k)Y^k + \lambda_k U(Y^k) \), we have
\[
2(Y^k - P_D P_R(Y^k)) = (I - U)(Y^k) = (X^k + (Y^k - X^k)) - U(Y^k)
\]
\[ = (1 - \frac{\lambda_{k-1}}{2})Y^k + \frac{\lambda_{k-1}}{2}U(Y^k) + U(Y^k), \]
moreover
\[
2(Y^k - P_D P_R(Y^k)) = (1 - \frac{\lambda_{k-1}}{2})Y^k + \frac{\lambda_{k-1}}{2}U(Y^k) + U(Y^k).
\]
Since \( \lambda_k \in (0, 2), U \) is non-expansive, then
\[
2\|Y^k - P_D P_R(Y^k)\| \leq (1 - \frac{\lambda_{k-1}}{2})\|Y^{k-1} - U(Y^{k-1})\|
\]
\[+ 2\|Y^k - X^k\| + \|Y^{k-1} - X^{k-1}\|. \]
By 2(\(I-P_DP_R\)) = \(I-U\) and \(X^k - Y^{k-1} = \lambda_{k-1}(P_DP_R(Y^{k-1}) - Y^{k-1})\), we get
\[
2\|Y^k - P_DP_R(Y^k)\| \leq (1 - \frac{\lambda_{k-1}}{2}) \times 2\|Y^{k-1} - P_DP_R(Y^{k-1})\| + 2\|Y^k - X^k\| + \lambda_{k-1}\|Y^k - P_DP_R(Y^{k-1})\|,
\]
which implies that
\[
\|Y^k - P_DP_R(Y^k)\| \leq \|Y^{k-1} - P_DP_R(Y^{k-1})\| + \|Y^k - X^k\| = \|Y^{k-1} - P_DP_R(Y^{k-1})\| + \theta_k\|X^k - X^{k-1}\|. \tag{22}
\]
By the choosing of \(\theta_k\), applying Lemma 2.2 to (22), we get that the sequence \(\{\|Y^k - P_DP_R(Y^k)\|\}\) is convergent, and since \(\lim \inf\|Y^k - P_DP_R(Y^k)\| = 0\), we obtain \(\lim\|Y^k - P_DP_R(Y^k)\| = 0\). The rest part of the proof is similar to that of Theorem 4.1. We omit for brevity and this completes the proof.

By transforming the sequence \(\{X^k\}\) generated by Algorithm 3.2 to the corresponding sequence \(\{x^k\}\) in the primal space \(\mathbb{R}^n\), we can derive similar convergence results of Algorithm 3.1 under appropriate assumptions. For simplicity in description, we only state the results while their proofs can be achieved in the same manner as Theorems 4.1-4.2 and Proposition 4.1.

**Theorem 4.3.** Assume \(C \neq \emptyset\), \(x^0, x^1 \in \mathbb{R}^n\), \(\inf_k \lambda_k > 0, \sup_k \lambda_k < 2\). Choose \(\theta_k\) satisfying \(\{\theta_k\} \in [0, \tilde{\theta}]\) with \(\tilde{\theta} := \min\{\theta, 1/(k\|x^k - x^{k-1}\|)^2\}, \theta \in [0, 1]\). Then, the sequence \(\{x^k\}\) generated by Algorithm 3.1 converges to a point in \(C\).

**Proposition 4.2.** Let \(C \neq \emptyset\) and let the relaxed parameter \(\lambda_k\) satisfy: \(R_1 := \inf_{k \geq 0} \lambda_k > 0\) and \(R_2 := \sup_{k \geq 0} \lambda_k < 2\). \(\{\theta_k\}\) is a nondecreasing sequence, and satisfying \(0 \leq \theta_k \leq \theta\) for some \(\theta \in [0, 1]\) such that
\[
0 < p(\theta) := 1/R_2 - 1/2 - (2/R_1 - 1/2)\theta - (1 - 1/R_2)\theta^2.
\]
Then \(\sum_{k=1}^{\infty} \|x^{k+1} - x^k\|^2 < \infty\), and the sequence \(\{x^k\}\) generated by Algorithm 3.1 converges to a point in \(C\).

**Theorem 4.4.** Assume \(C \neq \emptyset\), \(x^0, x^1 \in \mathbb{R}^n\), \(\sum_k \lambda_k(2 - \lambda_k) = \infty\). Choose \(\theta_k\) satisfying \(\{\theta_k\} \in [0, \tilde{\theta}]\) with \(\tilde{\theta} := \min\{\theta, 1/\max((k\|x^k - x^{k-1}\|)^2, k^2\|x^k - x^{k-1}\|)\}, \theta \in [0, 1]\). Then, the sequence \(\{x^k\}\) generated by Algorithm 3.1 converges to a point in \(C\).

5. **Numerical experiments.** To illustrate the proposed approach for solving CFP, we have carried out numerical tests on three examples. We also compare the numerical results with commonly-adopted extrapolated algorithm for solving the CFP. In this section, we report some preliminary numerical results. The tests are carried out by implementing codes in Matlab 7.8.0 installed in a PC with Windows XP Operating System.

We briefly state the extrapolated algorithm for solving the CFP in the following. Given an arbitrary initiative point \(x^0\), the iterative step is as follows
\[
x^{k+1} = x^k + \lambda_k\left(\frac{\sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x^k) - x^k}{\sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x^k) - x^k}\right)^2 \left(\frac{\sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x^k) - x^k}{\sum_{i=1}^{m} \frac{1}{m} P_{C_i}(x^k) - x^k}\right),
\]
where \(\lambda_k \in (0, 2)\).
5.1. **Three examples.** We conduct numerical experiments on three man-made examples of convex feasibility problem, the dimension of the problem ranging from 2 to 120.

**Example 5.1.** Let
\[ C_1 = \{ x \in \mathbb{R}^2 | f_1(x) = (x_1 - 3)^2 + (x_2 - 3)^2 - 4 \leq 0 \}, \]
\[ C_2 = \{ x \in \mathbb{R}^2 | f_2(x) = x_1^2 - 9 + x_2^2 \leq 0 \}. \]

Find \( x \in C_1 \cap C_2 \).

**Example 5.2.** Let
\[ C_1 = \{ x \in \mathbb{R}^2 | f_1(x) = x_1 + x_2 - 4 \leq 0 \}, \]
\[ C_2 = \{ x \in \mathbb{R}^2 | f_2(x) = 3x_1 + x_2 \leq 0 \}, \]
\[ C_3 = \{ x \in \mathbb{R}^2 | f_3(x) = 2x_1 - 3x_2 - 2 \leq 0 \}, \]
\[ C_4 = \{ x \in \mathbb{R}^2 | f_4(x) = x_1 + 2x_2 - 1 \leq 0 \}. \]

Find \( x \in C_1 \cap C_2 \cap C_3 \cap C_4 \).

**Example 5.3.** Let
\[ C_1 = \{ x \in \mathbb{R}^{120} | \sum_{j=1}^{120} (j^2 - 1)x_j \leq 1 \}, \]
\[ C_2 = \{ x \in \mathbb{R}^{120} | \sum_{j=1}^{120} jx_j^2 \leq 2 \}, \]
\[ C_3 = \{ x \in \mathbb{R}^{120} | \sum_{j=1}^{120} (-1)^j x_j \leq 1 \}. \]

Find \( x \in C_1 \cap C_2 \cap C_3 \). In this example, we consider three scenarios on the choice of initial point \( x^0 \).

- Case 1: \( x^0 = (1, 1, \cdots, 1) \in \mathbb{R}^{120} \);
- Case 2: \( x^0 = (-1, -2, -3, \cdots, -120) \in \mathbb{R}^{120} \);
- Case 3: \( x^0 = (-10, -20, -30, \cdots, -1200) \in \mathbb{R}^{120} \).

5.2. **Numerical results.** Our results of numerical experiment are listed in the following Tables 1-3, where EA denotes the extrapolated algorithm, IPPA denotes the inertial parallel projection algorithm, “Iter.” and “Sec.” denote the number of iterations and the CPU time in seconds, respectively, \( p(\varepsilon) \) denotes the sum of distance of the approximate solution to the given sets, and \( x^* \) denotes the approximate solution. The results of Example 5.1-5.2 are respectively reported in Tables 1-2 with various different \( \lambda_k \). Table 3 shows the numbers of iterations needed to get the approximation solutions by the EA and the IPPA for Example 5.3 with \( \lambda_k = 1 \). For comparison, we use the same initiative points \( x^0 \) and \( x^1 \), where \( x^1 \) is generated by the EA.

In these tests, we take the parameters as follows: \( \theta_k = \frac{\theta_k}{2} \) with
\[ \hat{\theta}_k := \min \left\{ 0.8, \frac{1}{\max\{k^2\|x^k - x^{k-1}\|, k^2\|x^k - x^{k-1}\|\}} \right\}, \]
k = 1, 2, \cdots. The stopping criterion is \( p(\varepsilon) \leq \varepsilon = 10^{-7} \).
| starting points | EA $\lambda_k = 0.6$ | IPPA $\lambda_k = 0.6$ | EA $\lambda_k = 1$ | IPPA $\lambda_k = 1$ | EA $\lambda_k = 1.6$ | IPPA $\lambda_k = 1.6$ |
|----------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| $x^0 = (0, 8)$ | $x^1 = (0.3917, 6.3070)$ | $x^1 = (0.6618, 5.1783)$ | $x^1 = (1.0588, 3.4853)$ | $x^1 = (1.0588, 3.4853)$ | $x^1 = (0.3917, 6.3070)$ | $x^1 = (0.6618, 5.1783)$ |
| Iter. = 82 | Iter. = 12 | Iter. = 41 | Iter. = 8 | Iter. = 12 | Iter. = 4 |
| Sec. = 0.079 | Sec. = 0.018 | Sec. = 0.052 | Sec. = 0.008 | Sec. = 0.024 | Sec. = 0.003 |
| $x^* = (1.0094, 2.7791)$ | $x^* = (1.0086, 2.8245)$ | $x^* = (1.0231, 2.7979)$ | $x^* = (1.0590, 2.7028)$ | $x^* = (1.0250, 2.8000)$ |
| $p(\varepsilon) = 9.5040 \times 10^{-7}$ | $p(\varepsilon) = 8.1489 \times 10^{-7}$ | $p(\varepsilon) = 8.8463 \times 10^{-7}$ | $p(\varepsilon) = 5.6308 \times 10^{-7}$ | $p(\varepsilon) = 2.7006 \times 10^{-7}$ |
| $x^0 = (6, 6)$ | $x^1 = (4.8625, 4.8625)$ | $x^1 = (4.1042, 4.1042)$ | $x^1 = (4.1042, 4.1042)$ | $x^1 = (2.9667, 2.9667)$ |
| Iter. = 53 | Iter. = 9 | Iter. = 31 | Iter. = 5 | Iter. = 15 | Iter. = 2 |
| Sec. = 0.072 | Sec. = 0.010 | Sec. = 0.043 | Sec. = 0.006 | Sec. = 0.027 | Sec. = 0.002 |
| $x^* = (2.1232, 2.1232)$ | $x^* = (2.1145, 2.1230)$ | $x^* = (2.1230, 2.0886)$ | $x^* = (2.1230, 2.1230)$ | $x^* = (2.1098, 2.1098)$ |
| $p(\varepsilon) = 8.4796 \times 10^{-7}$ | $p(\varepsilon) = 6.9478 \times 10^{-7}$ | $p(\varepsilon) = 6.8137 \times 10^{-7}$ | $p(\varepsilon) = 0 \times 10^{-7}$ | $p(\varepsilon) = 0 \times 10^{-7}$ |

Table 1. Numerical Results for Example 5.1
Table 2. Numerical Results for Example 5.2

| starting points | EA $\lambda_k = 0.8$ | IPPA $\lambda_k = 0.8$ | EA $\lambda_k = 1$ | IPPA $\lambda_k = 1$ | EA $\lambda_k = 1.8$ | IPPA $\lambda_k = 1.8$ |
|-----------------|----------------------|------------------------|-------------------|----------------------|----------------------|----------------------|
| $x^0 = (10, 10)$| $x^1 = (-27.6400, -59.6800)$ | $x^1 = (-27.6400, -59.6800)$ | $x^1 = (37.0500, -77.1000)$ | $x^1 = (37.0500, -77.1000)$ | $x^1 = (37.0500, -77.1000)$ | $x^1 = (37.0500, -77.1000)$ |
| Iter. = 42 | Iter. = 25 | Iter. = 36 | Iter. = 18 | Iter. = 16 | Iter. = 11 |
| Sec. = 0.047 | Sec. = 0.045 | Sec. = 0.052 | Sec. = 0.040 | Sec. = 0.040 | Sec. = 0.011 |
| $x^* = (-46.3718, -31.5823)$ | $x^* = (-60.9265, -41.2853)$ | $x^* = (-60.9333, -41.2833)$ | $x^* = (-119.1449, -80.0977)$ | | |
| $p(\varepsilon) = 8.8619 \times 10^{-7}$ | $p(\varepsilon) = 7.4880 \times 10^{-7}$ | $p(\varepsilon) = 5.9224 \times 10^{-7}$ | $p(\varepsilon) = 8.7401 \times 10^{-7}$ | |

| $x^0 = (20, 0)$| $x^1 = (-0.8092, -29.9262)$ | $x^1 = (-0.8092, -29.9262)$ | $x^1 = (-6.0115, -37.4077)$ | $x^1 = (-6.0115, -37.4077)$ | $x^1 = (-6.0115, -37.4077)$ | $x^1 = (-6.0115, -37.4077)$ |
| Iter. = 44 | Iter. = 23 | Iter. = 32 | Iter. = 15 | Iter. = 16 | Iter. = 9 |
| Sec. = 0.047 | Sec. = 0.043 | Sec. = 0.051 | Sec. = 0.030 | Sec. = 0.040 | Sec. = 0.006 |
| $x^* = (-14.0642, -10.0438)$ | $x^* = (-21.1188, -14.7469)$ | $x^* = (-21.1284, -14.7509)$ | $x^* = (-49.3372, -33.5592)$ | | |
| $p(\varepsilon) = 6.9334 \times 10^{-7}$ | $p(\varepsilon) = 3.4312 \times 10^{-7}$ | $p(\varepsilon) = 7.4936 \times 10^{-7}$ | $p(\varepsilon) = 7.4124 \times 10^{-7}$ | |
From Tables 1-2, we can see the IPPA converges faster than the EA. In Example 5.1-5.2, we can get the approximate solution in a finite number of iterations by using the IPPA, while the EA can not get any approximate solution after finitely many iterations. As shown in Table 3, the number of iterations needed to get the approximate solution in Example 5.3 by using the EA is much larger than that by using the IPPA.

**Acknowledgments.** The authors would like to thank the two anonymous reviewers for their constructive suggestions and comments, which help to improve the presentation of the paper.

**REFERENCES**

[1] H. H. Bauschke and J. M. Borwein, *On projection algorithms for solving convex feasibility problems*, SIAM Rev., **38** (1996), 367–426.

[2] H. H. Bauschke, J. M. Borwein, and A. S. Lewis, *The method of cyclic projections for closed convex sets in Hilbert space*, in *Proceedings of the Special Session on Optimization and Nonlinear Analysis*, 1995.

[3] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1994.

[4] Y. Censor and A. Lent, *Cyclic subgradient projections*, Math. Program., **24** (1982), 233–235.

[5] Y. Censor, *Parallel application of block iterative methods in medical imaging and radiation therapy*, Math. Program., **42** (1998), 307–325.

[6] J. W. Chinneck, *The constraint consensus method for finding approximately feasible points in nonlinear programs*, INFORMS J. Comput., **16** (2004), 255–265.

[7] P. L. Combettes and S. G. Kruk, *Extrapolation algorithm for affine-convex feasibility problems*, Numer. Algorithms, **41** (2006), 239–274.

[8] Y. Dang and Y. Gao, *Non-monotonous accelerated parallel subgradient projection algorithm for convex feasibility problem*, Optimization, **63** (2014), 571–584.

[9] F. Deutsch, *The method of alternating orthogonal projections*, Approximation Theory, *Spline Functions and Applications*, Kluwer Academic Publishers, Dordrecht, 1992.

[10] Y. Gao, *Viability criteria for differential inclusions*, J. Syst. Sci. Complex., **24** (2011), 825–834.

[11] U. Garcia-Palomares, *Parallel projected aggregation methods for solving the convex feasibility problem*, SIAM J. Optim., **3** (1993), 882–900.

[12] U. M. Garcia-Palomares and F. J. Gonzalez-Castano, *Incomplete projection algorithms for solving the convex feasibility problem*, Numer. Algorithms, **18** (1998), 177–193.

[13] K. Goebel and W. A. Kirk, *Topics in Metric Fixed Point Theory*, Cambridge Studies in Advanced Mathematics 28, Cambridge University Press, Cambridge, 1990.

[14] N. I. M. Gould, *How good are projection methods for convex feasibility problems*, Comput. Optim. Appl., **40** (2008), 1–12.

[15] G. T. Herman, *Image Reconstruction From Projections: The Fundamentals of Computerized Tomography*, Academic Press, New York, 1980.

[16] K. C. Kiwiel, *Block-iterative surrogate projection methods for convex feasibility problem*, Linear Algebra Appl., **215** (1995), 225–260.

[17] L. Li and Y. Gao, *Approximate subgradient projection algorithm for a convex feasibility problem*, J. Syst. Eng. Electron., **21** (2010), 527–530.
T. Lucio, A parallel subgradient projections method for the convex feasibility problem, *J. Comput. Appl. Math.*, 18 (1987), 307–320.

P. E. Mainge, Convergence theorem for inertial KM-type algorithms, *J. Comput. Appl. Math.*, 219 (2008), 223–236.

M. Moudafi, Convergence of a splitting inertial proximal method for monotone operators, *J. Comput. Appl. Math.*, 155 (2008), 447–454.

A. Moudafi and E. Elisabeth, An approximate inertial proximal method using enlargement of a maximal monotone operator, *Int. J. Pure Appl. Math.*, 5 (2003), 283–299.

Z. Opial, Weak convergence of the sequence of successive approximations for nonexpansive mappings, *Bull. Am. Math. Soc.*, 73 (1967), 591–597.

G. Pierra, Decomposition through formalization in a product space, *Math. Program.*, 28 (1984), 96–115.

B. T. Polyak, Some methods of speeding up the convergence of iteration method, *Zh. Vych. Math.*, 4 (1964), 791–803.

Received January 2016; 1st revision October 2016; final revision November 2016.

E-mail address: jgdyz@163.com
E-mail address: fanwen_meng@nhg.com.sg
E-mail address: jie.sun@curtin.edu.au