Dang and Xue *Journal of Inequalities and Applications* (2015) 2015:47
DOI 10.1186/s13660-015-0576-9

Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

Iterative process for solving a multiple-set split feasibility problem

Yazheng Dang[1,2*†] and Zhonghui Xue[3†]

*Correspondence: jgdyz@163.com
[1]School of Management, University of Shanghai for Science and Technology, Shanghai, 200093, China
[2]College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan 454000, China
†Equal contributors
Full list of author information is available at the end of the article

Abstract
This paper deals with a variant relaxed CQ algorithm by using a new searching direction, which is not the gradient of a corresponding function. The strategy is to intend to improve the convergence. Its convergence is proved under some suitable conditions. Numerical results illustrate that our variant relaxed CQ algorithm converges more quickly than the existing algorithms.

MSC: 47H05; 47J05; 47J25

Keywords: multiple-set split feasibility problem; subgradient; accelerated iterative algorithm; convergence

1 Introduction

The multiple-set split feasibility problem (MSSFP) is to find a point contained in the intersection of a family of closed convex sets in one space such that its image under a linear transformation is contained in the intersection of another family of closed convex sets in the image space. Formally, given nonempty closed convex sets $C_i \subseteq \Re^N$, $i = 1, 2, \ldots, t$, in the N-dimensional Euclidean space $\Re^N$ and nonempty closed convex sets $Q_j \subseteq \Re^M$, $j = 1, 2, \ldots, r$, and an $M \times N$ real matrix A, the MSSFP is to find a point x such that

$$x \in C = \bigcap_{i=1}^{t} C_i, \qquad Ax \in Q = \bigcap_{j=1}^{r} Q_j. \tag{1.1}$$

Such MSSFP, formulated in [1], arises in the field of intensity-modulated radiation therapy (IMRT) when one attempts to describe physical dose constrains and equivalent uniform dose (EUD) constraints within a single model, see [2, 3]. Specially, when $t = r = 1$, the problem reduces to the two-set split feasibility problem (abbreviated as SFP), which is to find a point $x \in C$ such that $Ax \in Q$ (see [4–6]).

For solving the MSSFP, Censor *et al.* in [1] introduced a proximity function $p(x)$ to measure the aggregate distance of a point to all sets. The function $p(x)$ is defined as

$$p(x) := \frac{1}{2}\sum_{i=1}^{t} \alpha_i \left\| x - P_{C_i}(x) \right\|^2 + \frac{1}{2}\sum_{j=1}^{r} \beta_j \left\| Ax - P_{Q_j}(Ax) \right\|^2,$$

where $\alpha_i > 0$, $\beta_j > 0$ for all i and j, respectively, and $\sum_{i=1}^{t}\alpha_i + \sum_{j=1}^{r}\beta_j = 1$. Then they proposed a projection algorithm as follows:

$$x^{k+1} = P_\Omega\bigl(x^k - \gamma\nabla p\bigl(x^k\bigr)\bigr),$$

where $\Omega \subset \Re^N$ is an auxiliary set, x^k is the current iterative point. $0 < \gamma < 2/L$ with $L = \sum_{i=1}^{t}\alpha_i + \rho(A^TA)\sum_{j=1}^{r}\beta_j$ and $\rho(A^TA)$ is the spectral radius of A^TA. Subsequently, many methods have been developed for solving the MSSFP [7–14], while most of these algorithms aimed at minimizing the proximity function $p(x)$ and used its gradient ∇p.

Different from most of the existing methods, in this paper, we construct a new searching direction, which is not the gradient ∇p. And this difference causes a very different way of analysis. Moreover, some preliminary numerical experiments show that our new method converges faster than most existing methods.

The paper is organized as follows. Section 2 reviews some preliminaries. Section 3 gives a variant relaxed projection algorithm and shows its convergence. Section 4 gives some numerical experiments. Some conclusions are drawn in Section 5.

2 Preliminaries

Throughout the rest of the paper, I denotes the identity operator, $\mathrm{Fix}(T)$ denotes the set of the fixed points of an operator T, *i.e.*, $\mathrm{Fix}(T) := \{x \mid x = T(x)\}$.

Let T be a mapping from $\aleph \subseteq \Re^N$ into $\Re^N$. T is called co-coercive on $\aleph$ with modulus $\mu > 0$ if

$$\bigl\langle T(x) - T(y), x - y\bigr\rangle \geq \mu\bigl\|T(x) - T(y)\bigr\|^2, \quad \forall x, y \in \aleph;$$

it is called Lipschitz continuous on $\aleph$ for constant $L > 0$ if

$$\bigl\|T(x) - T(y)\bigr\| \leq L\|x - y\|, \quad x, y \in \aleph;$$

it is called monotone on $\aleph$ if

$$\bigl\langle T(x) - T(y), x - y\bigr\rangle \geq 0, \quad \forall x, y \in \aleph.$$

It is obvious that the co-coercivity (with modulus μ) implies the Lipschitz continuity (with constant $1/\mu$) and monotonicity.

Let S be a nonempty closed convex subset of $\Re^N$. Denote by P_S the orthogonal projection onto S; that is,

$$P_S(x) = \arg\min_{y\in\Re^N}\|x - y\|,$$

over all $x \in S$.

It is well known that the orthogonal projection operator P_S, for any $x, y \in \Re^N$ and any $z \in S$, is characterized by the inequalities [15]

$$\bigl\langle x - P_S(x), z - P_S(x)\bigr\rangle \leq 0 \tag{2.1}$$

and

$$\left\|P_S(x)-z\right\|^2 \le \|x-z\|^2 - \left\|P_S(x)-x\right\|^2. \tag{2.2}$$

Recall the notion of the subdifferential for an appropriate convex function.

Definition 2.1 Let $f:\Re^N \to \Re$ be convex. The subdifferential of f at x is defined as

$$\partial f(x) = \left\{\xi \in \Re^N \mid f(y) \ge f(x) + \langle \xi, y-x\rangle, \forall y \in \Re^N\right\}. \tag{2.3}$$

Evidently, an element of $\partial f(x)$ is said to be a subgradient.

Lemma 2.1 [16] *An operator T is co-coercive with modulus* 1 *if and only if the operator $I-T$ is co-coercive with modulus* 1, *where I denotes the identity operator.*

It is easy to see from the above lemmas that the orthogonal projection operators are monotone, co-coercive with modulus 1, and the operator $I-P_Q$ is also co-coercive with modulus 1.

3 Algorithm and its convergence

3.1 The variant relaxed-CQ algorithm

As in [12], we suppose that the following conditions are satisfied:

(1) The solution set of the MSSFP is nonempty.

(2) The sets C_i, $i=1,3,\dots,t$, are denoted as

$$C_i = \left\{x \in \Re^N \mid c_i(x) \le 0\right\}, \tag{3.1}$$

where $c_i:\Re^N \to \Re$, $i=1,2,\dots,t$, are appropriately convex and C_i, $i=1,2,\dots,t$, are nonempty.

The set Q_j, $j=1,2,\dots,r$, is denoted as

$$Q_j = \left\{y \in \Re^M \mid q_j(y) \le 0\right\}, \tag{3.2}$$

where $q_j:\Re^M \to \Re$, $j=1,2,\dots,r$, are appropriately convex and Q_j, $j=1,2,\dots,r$, are nonempty.

(3) For any $x \in \Re^N$, at least one subgradient $\xi_i \in \partial c_i(x)$ can be calculated.

For any $y \in \Re^M$, at least one subgradient $\eta_j \in \partial q_j(y)$ can be computed.

Now, we define the following half-spaces at point x^k:

$$C_i^k = \left\{x \in \Re^N \mid c_i\left(x^k\right) + \left\langle \xi_i^k, x-x^k\right\rangle \le 0\right\}, \tag{3.3}$$

where ξ_i^k is an element in $\partial c_i(x^k)$ for $i=1,2,\dots,t$, and

$$Q_j^k = \left\{y \in \Re^M \mid q_j\left(Ax^k\right) + \left\langle \eta_j^k, y-Ax^k\right\rangle \le 0\right\}, \tag{3.4}$$

where η_j^k is an element in $\partial q_j(Ax^k)$ for $j=1,2,\dots,r$.

By the definition of subgradient, it is clear that the half-spaces C_i^k and Q_j^k contain C_i and Q_j, $i = 1, 2, \dots, r$; $j = 1, 2, \dots, t$, respectively. Due to the specific form of C_i^k and Q_j^k, the orthogonal projections onto C_i^k and Q_j^k, $i = 1, 2, \dots, r$; $j = 1, 2, \dots, t$, may be computed directly, see [15].

Now, we give the variant relaxed CQ algorithm.

Algorithm 3.1 Given $\alpha_i > 0$ and $\beta_j \geq 0$ such that $\sum_{i=1}^t \alpha_i = 1$, $\sum_{j=1}^r \beta_j = 1$, $\gamma \in (0, \frac{1}{\rho(A^T A)})$, $t_k \in (0, 2)$.

For an arbitrary initial point, $x^0 \in \Re^n$ is the current point. Define a mapping $F_k : \Re^N \to \Re^N$ as

$$F_k(x) = \sum_{j=1}^r \beta_j A^T (I - P_{Q_j^k}) Ax. \tag{3.5}$$

For $k = 0, 1, 2, \dots$, compute

$$y^k = \sum_{i=1}^t \alpha_i P_{C_i^k}\bigl(x^k - \gamma F_k(x^k)\bigr). \tag{3.6}$$

Let

$$d^k = x^k - y^k + \gamma\bigl(F_k(y^k) - F_k(x^k)\bigr). \tag{3.7}$$

Set

$$x^{k+1} = x^k - t_k d^k. \tag{3.8}$$

In this algorithm, we can take $\|d^k\| < \varepsilon$ for some given precision as the stopping criterion. And we apply y^k and F_k to construct the searching direction d^k. The choice of a new searching direction leads to quite different in establishing the convergence result of Algorithm 3.1.

By Lemma 8.1 in [17], the operator $A^T(I - P_{Q_j^k})A$ is $1/\rho(A^T A)$-inverse strongly monotone ($1/\rho(A^T A)$-ism) or co-coercive with modulus $1/\rho(A^T A)$ and Lipschitz continuous with $\rho(A^T A)$.

3.2 Convergence of the variant relaxed-CQ algorithm

In this subsection, we establish the convergence of Algorithm 3.1.

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

Lemma 3.1 [18, 19] *Suppose that $f : \Re^N \to \Re$ is convex. Then its subdifferential is uniformly bounded on any bounded subsets of $\Re^N$.*

Lemma 3.2 *Assume that z is an arbitrary solution of the MSSFP (i.e., $z \in SOL(MSSFP)$) and $u \in \Re^N$, it holds that*

$$\bigl\langle F_k(u), u - z\bigr\rangle \geq \sum_{j=1}^r \beta_j \bigl\|(I - P_{Q_j^k})(Au)\bigr\|^2 \geq 0. \tag{3.9}$$

Proof If $z \in SOL(MSSFP)$, then $Az \in Q_j \subset Q_j^k$ for all $j = 1, \dots, r$, thus $F_k(z) = 0$, we have known that the mappings $I - P_{Q_j^k}$ are co-coercive with modulus 1, it follows that

$$\begin{aligned}\langle F_k(u), u - z\rangle &= \langle F_k(u) - F_k(z), u - z\rangle \\ &= \sum_{j=1}^{r} \beta_j \langle (A^T(I - P_{Q_j^k})Au - A^T(I - P_{Q_j^k})Az, u - z\rangle \\ &= \sum_{j=1}^{r} \beta_j \langle (I - P_{Q_j^k})Au - (I - P_{Q_j^k})Az, Au - Az\rangle \\ &\geq \sum_{j=1}^{r} \beta_j \| (I - P_{Q_j^k})Au - (I - P_{Q_j^k})Az \|^2 \\ &= \sum_{j=1}^{r} \beta_j \| (I - P_{Q_j^k})Au \|^2. \end{aligned}$$

□

Now, we state the convergence of Algorithm 3.1.

Theorem 3.1 *Assume that the set of solutions of the constrained multiple-set split feasibility problem is nonempty. Then any sequence* $\{x^k\}_{k=0}^{\infty}$ *generated by Algorithm* 3.1 *converges to a solution of the multiple-set split feasibility problem.*

Proof Let z be a solution of MSSFP. Since $C_i \subset C_{i,k}$, $Q_j \subset Q_j^k$, then $z = P_{C_i}z = P_{C_{i,k}}z$ and $Az = P_{Q_j}Az = P_{Q_{j,k}}Az$ for all i and j and therefore $F_k(z) = 0$. By Algorithm 3.1, we have

$$\|x^{k+1} - z\|^2 = \|x^k - t_k d^k - z\|^2 = \|x^k - z\|^2 - 2t_k\langle d^k, x^k - z\rangle + t_k^2\|d^k\|^2,$$

hence

$$\|x^{k+1} - z\|^2 = \|x^k - z\|^2 - 2t_k\langle d^k, y^k - z\rangle - 2t_k\langle d^k, x^k - y^k\rangle + t_k^2\|d^k\|^2. \tag{3.10}$$

By (3.7) we have

$$\langle d^k, y^k - z\rangle = \langle x^k - \gamma F_k(x^k) - y^k, y^k - z\rangle + \gamma\langle F_k(y^k), y^k - z\rangle. \tag{3.11}$$

From Lemma 3.2, we obtain

$$\langle F_k(y^k), y^k - z\rangle \geq \sum_{j=1}^{r} \beta_j \| (I - P_{Q_j^k})Ay^k \|^2 \geq 0. \tag{3.12}$$

Let $z^k = x^k - \gamma F_k(x^k)$. For that $\sum_{i=1}^{t} \alpha_i = 1$, we obtain from (3.5) that

$$\begin{aligned}\langle x^k - \gamma F_k(x^k) - y^k, y^k - z\rangle &= \langle z^k - y^k, y^k - z\rangle \\ &= \left\langle \sum_{i=1}^{t} \alpha_i\bigl(z^k - P_{C_i^k}(z^k)\bigr), \sum_{i=1}^{t} \alpha_i P_{C_i^k}(z^k) - z \right\rangle \\ &= \sum_{i=1}^{t}\sum_{h=1}^{t} \alpha_i\alpha_h\langle z^k - P_{C_i^k}(z^k), P_{C_h^k}(z^k) - z\rangle.\end{aligned}$$

If $i = h$, then $\langle z^k - P_{C_i^k}(z^k), P_{C_h^k}(z^k) - z\rangle \geq 0$, since $z \in C_i \subset C_i^k$ by Lemma 2.1. Otherwise, if $i \neq h$, we have

$$\begin{aligned}
&\alpha_i\alpha_h\langle z^k - P_{C_i^k}(z^k), P_{C_h^k}(z^k) - z\rangle + \alpha_h\alpha_i\langle z^k - P_{C_h^k}(z^k), P_{C_i^k}(z^k) - z\rangle \\
&\quad = \alpha_i\alpha_h\big[\langle z^k - P_{C_i^k}(z^k), P_{C_i^k}(z^k) - z\rangle + \langle z^k - P_{C_i^k}(z^k), P_{C_h^k}(z^k) - P_{C_i^k}(z^k)\rangle\big] \\
&\qquad + \alpha_h\alpha_i\big[\langle z^k - P_{C_h^k}(z^k), P_{C_h^k}(z^k) - z\rangle + \langle z^k - P_{C_h^k}(z^k), P_{C_i^k}(z^k) - P_{C_h^k}(z^k)\rangle\big] \\
&\quad \geq \alpha_i\alpha_h\big\|P_{C_i^k}(z^k) - P_{C_h^k}(z^k)\big\|^2.
\end{aligned}$$

It means

$$\langle x^k - \gamma_k F_k(x^k) - y^k, y^k - z\rangle \geq \sum_{i<h}\alpha_i\alpha_h\big\|P_{C_i^k}(z^k) - P_{C_h^k}(z^k)\big\|^2 \geq 0. \tag{3.13}$$

By combining (3.12) and (3.13) with (3.11), we obtain

$$\langle d^k, y^k - z\rangle \geq \sum_{i<h}\alpha_i\alpha_h\big\|P_{C_i^k}(z^k) - P_{C_h^k}(z^k)\big\|^2 + \gamma\sum_{j=1}^r\beta_j\big\|(I - P_{Q_j^k})Ay^k\big\|^2 \geq 0. \tag{3.14}$$

On the other hand, by definition of d^k in (3.7), we have

$$\begin{aligned}
\langle d^k, x^k - y^k\rangle &= \langle d^k, x^k - y^k + \gamma F_k(y^k) - \gamma_k F_k(x^k)\rangle + \gamma\langle d^k, F_k(x^k) - F_k(y^k)\rangle \\
&= \|d^k\|^2 + \gamma\langle x^k - y^k + \gamma F_k(y^k) - \gamma F_k(x^k), F_k(x^k) - F_k(y^k)\rangle \\
&= \|d^k\|^2 + \gamma\langle x^k - y^k, F_k(x^k) - F_k(y^k)\rangle - \gamma^2\|F_k(x^k) - F_k(y^k)\|^2.
\end{aligned}$$

From Lemma 3.1, we arrive at $\langle x^k - y^k, F_k(x^k) - F_k(y^k)\rangle \geq 1/\rho(A^TA)\|F_k(x^k) - F_k(y^k)\|^2$ for all k, hence

$$\begin{aligned}
\langle d^k, x^k - y^k\rangle &\geq \|d^k\|^2 + (\gamma - \gamma^2\rho(A^TA))\langle x^k - y^k, F_k(x^k) - F_k(y^k)\rangle \\
&= \|d^k\|^2 + \gamma(1 - \gamma\rho(A^TA))\sum_{j=1}^r\beta_j\langle Ax^k - Ay^k, \\
&\qquad (I - P_{Q_j^k})Ax^k - (I - P_{Q_j^k})Ay^k\rangle.
\end{aligned}$$

Furthermore, from the 1-co-coercivity of $I - P_{Q_j^k}$, we have

$$\langle d^k, x^k - y^k\rangle \geq \|d^k\|^2 + \gamma(1 - \gamma\rho(A^TA))\sum_{j=1}^r\beta_j\big\|(I - P_{Q_j^k})Ax^k - (I - P_{Q_j^k})Ay^k\big\|^2. \tag{3.15}$$

From (3.14), (3.15) and (3.10), we have

$$\begin{aligned}
\|x^{k+1} - z\|^2 &= \|x^k - z\|^2 - 2t_k\langle d^k, y^k - z\rangle - 2t_k\langle d_k, x^k - y^k\rangle + t_k^2\|d^k\|^2, \\
&\leq \|x^k - z\|^2 - t_k(2 - t_k)\|d^k\|^2
\end{aligned}$$

$$
\begin{aligned}
&- 2t_k\gamma(1-\gamma\rho(A^TA)\sum_{j=1}^{r}\beta_j\left\|(I-P_{Q_j^k})Ax^k-(I-P_{Q_j^k})Ay^k\right\|^2 \\
&- 2t_k\Bigg[\sum_{i<h}\alpha_i\alpha_h\left\|P_{C_i^k}(z^k)-P_{C_h^k}(z^k)\right\|^2 \\
&+ \gamma\sum_{j=1}^{r}\beta_j\left\|(I-P_{Q_j^k})Ay^k\right\|^2\Bigg].
\end{aligned}
\tag{3.16}
$$

Since $t_k \in (0,2)$, $\gamma \in (0, \frac{1}{\rho(A^TA)})$ in the algorithm, we conclude that the sequence $\{\|x^k - z\|\}$ is monotonously nonincreasing and convergent and $\{x^k\}$ is bounded. We have shown that the sequence $\{\|x^k - z\|\}$ is monotonically decreasing and bounded, therefore there exists the limit

$$\lim_{k\to\infty}\left\|x^k - z\right\| = d, \tag{3.17}$$

which combined with (3.9)-(3.10), (3.16) implies

$$\lim_{k\to\infty}\left\|d^k\right\| = \lim_{k\to\infty}\left\|x^k - y^k + \gamma\left(F_k(y^k) - F_k(x^k)\right)\right\| = 0, \tag{3.18}$$

$$\left\|\lim_{k\to\infty}(I-P_{Q_j^k})Ax^k - (I-P_{Q_j^k})Ay^k\right\|^2 = 0, \quad \forall j, \tag{3.19}$$

$$\lim_{k\to\infty}\left\|P_{C_i^k}(z^k) - P_{C_h^k}(z^k)\right\| = 0, \quad \forall i \neq h, \tag{3.20}$$

$$\lim_{k\to\infty}\left\|(I-P_{Q_j^k})Ay^k\right\| = 0, \quad \forall j. \tag{3.21}$$

Since the sequence $\{x^k\}$ is bounded, there exist a subsequence $\{x^{k_l}\}$ of $\{x^k\}$ converging to a point x^* and a corresponding subsequence $\{Ax^{k_l}\}$ of $\{Ax^k\}$ converging to a point Ax^*. Now we will show that $x^* \in SOL(MSFP)$, namely we will show $\lim_{k_l\to\infty} c_i(x^{k_l}) \leq 0$ and $\lim_{k_l\to\infty} q_i(x^{k_l}) \leq 0$ for all i and j.

First, since $P_{Q_j^{k_l}} \in Q_j^{k_l}$, we have

$$q_j(Ax^{k_l}) + \langle\eta_j^{k_l}, P_{Q_j^{k_l}}(Ax^{k_l}) - Ax^{k_l}\rangle \leq 0.$$

We know from Lemma 3.1 that the subgradient sequence $\{\eta_j^k\}$ is bounded. By (3.16) we get $P_{Q_j^{k_l}}(Ax^{k_l}) - Ax^{k_l} \to 0$. Thus, we have $\lim_{k_l\to\infty} q_i(x^{k_l}) \leq 0$ for all and j.

Second, noting that $P_{C_i^{k_l}} \in C_i^{k_l}$, we have

$$c_i(x^{k_l}) + \langle\xi_i^{k_l}, P_{C_i^{k_l}}(x^{k_l}) - x^{k_l}\rangle \leq 0.$$

Since $\{x^k\}$ is bounded, by Lemma 3.1 the sequence $\{\xi_i^k\}$ is also bounded. Then all we need is to show that $P_{C_i^{k_l}} - x^{k_l} \to 0$. We know from (3.19) and (3.21) that $F_{k_l}(y^{k_l}) \to 0$ and $F_{k_l}(x^{k_l}) \to 0$. It follows that $z^{k_l} = x^{k_l} - \gamma F_{k_l}(x^{k_l}) \to x^*$, and then by (3.6), $y^{k_l} \to x^*$. Combining $y^{k_l} = \sum_{i=1}^{t}\alpha_i P_{C_i^{k_l}}(x^{k_l} - \gamma F_{k_l}(x^{k_l}))$ with $F_{k_l}(x^{k_l}) \to 0$ and $\|P_{C_i^{k_l}}(x^{k_l}) - P_{C_h^{k_l}}(x^{k_l})\| \to 0$,

$\forall i \neq h$ by (3.20), we conclude that $y^{k_l} - P_{C_i^{k_l}}(x^{k_l}) \to 0$ since $\sum_{i=1}^{t} \alpha_i = 1$. This leads to $P_{C_i^{k_l}}(x^{k_l}) - x^{k_l} \to 0$, and thereby $\lim_{k_l \to \infty} c_i(x^{k_l}) \le 0$ for $i = 1, 2, \dots, t$.

Replacing z by x^* in (3.17), we have

$$\lim_{k \to \infty} \| x^k - x^* \| = d,$$

furthermore

$$\lim_{k \to \infty} \| Ax^k - Ax^* \| = Ad,$$

on the other hand,

$$\lim_{l \to \infty} \| x^{k_l} - x^* \| = \lim_{l \to \infty} \| Ax^{k_l} - Ax^* \| = 0.$$

Thus, $\lim_{k \to \infty} \|x^k - x^*\| = \lim_{l \to \infty} \|Ax^k - Ax^*\| = 0$. The proof of Theorem 3.1 is complete. □

4 Numerical experiments

In the numerical results listed in Tables 1 and 2, 'Iter', 'Sec.' denote the number of iterations and the cpu time in seconds, respectively. We denote $e_0 = (0, 0, \dots, 0) \in \Re^N$ and $e_1 = (1, 1, \dots, 1) \in \Re^N$. In the both numerical experiments, we take the weights $1/(r + t)$ for both Algorithm 3.1 and Censor's algorithm. The stopping criterion is $\|d\| < \varepsilon = 10^{-5}$.

Example 4.1 The MSFP with

$$A = \begin{bmatrix} 2 & -1 & 3 & 2 & 3 \\ 1 & 2 & 5 & 2 & 1 \\ 2 & 0 & 2 & 1 & -2 \\ 2 & -1 & 0 & -3 & 5 \end{bmatrix};$$

$$C_1 = \{ x \in \Re^5 \mid x_1 + 2x_2 + x_3 + x_4 \le 5 \};$$

$$C_2 = \{ x \in \Re^5 \mid x_2 + 4x_4 + 4x_5 \le 1 \}$$

and

$$Q_1 = \{ y \in \Re^4 \mid y_1 + y_4 \le 1 \};$$

$$Q_2 = \{ y \in \Re^4 \mid 2y_2 + 3y_3 \le 6 \};$$

$$Q_3 = \{ y \in \Re^4 \mid y_3 + 2y_4 \le 10 \}.$$

Consider the following three cases:

Case 1: $x^0 = (1, -1, 1, -1, 1)$;

Case 2: $x^0 = (1, 1, 1, 1, 1)$;

Case 3: $x^0 = (5, 0, 5, 0, 5)$.

Table 1 The numerical results of Example 4.1

Case	Censor $\gamma = 1$	Algo. 3.1 $\gamma = 1$ $t_k = 0.1$	Censor $\gamma = 0.6$	Algo. 3.1 $\gamma = 0.6$ $t_k = 0.1$	Censor $\gamma = 1.8$	Algo. 3.1 $\gamma = 1.8$ $t_k = 0.1$
I	Iter. = 1,051 Sec. = 1.043	Iter. = 146 Sec. = 0.401	Iter. = 1,867 Sec. = 1.480	Iter. = 224 Sec. = 0.334	Iter. = 832 Sec. = 0.700	Iter. = 89 Sec. = 0.062
II	Iter. = 197 Sec. = 0.320	Iter. = 28 Sec. = 0.017	Iter. = 289 Sec. = 0.466	Iter. = 62 Sec. = 0.0751	Iter. = 87 Sec. = 0.068	Iter. = 9 Sec. = 0.010
III	Iter. = 207 Sec. = 0.360	Iter. = 62 Sec. = 0.049	Iter. = 362 Sec. = 0.551	Iter. = 67 Sec. = 0.0728	Iter. = 139 Sec. = 0.217	Iter. = 17 Sec. = 0.020

Table 2 The numerical results of Example 4.2

N	t, r	Censor $\gamma = 1$	Algo. 3.1 $\gamma = 1$ $t_k = 0.01$	Censor $\gamma = 0.8$	Algo. 3.1 $\gamma = 0.8$ $t_k = 0.01$	Censor $\gamma = 1.6$	Algo. 3.1 $\gamma = 1.6$ $t_k = 0.01$
$N = 20$	$t = 5$ $r = 5$	Iter. = 181 Sec. = 0.268	Iter. = 16 Sec. = 0.021	Iter. = 288 Sec. = 0.499	Iter. = 20 Sec. = 0.022	Iter. = 147 Sec. = 0.213	Iter. = 9 Sec. = 0.017
$N = 40$	$t = 10$ $r = 15$	Iter. = 1,012 Sec. = 1.032	Iter. = 39 Sec. = 0.048	Iter. = 2,320 Sec. = 2.122	Iter. = 57 Sec. = 0.059	Iter. = 893 Sec. = 0.795	Iter. = 19 Sec. = 0.031

Example 4.2 [19] In this example, because the step is related to $\rho(A^TA)$, for easy control of the spectral radius, we take diagonal matrices A and $a_{ii} \in (0,1)$ generated randomly

$$C_i = \{x \in \Re^N \mid \|x - d_i\|^2 \le r_i\}, \quad i = 1, 2, \ldots, t;$$
$$Q_j = \{x \in \Re^N \mid L_j \le y \le U_j\}, \quad j = 1, 2, \ldots, r;$$

where d_i is thecenter of the ball C_i, $e_0 \le d_i \le 10e_1$, and $r_i \in (40, 50)$ is the radius, d_i and r_i are all generated randomly. L_j and U_j are the boundary of the box Q_j and are also generated randomly, satisfying $20e_1 \le L_j \le 30e_1$, $40e_1 \le U_j \le 80e_1$. In this test, we take e_0 as the initial point.

In Tables 1-2, the results showed that for most of the initial point, the number of iterative steps and the CPU time of Algorithm 3.1 are obviously less than those of Censor *et al.*'s algorithm. Moreover, when we take $N = 1{,}000$, the number of iteration steps of Algorithm 3.1 is only hundreds of times. The numerical results also show that for large scale problems Algorithm 3.1 converges faster than Censor's algorithm.

5 Conclusion

The multiple-set split feasibility problem arises in many practical applications in the real world. This paper constructed a new searching direction, which is not the gradient of a corresponding function. This different direction results in a very different way of analysis. And preliminary numerical results show that our new method converges faster, and this becomes more obvious while the dimension is increasing. Finally, the theoretically analysis is based on the assumption that the solution set of the MSSFP is nonempty.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Author details
[1]School of Management, University of Shanghai for Science and Technology, Shanghai, 200093, China. [2]College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan 454000, China. [3]School of Physics and Chemistry, Henan Polytechnic University, Shiji Road, Jiaozuo, Henan 454000, China.

Acknowledgements
This work was supported by Natural Science Foundation of Shanghai (14ZR1429200), National Science Foundation of China (11171221), National Natural Science Foundation of China (61403255), Shanghai Leading Academic Discipline Project under Grant XTKX2012, Innovation Program of Shanghai Municipal Education Commission under Grant 14YZ094, Doctoral Program Foundation of Institutions of Higher Education of China under Grant 20123120110004, Doctoral Starting Projection of the University of Shanghai for Science and Technology under Grant ID-10-303-002, and Young Teacher Training Projection Program of Shanghai for Science and Technology.

Received: 2 November 2014 Accepted: 26 January 2015 Published online: 05 February 2015

References

1. Censor, Y, Elfving, T, Kopf, N, Bortfeld, T: The multiple-sets split feasibility problem and its applications for inverse problems. Inverse Problems **21**, 2071-2084 (2005)
2. Censor, Y, Elfving, T: A multiprojection algorithm using Bregman projections in a product space. Numer. Algorithms **8**, 221-239 (1994)
3. Censor, Y, Bortfel, D, Martin, B, Trofimov, A: A unified approach for inversion problems in intensity-modulated radiation therapy. Phys. Med. Biol. **51**, 2353-2365 (2006)
4. Byrne, C: Iterative oblique projection onto convex sets and the split feasibility problem. Inverse Probl. **18**, 441-453 (2002)
5. Dang, Y, Gao, Y: The strong convergence of a KM-CQ-like algorithm for split feasibility problem. Inverse Problems **27**, 015007 (2011)
6. Dang, Y, Gao, Y: A perturbed projection algorithm with inertial technique for split feasibility problem. J. Appl. Math. (2012). doi:10.1155/2012/207323
7. Dang, Y, Gao, Y: An extrapolated iterative algorithm for multiple-set split feasibility problem. Abstr. Appl. Anal. **2012**, Article ID 149508 (2012)
8. Yang, Q: The relaxed CQ algorithm solving the split feasibility problem. Inverse Problems **20**, 1261-1266 (2004)
9. Xu, H: A variable Krasnosel'skii-Mann algorithm and the multiple-set split feasibility problem. Inverse Problems **22**, 2021-2034 (2006)
10. Masad, E, Reich, S: A note on the multiple-set split convex feasibility problem in Hilbert space. J. Nonlinear Convex Anal. **8**, 367-371 (2007)
11. Censor, Y, Alexander, S: On string-averaging for sparse problems and on the split common fixed point problem. Contemp. Math. **513**, 125-142 (2010)
12. Censor, Y, Alexander, S: The split common fixed point problem for directed operators. J. Convex Anal. **16**, 587-600 (2009)
13. Censor, Y, Motova, A, Segal, A: Perturbed projections and subgradient projections for the multiple-sets split feasibility problem. J. Math. Anal. Appl. **327**, 1244-1256 (2007)
14. Gao, Y: Piecewise smooth Lyapunov function for a nonlinear dynamical system. J. Convex Anal. **19**, 1009-1016 (2012)
15. Bauschke, HH, Jonathan, MB: On projection algorithms for solving convex feasibility problems. SIAM Rev. **38**, 367-426 (1996)
16. Bauschke, HH, Combettes, PL: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer, Berlin (2011)
17. Byne, C: An unified treatment of some iterative algorithms in signal processing and image reconstruction. Inverse Probl. **20**, 103-120 (2004)
18. Gao, Y: Nonsmooth Optimization. Science Press, Beijing (2008) (in Chinese)
19. Rocafellar, RT: Convex Analysis. Princeton University Press, Princeton (1971)