Stud. Univ. Babeş-Bolyai Math. 67(2022), No. 1, 129–144
DOI: 10.24193/subbmath.2022.1.09

A new splitting algorithm for equilibrium problems and applications

Trinh Ngoc Hai and Ngo Thi Thuong

Dedicated to the memory of Professor Gábor Kassay.

Abstract. In this paper, we discuss a new splitting algorithm for solving equilibrium problems arising from Nash-Cournot oligopolistic equilibrium problems in electricity markets with non-convex cost functions. Under the strong pseudomonotonicity of the original bifunction and suitable conditions of the component bifunctions, we prove the strong convergence of the proposed algorithm. Our results improve and develop previously discussed extragradient-like splitting algorithms and general extragradient algorithms. We also present some numerical experiments and compare our algorithm with the existing ones.

Mathematics Subject Classification (2010): 54AXX.

Keywords: Equilibrium problem, splitting algorithm, strong pseudomonotonicity, extragradient algorithm.

1. Introduction

In recent years, equilibrium problems (EP) have been investigated by many researchers. It is well known that various classes of optimization, variational inequality, Kakutani fixed point, Nash equilibrium in noncooperative game theory and minimax problems can be formulated as an equilibrium problem [5].

An equilibrium problem can be formulated by means of Ky Fan's inequality [5]:

$$\text{find } x^* \in C \text{ such that } f(x^*, y) \geq 0 \text{ for all } y \in C, \qquad EP(f,C)$$

where C is a nonempty closed convex subset in a Hilbert space H and $f : C \times C \to \mathbb{R}$ is a bifunction such that $f(x,x) = 0$ for all $x \in C$. The set of solutions of $EP(f,C)$ is denoted by $Sol(f,C)$.

Received 17 August 2021; Revised 08 February 2022; Accepted 09 February 2022.

Projection-type methods are very popular for solving equilibrium problems because the iterations can be performed cheaply. At each iteration of these algorithms, we have to solve the strongly convex problem

$$\min\{\lambda_k f(x^k, y) + \frac{1}{2}\|y - x^k\|^2 : y \in C\}, \tag{1.1}$$

where $\lambda_k > 0$ is the step size and x^k is the current approximation of the solution. Note that, in the variational inequalities case, when $f(x, y) := \langle F(x), y - x\rangle$, where $F : C \to C$ is a mapping, problem (1.1) becomes

$$\text{find} \quad P_C\left(x^k - \lambda_k F(x^k)\right), \tag{1.2}$$

where P_C is the projection onto C.

The computational cost of solving problems (1.1) is the main factor influencing performance of projection-type methods. One effective way to reduce the computational cost is to decompose f into two or more component bifunctions. Then, instead of solving (1.1), we have to solve only the simpler subproblems for these component bifunctions [4, 6, 11, 12]. Since 1950s, operator splitting techniques have been successfully used in PDE, large-scale optimization problems and signal processing to reduce complex problems into a series of simpler subproblems [7]. In the past decade, this technique has been received much attention due to its vast applications [2, 6, 4, 12]. Recently, in [1], the authors have introduced splitting algorithms for equilibrium problems when $f = f_1 + f_2$. Under the strong pseudomonotonicity of the bifunction f and suitable conditions of f_1 and f_2, the algorithm proposed in [1] is strongly convergent. However, it may happen that the bifunction f is decomposed into three components, i.e., $f = f_1 + f_2 + f_3$ (see Example 4.1 in Sect. 4). Then, the two-component splitting algorithm in [1] is not suitable. In this paper, inspired by work in [1, 9], we propose a new splitting algorithm for solving this class of equilibrium problems.

The rest of this article is divided into three sections. Section 2 recalls some mathematical preliminaries needed in the sequel. Section 3 presents a three-component splitting algorithm for equilibrium problems and provides the convergence analysis of the proposed algorithm. Some preliminary computational results are presented in the last section. Also in this section, we introduce a new Nash-Cournot equilibrium model for electricity markets. In contrast to the existing ones, the new model contains nonconvex cost functions, and hence, the bifunction f of the corresponding equilibrium problem is decomposed into three components. We then apply the proposed algorithm to solve this problem.

2. Preliminaries

In this section, we present some basic concepts, properties, and notations which will be useful in the sequel. Let H be a real Hilbert space, equipped with the Euclidean inner product $\langle ., .\rangle$ and the associated norm $\|.\|$, C be a nonempty closed convex subset in H. Let $f : C \times C \to \mathbb{R}$ be a bifunction on C, satisfying $f(x, x) = 0$ for all $x \in C$.

Definition 2.1. [15] A bifunction $f : C \times C \to \mathbb{R}$ is said to be

1. γ-strongly monotone on C if there exists a constant $\gamma > 0$ such that for all $x, y \in C$,
$$f(x, y) + f(y, x) \leq -\gamma \|x - y\|^2;$$
2. monotone on C if for all $x, y \in C$,
$$f(x, y) + f(y, x) \leq 0;$$
3. γ-strongly pseudomonotone on C if there exists a constant $\gamma > 0$ such that for all $x, y \in C$,
$$f(x, y) \geq 0 \Rightarrow f(y, x) \leq -\gamma \|x - y\|^2;$$
4. pseudomonotone on C if for all $x, y \in C$,
$$f(x, y) \geq 0 \Rightarrow f(y, x) \leq 0.$$

Definition 2.2. [1] A bifunction $f : C \times C \to \mathbb{R}$ is said to be Lipschitz-type continuous if there exists a constant $Q > 0$ such that for all $x, y, z \in C$,

$$|f(x, y) + f(y, z) - f(x, z)| \leq Q \|x - y\| \|y - z\|. \tag{2.1}$$

Note that, if we choose $f(x, y) := \langle Fx, y - x \rangle$, where $F : C \to C$ is a Lipschitz continuous mapping, then the corresponding bifunction f is Lipschitz-type continuous.

Definition 2.3. [1] A bifunction $f : C \times C \to \mathbb{R}$ is said to be partially τ-Hölder continuous on C if there exist a constant $L > 0$ and $\tau \in (0, 1]$ such that for all $x, y, z \in C$, at least one of the following conditions is satisfied:

(i) $|f(x, y) - f(z, y)| \leq L \|x - z\|^\tau$;
(ii) $|f(x, y) - f(x, z)| \leq L \|y - z\|^\tau$.

It is easy seen that, if an equilibrium bifunction f is τ-Hölder continuous on C then

$$|f(x, y)| \leq L \|x - y\|^\tau \quad \forall x, y \in C. \tag{2.2}$$

Definition 2.4. The subdifferential of a function $u : H \to \mathbb{R}$ at x is the set:

$$\partial u(x) := \{w \in H : u(y) - u(x) \geq \langle w, y - x \rangle \ \ \forall y \in H\}.$$

The normal cone of C at $x \in C$ is defined by

$$N_C(x) := \{q \in H : \langle q, y - x \rangle \leq 0 \ \forall y \in C\}.$$

In order to prove our main results, we need the following lemmas.

Lemma 2.5. [16] *Let $f : C \to \mathbb{R}$ be convex and subdifferentiable on C. Then, x^* is a solution of the problem*

$$\min\{f(x) : x \in C\}$$

if and only if $0 \in \partial f(x^*) + N_C(x^*)$.

Lemma 2.6. (Lemma 2.5 [18]) *Let $\{\alpha_k\}$, $\{\beta_k\}$, $\{\lambda_k\}$ be sequences of nonnegative numbers satisfying*

$$\alpha_{k+1} \leq (1-\lambda_k)\alpha_k + \lambda_k\gamma_k + \beta_k \ \forall k \geq 1.$$

If $\lambda_k \in (0,1)$ $\forall k \geq 1$, $\sum_{k=1}^{\infty} \lambda_k = \infty$, $\limsup_{k\to\infty} \gamma_k \leq 0$ and $\sum_{k=1}^{\infty} \beta_k < \infty$ then $\lim_{k\to\infty} \alpha_k = 0$.

3. Three-component splitting algorithm

Let C be a nonempty, closed, convex subset in a Hilbert space H and $f : C \times C :\to \mathbb{R}$ be a bifunction on C. We are interested in the equilibrium problem

$$\text{find } x^* \in C \text{ such that } f(x^*, y) \geq 0 \text{ for all } y \in C, \tag{3.1}$$

where f can be decomposed into three components: $f = f_1 + f_2 + f_3$, f_i $(i = 1, 2, 3)$ are equilibrium bifunctions on C, i.e., $f_i(x, x) = 0$ for all $x \in C$.

Assumption 1. In this paper, we assume that

A.1 For each $x \in C$, the function $f_i(x, .)$ $(i = 1, 2, 3)$ is lower semicontinuous, convex and for each $y \in C$, the function $f(., y)$ is hemicontinuous on C, i.e.

$$\lim_{t\to 0} f(tz + (1-t)x, y) = f(x, y), \quad \forall x, y, z \in C.$$

A.2 The bifunction f is γ-strongly pseudomonotone.

Note that under assumptions **A.1** and **A.2**, problem $EP(f, C)$ has a unique solution [13]. To find this solution, we propose the following three-component splitting algorithm.

Algorithm 3.1. (Three-component splitting algorithm - 3-CSA))
Step 0. Choose $x^0 \in C$, $\lambda_k \subset (0, +\infty)$. Set $k = 0$.

Step 1. Given x^k, compute x^{k+1} as

$$\bar{x}^k = \text{argmin}\Big\{\lambda_k f_1(x^k, y) + \frac{1}{2}\|y - x^k\|^2 : y \in C\Big\},$$
$$\tilde{x}^k = \text{argmin}\Big\{\lambda_k f_2(\bar{x}^k, y) + \frac{1}{2}\|y - \bar{x}^k\|^2 : y \in C\Big\},$$
$$x^{k+1} = \text{argmin}\Big\{\lambda_k f_3(\tilde{x}^k, y) + \frac{1}{2}\|y - \tilde{x}^k\|^2 : y \in C\Big\}.$$

Step 2. Update $k := k + 1$ and go to **Step 1**.

Theorem 3.2. *Assume that conditions* **A.1**, **A.2** *hold, f_1 is Q-Lipschitz-type continuous, f_i is partially τ_i-Holder continuous, $i = 2, 3$. Moreover, suppose that*

(B.1) $\sum_{k=1}^{+\infty} \lambda_k = +\infty$,

(B.3) $\sum_{k=1}^{+\infty} (\lambda_k)^{\frac{2}{2-\tau}} < +\infty$,

where $\tau = \min\{\tau_2, \tau_3\}$. Then, the sequence $\{x^k\}$ generated by Algorithm 3.1 strongly converges to the unique solution x^* of $EP(f, C)$.

Proof. Since $\bar{x}^k$ is the unique solution of the problem

$$\min\Big\{\lambda_k f_1(x^k,y)+\frac{1}{2}\|y-x^k\|^2 : y\in C\Big\},$$

there exist $\omega^k \in \partial f_1(x^k,.)(\bar{x}^k)$ and $q^k \in N_C(\bar{x}^k)$ such that

$$0=\lambda_k\omega^k+\bar{x}^k-x^k+q^k.$$

From the definition of $N_C(.)$, we have

$$\big\langle x^k-\bar{x}^k-\lambda_k\omega^k, y-\bar{x}^k\big\rangle \le 0 \ \ \forall y\in C. \tag{3.2}$$

Hence,

$$\lambda_k\big\langle \omega^k, y-\bar{x}^k\big\rangle \le \lambda_k(f_1(x^k,y)-f_1(x^k,\bar{x}^k)) \ \ \forall y\in C. \tag{3.3}$$

Combining (3.2) and (3.3), we get

$$\big\langle x^k-\bar{x}^k, y-x^k\big\rangle \le \lambda_k(f_1(x^k,y)-f_1(x^k,\bar{x}^k))-\|x^k-\bar{x}^k\|^2 \ \ \forall y\in C. \tag{3.4}$$

Analogously, since $\tilde{x}^k$ and x^{k+1} are the solutions of the problems

$$\min\Big\{\lambda_k f_2(\bar{x}^k,y)+\frac{1}{2}\|y-\bar{x}^k\|^2 : y\in C\Big\},$$

$$\min\Big\{\lambda_k f_3(\tilde{x}^k,y)+\frac{1}{2}\|y-\tilde{x}^k\|^2 : y\in C\Big\},$$

it follows that

$$\big\langle \bar{x}^k-\tilde{x}^k, y-\bar{x}^k\big\rangle \le \lambda_k(f_2(\bar{x}^k,y)-f_2(\bar{x}^k,\tilde{x}^k))-\|\bar{x}^k-\tilde{x}^k\|^2 \ \ \forall y\in C. \tag{3.5}$$

and

$$\big\langle \tilde{x}^k-x^{k+1}, y-\tilde{x}^k\big\rangle \le \lambda_k(f_3(\tilde{x}^k,y)-f_3(\tilde{x}^k,x^{k+1}))-\|\tilde{x}^k-x^{k+1}\|^2 \ \ \forall y\in C. \tag{3.6}$$

In (3.6), taking $y=\tilde{x}^k$, we get

$$\|x^{k+1}-\tilde{x}^k\|^2 \le -\lambda_k f_3(\tilde{x}^k,x^{k+1}) \le \lambda_k|f_3(\tilde{x}^k,x^{k+1})|. \tag{3.7}$$

Since f_3 is partially τ_3-Holder continuous and $f_3(x,x)=0$ for all $x\in C$, there exists a constant $L_3>0$ such that for all $k\ge 1$, it holds that

$$|f_3(\tilde{x}^k,x^{k+1})| \le L_3\|x^{k+1}-\tilde{x}^k\|^{\tau_3}. \tag{3.8}$$

Combining (3.7) and (3.8), we obtain

$$\|x^{k+1}-\tilde{x}^k\| \le (L_3\lambda_k)^{\frac{1}{2-\tau_3}} \tag{3.9}$$

and

$$\lambda_k|f_3(\tilde{x}^k,x^{k+1})| \le (L_3\lambda_k)^{\frac{2}{2-\tau_3}}. \tag{3.10}$$

In (3.5), taking $y=\bar{x}^k$ and using the partial Holder continuity of f_2, we get

$$\|\tilde{x}^k-\bar{x}^k\| \le (L_2\lambda_k)^{\frac{1}{2-\tau_2}} \tag{3.11}$$

and

$$\lambda_k|f_2(\bar{x}^k,\tilde{x}^k)| \le (L_2\lambda_k)^{\frac{2}{2-\tau_2}}. \tag{3.12}$$

From (3.4) and the Q-Lipschitz continuity of f_1, we arrive at

$$\begin{aligned}
\|\bar{x}^k - y\|^2 &= \|\bar{x}^k - x^k\|^2 + \|x^k - y\|^2 + \langle \bar{x}^k - x^k, x^k - y\rangle \\
&\leq \|x^k - y\|^2 - \|\bar{x}^k - x^k\|^2 + 2\lambda_k(f_1(x^k, y) - f_1(x^k, \bar{x}^k)) \\
&\leq \|x^k - y\|^2 - \|\bar{x}^k - x^k\|^2 + 2\lambda_k(f_1(\bar{x}^k, y) + Q\|\bar{x}^k - x^k\|.\|\bar{x}^k - y\|) \\
&\leq \|x^k - y\|^2 - \|\bar{x}^k - x^k\|^2 + 2\lambda_k f_1(\bar{x}^k, y) \\
&+ (Q\lambda_k)^2\|\bar{x}^k - y\|^2 + \|\bar{x}^k - x^k\|^2 \\
&= \|x^k - y\|^2 + 2\lambda_k f_1(\bar{x}^k, y) + (Q\lambda_k)^2\|\bar{x}^k - y\|^2. \qquad (3.13)
\end{aligned}$$

Analogously to (3.5), (3.6) we get

$$\begin{aligned}
\|\tilde{x}^k - y\|^2 &\leq \|\bar{x}^k - y\|^2 - \|\tilde{x}^k - \bar{x}^k\|^2 + 2\lambda_k(f_2(\bar{x}^k, y) - f_2(\bar{x}^k, \tilde{x}^k)) \\
&\leq \|\bar{x}^k - y\|^2 + 2\lambda_k(f_2(\bar{x}^k, y) - f_2(\bar{x}^k, \tilde{x}^k)) \\
&\leq \|\bar{x}^k - y\|^2 + 2\lambda_k f_2(\bar{x}^k, y) + 2\lambda_k|f_2(\bar{x}^k, \tilde{x}^k)|. \qquad (3.14)
\end{aligned}$$

and

$$\begin{aligned}
\|x^{k+1} - y\|^2 &\leq \|\tilde{x}^k - y\|^2 - \|x^{k+1} - \tilde{x}^k\|^2 + 2\lambda_k(f_3(\tilde{x}^k, y) - f_3(\tilde{x}^k, x^{k+1})) \\
&\leq \|\tilde{x}^k - y\|^2 + 2\lambda_k(f_3(\tilde{x}^k, y) - f_3(\tilde{x}^k, x^{k+1})) \\
&= \|\tilde{x}^k - y\|^2 + 2\lambda_k f_3(\bar{x}^k, y) - 2\lambda_k f_3(\tilde{x}^k, x^{k+1}) \\
&+ 2\lambda_k(f_3(\tilde{x}^k, y) - f_3(\bar{x}^k, y)) \\
&\leq \|\tilde{x}^k - y\|^2 + 2\lambda_k f_3(\bar{x}^k, y) + 2\lambda_k|f_3(\tilde{x}^k, x^{k+1})| \\
&+ 2\lambda_k|f_3(\tilde{x}^k, y) - f_3(\bar{x}^k, y)|. \qquad (3.15)
\end{aligned}$$

From (3.9)-(3.15) and the partial τ_3-Holder continuity of f_3, we have

$$\begin{aligned}
\|x^{k+1} - y\|^2 &\leq \|x^k - y\|^2 + 2\lambda_k f(\bar{x}^k, y) + (Q\lambda_k)^2\|\bar{x}^k - y\|^2 \\
&+ 2\lambda_k|f_2(\bar{x}^k, \tilde{x}^k)| + 2\lambda_k|f_3(\tilde{x}^k, x^{k+1})| \\
&+ 2\lambda_k|f_3(\tilde{x}^k, y) - f_3(\bar{x}^k, y)| \\
&\leq \|x^k - y\|^2 + 2\lambda_k f(\bar{x}^k, y) + (Q\lambda_k)^2\|\bar{x}^k - y\|^2 \\
&+ 2(L_2\lambda_k)^{\frac{2}{2-\tau_2}} + 2(L_3\lambda_k)^{\frac{2}{2-\tau_3}} + 2\lambda_k(L_3\lambda_k)^{\frac{\tau_3}{2-\tau_2}}. \qquad (3.16)
\end{aligned}$$

In (3.16), taking $y = x^*$ and using the γ-strong pseudomonotonicity of f, we obtain

$$\begin{aligned}
\|x^{k+1} - x^*\|^2 &\leq \|x^k - x^*\|^2 - \lambda_k(2\gamma - Q^2\lambda_k)\|\bar{x}^k - x^*\|^2 + \\
&+ 2(L_2\lambda_k)^{\frac{2}{2-\tau_2}} + 2(L_3\lambda_k)^{\frac{2}{2-\tau_3}} + 2\lambda_k(L_3\lambda_k)^{\frac{\tau_3}{2-\tau_2}}. \qquad (3.17)
\end{aligned}$$

Using the inequality $\|a + b\| \geq |\|a\| - \|b\||$ for all $a, b \in H$, we infer that

$$\begin{aligned}
\lambda_k\|\bar{x}^k - x^*\|^2 &\geq \lambda_k(\|\bar{x}^k - x^{k+1}\| - \|x^{k+1} - x^*\|)^2 \\
&\geq (\lambda_k - 1)\|\bar{x}^k - x^{k+1}\|^2 + \lambda_k(1 - \lambda_k)\|x^{k+1} - x^*\|^2. \qquad (3.18)
\end{aligned}$$

Since $\lim_{k\to+\infty} \lambda_k = 0$, without loss of generality we can assume that $1-\lambda_k > 0$ and $2\gamma - Q^2\lambda_k > 0$ for all $k \geq 1$. Combining (3.17) and (3.18), we arrive at

$$\begin{aligned}\|x^{k+1}-x^*\|^2 &\leq \|x^k - x^*\|^2 + (2\gamma - Q^2\lambda_k)(1-\lambda_k)\|\bar{x}^k - x^{k+1}\|^2 \\ &\quad + \lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)\|x^{k+1}-x^*\|^2 \\ &\quad + 2(L_2\lambda_k)^{\frac{2}{2-\tau_2}} + 2(L_3\lambda_k)^{\frac{2}{2-\tau_3}} + 2\lambda_k(L_3\lambda_k)^{\frac{\tau_3}{2-\tau_2}}. \end{aligned} \tag{3.19}$$

On the other hand, it holds that

$$\begin{aligned}\|\bar{x}^k - x^{k+1}\|^2 &= \|\bar{x}^k - \tilde{x}^k\|^2 + \|\tilde{x}^k - x^{k+1}\|^2 + 2\big\langle \tilde{x}^k - x^{k+1}, \bar{x}^k - \tilde{x}^k\big\rangle \\ &\leq (L_2\lambda_k)^{\frac{2}{2-\tau_2}} + (L_3\lambda_k)^{\frac{2}{2-\tau_3}} + 2\big\langle \tilde{x}^k - x^{k+1}, \bar{x}^k - \tilde{x}^k\big\rangle. \end{aligned} \tag{3.20}$$

In (3.6), taking $y = \bar{x}^k$, we get

$$\begin{aligned}2\big\langle \tilde{x}^k - x^{k+1}, \bar{x}^k - \tilde{x}^k\big\rangle &\leq 2\lambda_k(f_3(\tilde{x}^k, \bar{x}^k) - f_3(\tilde{x}^k, x^{k+1})) \\ &\leq 2\lambda_k|f_3(\tilde{x}^k,\bar{x}^k)| + 2\lambda_k|f_3(\tilde{x}^k, x^{k+1})| \\ &\leq 2\lambda_k L_3\|\tilde{x}^k - \bar{x}^k\|^{\tau_3} + 2(L_3\lambda_k)^{\frac{2}{2-\tau_3}} \\ &\leq 2\lambda_k L_3(L_2\lambda_k)^{\frac{\tau_3}{2-\tau_2}} + 2(L_3\lambda_k)^{\frac{2}{2-\tau_3}}. \end{aligned} \tag{3.21}$$

Combining (3.19)-(3.21), we have

$$\begin{aligned}[1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)]\|x^{k+1}-x^*\|^2 &\leq \|x^k - x^*\|^2 + 2(\gamma+1)(L_2\lambda_k)^{\frac{2}{2-\tau_2}} \\ &\quad + 2(2\gamma+1)(L_3\lambda_k)^{\frac{2}{2-\tau_3}} \\ &\quad + 2\Big(L_3L_2^{\frac{\tau_3}{2-\tau_2}} + L_3^{\frac{\tau_3}{2-\tau_2}}\Big)\lambda_k^{\frac{2+\tau_3-\tau_2}{2-\tau_2}}, \end{aligned}$$

or

$$\|x^{k+1}-x^*\|^2 \leq (1-A_k)\|x^k - x^*\|^2 + B_k + C_k + D_k,$$

where

$$\begin{aligned} A_k &= \frac{\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}{1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}, \\ B_k &= \frac{2(\gamma+1)(L_2\lambda_k)^{\frac{2}{2-\tau_2}}}{1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}, \\ C_k &= \frac{2(2\gamma+1)(L_3\lambda_k)^{\frac{2}{2-\tau_3}}}{1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}, \\ D_k &= \frac{2\Big(L_3L_2^{\frac{\tau_3}{2-\tau_2}} + L_3^{\frac{\tau_3}{2-\tau_2}}\Big)\lambda_k^{\frac{2+\tau_3-\tau_2}{2-\tau_2}}}{1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}. \end{aligned}$$

We have

$$\begin{aligned} &\lim_{k\to+\infty}\Big(\frac{\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}{1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}.\frac{1}{\lambda_k}\Big) \\ &= \lim_{k\to+\infty}\Big(\frac{(1-\lambda_k)(2\gamma - Q^2\lambda_k)}{1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}\Big) = 2\gamma, \end{aligned}$$

moreover, since $\sum_{k=1}^{+\infty} \lambda_k = +\infty$, it follows that

$$\sum_{k=1}^{+\infty} A_k = \sum_{k=1}^{+\infty} \Big(\frac{\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}{1+\lambda_k(1-\lambda_k)(2\gamma - Q^2\lambda_k)}\Big) = +\infty.$$

On the other hand,

$$\lim_{k\to+\infty} \big(B_k.\frac{1}{\lambda_k^{\frac{2}{2-\tau}}}\big) = 0, \ \lim_{k\to+\infty} \big(C_k.\frac{1}{\lambda_k^{\frac{2}{2-\tau}}}\big) = 0, \ \lim_{k\to+\infty} \big(D_k.\frac{1}{\lambda_k^{\frac{2}{2-\tau}}}\big) = 0$$

it implies that

$$\sum_{k=1}^{+\infty} (B_k + C_k + D_k) < +\infty.$$

Applying Lemma 2.6, we get $\lim_{k\to+\infty} \|x^k - x^*\|^2 = 0$ or $\lim_{k\to+\infty} x^k = x^*$. □

Remark 3.3. (a) Since $\tau \in (0,1]$, $\frac{2}{2-\tau} \in (1,2]$, we can choose a sequence $\{\lambda_k\}$ satisfying conditions (B.1)-(B.3), for example, $\lambda_k = \frac{1}{k^\alpha}$, where $\alpha \in (\frac{2-\tau}{2}, 1)$.

(b) In Algorithm 3.1, we need not to know the Lipschitz constant Q of f_1.

(c) Algorithm 3.1 reminds the so-called General Extragradient Algorithm in [9], in the sense that the both algorithms require three subproblems at each iteration. However, our algorithm has a clear advantage: at each iteration, we have to solve only subproblems for the component bifunctions f_i, instead of solving subproblems for the whole bifunction f. Hence, our algorithm may have a low computational cost when the function f has a complicated structure, while the component bifunctions f_i are simpler.

(d) If $f_3 = 0$, i.e. $f = f_1 + f_2$, then the new algorithm collapses to the existing one in [1].

4. Numerical examples

In this section, we provide an application of the proposed algorithm to electricity markets. We also compare our algorithm with some existing ones. All the programmings are implemented in MATLAB R2010b running on a PC with Intel®Core2$^{\text{TM}}$ Quad Processor Q9400 2.66Ghz 4GB Ram.

Example 4.1. (*Nash-Cournot oligopolistic equilibrium model for electricity markets with non-convex cost functions*)
We introduce a Nash-Cournot oligopolistic equilibrium model for electricity markets. In contrast to the existing ones considered in [14, 1], the new model contains non-convex cost functions. In this situation, the three-component splitting algorithm seem to be the most suitable one for solving the corresponding equilibrium problem.

Consider an electricity market with N companies. Suppose that x_j is the power generation level of company j, $(j = 1, \dots, N)$. Then, the total power generation of the market is

$$\sigma := \sum_{k=1}^{N} x_k.$$

Obviously, the more electricity companies produce, the lower electricity price is. Hence, we assume that the electricity price p is inversely proportional to σ and is defined by

$$p(x) = 200 - 2\sum_{k=1}^{N} x_k.$$

To produce electricity, companies have to pay two costs: production cost and environmental cost. The cost of production per unit of electricity decreases as the production level increases. Hence, we assume that the production cost h_j^{prod} of company j is a concave function of x_j:

$$h_j^{\text{prod}}(x_j) = a_j\sqrt{x_j} + b_j.$$

Meanwhile, the environmental charge per unit of electricity increases as the product level increases. Hence, the environmental cost h_j^{env} of company j is a convex function of x_j:

$$h_j^{\text{env}}(x^j) = c_j {x_j}^2 + d_j.$$

And so, the total cost h_j of company j is:

$$h_j(x_j) = h_j^{\text{prod}}(x_j) + h_j^{\text{env}}(x^j) = a_j\sqrt{x_j} + b_j + c_j {x_j}^2 + d_j.$$

Let $N = 6$. The parameters a_j, b_j, c_j, d_j are given in Table 1.

j	a_j	b_j	c_j	d_j
1	1.0	2.0	0.05	2.2
2	0.7	2.1	0.06	2.1
3	0.8	1.9	0.03	1.9
4	0.9	1.8	0.02	1.8
5	0.8	2.2	0.01	2.3
6	0.6	2.3	0.04	1.8

TABLE 1. The parameters of the cost function

The profit ξ_j of a company j is

$$\xi_j(x) := p(x)x_j - h_j(x_j) = \left(200 - 2\sum_{k=1}^{N} x_k\right) x_j - h_j(x_j),$$

where $x = (x_1, \ldots, x_N)^T \in C := \left\{x \in \mathbb{R}^N : \alpha_j \le x_j \le \beta_j\right\}$, α_j, β_j are given in Table 2.

j	1	2	3	4	5	6
α_j	10	10	10	10	10	10
β_j	90	70	100	60	110	50

TABLE 2. The lower and upper bounds for power generation levels x_j

We are interested in a such point $x^* = (x_1^*, \ldots, x_N^*) \in C$ satisfying

$$\xi_j(x_1^*, \ldots, x_{j-1}^*, y_j, x_{j+1}^*, \ldots, x_N^*) \leq \xi_j(x_1^*, \ldots, x_N^*).$$

for all $y = (y_1, \ldots, y_N) \in C, j = 1, \ldots, N$. The point x^* is called the Nash equilibrium. Let

$$\zeta(x, y) := \varphi(x, x) - \varphi(x, y),$$

where

$$\begin{aligned}\varphi(x, y) &= \sum_{j=1}^{N} \xi_j(x_1, \ldots, x_{j-1}, y_j, x_{j+1}, \ldots, x_N) \\ &= \sum_{j=1}^{N} \left[200 - 2\left(\sum_{k \neq j} x_k + y_j\right)\right] y_j - \sum_{j=1}^{N} h_j(y_j).\end{aligned}$$

Then x^* a Nash equilibrium point of this model if and only if it is a solution of the equilibrium problem (see [10]):

$$\text{find } x^* \in C \text{ such that } \zeta(x^*, y) \geq 0 \ \forall y \in C. \tag{4.1}$$

We have

$$\zeta(x, y) = \langle (A + B)x + By + q, y - x \rangle + h(y) - h(x), \tag{4.2}$$

where

$$A := \begin{pmatrix} 0 & 2 & 2 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 & 2 & 2 \\ 2 & 2 & 0 & 2 & 2 & 2 \\ 2 & 2 & 2 & 0 & 2 & 2 \\ 2 & 2 & 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 2 & 2 & 0 \end{pmatrix}, \ B := \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix},$$

and $q = -(100, 100, 100, 100, 100, 100)^T$. However, the bifunction ζ given by (4.2) is not strongly pseudomonotone (even not pseudomonotone). This bifunction can be rewritten as

$$\zeta(x, y) = f(x, y) + 0.6\langle B(y - x), y - x \rangle,$$

where $f(x, y) = \langle (A + 1.6B)x + 0.4By + q, y - x \rangle + h(y) - h(x)$. It is easy seen that the matrix B is positive definite, hence x^* is a solution of the equilibrium problem (4.1) if and only if it is a solution of the problem (see [14]):

$$\text{find } x^* \in C \text{ such that } f(x^*, y) \geq 0 \ \forall y \in C. \tag{4.3}$$

Let us prove the bifunction f is strongly pseudomonotone. Indeed, for all $x, y \in C$, we have

$$f(x,y) + f(y,x) = -\langle (A + 1.2B)(x-y), (x-y)\rangle .$$

Since $A+1.2B$ is a positive definite matrix, it implies that the bifunction f is strongly monotone, and hence, is strongly pseudomonotone. Let

$$f_1(x,y) = \langle (A+1.6B)x + 0.4By + q, y - x\rangle,$$

$$f_2(x,y) = \sum_{j=0}^{N} \left(h_j^{\text{env}}(y_j) - h_j^{\text{env}}(x_j)\right).$$

and

$$f_3(x,y) = \sum_{j=0}^{N} \left(h_j^{\text{prod}}(y_j) - h_j^{\text{prod}}(x_j)\right),$$

It is easy seen that f_1, f_2 and f_3 satisfy all conditions of the proposed algorithm. Now we will apply this algorithms to solve problem (4.3). Note that in this example, subproblems of f_1 and f_2 have quadratic forms and are much easier to solve than general convex problems. Moreover, although the cost functions h_j^{prod} are concave, the subproblems of f_3 is convex if $\lambda_k \le \frac{1}{6}$.
We implement the algorithm with the starting point $x^0 = (0,0,0,0,0,0)^T$, $\lambda_k = \frac{1}{k+6}$ and the stopping criteria $\|x^{k+1} - x^k\| \le 10^{-4}$. The test results are reported in Table 3. The algorithm finds the approximation of the solution after 105 iterations.

Iter(k)	x_k^1	x_k^2	x_k^3	x_k^4	x_k^5	x_k^6	$\|x_{k-1} - x^k\|$
0	0	0	0	0	0	0	
1	22.9133	22.8534	23.0463	23.1103	23.1777	22.9841	31.4327
2	10.0597	10.0000	10.2182	10.2922	10.3731	10.1480	12.9558
3	15.3184	15.2412	15.5167	15.6095	15.7111	15.4289	3.7680
4	13.7630	13.6767	13.9837	14.0868	14.2002	13.8865	0.7174
5	14.0422	13.9487	14.2802	14.3913	14.5139	14.1756	0.0755
6	14.0034	13.9046	14.2542	14.3713	14.5007	14.1441	0.0200
7	13.9975	13.8947	14.2579	14.3796	14.5143	14.1437	0.0152
...	...	...	...	...	...	...	...
105	13.9815	13.8658	14.2731	14.4099	14.5630	14.1455	9.903810^{-5}

TABLE 3. Iterations of the proposed algorithm with starting point $x_0 = (0,0,0,0,0,0)^T$

Example 4.2. We compare our algorithm with the Armijo Line Search Algorithm (*ALS*) (Algorithm 1 in [8]), the General Extragradient Algorithm (*GEA*) in [9], the Splitting Sequential Algorithm (*SAL*) (Algorithm 1 in [1]) and the Subgradient Algorithm (SGA) given by Santos in [17]. Consider the equilibrium problem

$$\text{find } x \in C \text{ such that } \langle Ax + P(x), y - x\rangle + \varphi(y) - \varphi(x) \ge 0 \ \forall y \in C,$$

where the feasible set $C \subset \mathbb{R}^5$ is given by

$$C := \{x \in \mathbb{R}^5 : -5 \le x_i \le 5 \ \forall i = 1, \ldots, 5\},$$

$$\varphi : \mathbb{R}^5 \to \mathbb{R},\ \varphi(x) = \|x\|^2,$$
$$F : \mathbb{R}^5 \to \mathbb{R}^5,\ F(x) = Ax + P(x),$$

with

$$A := \begin{pmatrix} 3 & 1 & 0 & 1 & 2 \\ 1 & 5 & -1 & 0 & 1 \\ 0 & -1 & 4 & 2 & -2 \\ 1 & 0 & 2 & 6 & -1 \\ 2 & 1 & -2 & -1 & 5 \end{pmatrix},$$

and $P : \mathbb{R}^5 \to \mathbb{R}^5$ is the proximal mapping of the function

$$h(x) := \frac{\|x\|^4}{4},$$

i.e.,

$$P(x) := \operatorname{argmin} \left\{ \frac{\|y\|^4}{4} + \frac{1}{2}\|y - x\|^2 : y \in \mathbb{R}^5 \right\}.$$

Note that, since we do not have a closed form of $P(x)$, to compute the value of this mapping, we have to solve a strongly convex problem. In our algorithm, let

$$f_1(x, y) := \langle Ax, y - x \rangle,$$
$$f_2(x, y) := \langle P(x), y - x \rangle,$$
$$f_3(x, y) := \varphi(y) - \varphi(x)$$

and $f := f_1 + f_2 + f_3$.

In Algorithm SAL, let

$$f_1(x, y) := \langle Ax + P(x), y - x \rangle,$$
$$f_2(x, y) := \varphi(y) - \varphi(x)$$

and $f := f_1 + f_2$. It is easy seen that all the conditions of the four algorithms are satisfied. Moreover, the mapping P is nonexpansive and the Lipschitz constants of f (defined in [8]) are

$$c_1 = c_2 = \frac{1}{2}(\|A\| + 1).$$

We apply the four algorithms for solving $EP(f, C)$ with the parameters:

- In Algorithm GEA: $\alpha_k = \beta_k = \frac{1}{7c_1}$.
- In Algorithm ALS: $G(x) := \|x\|^2$, $\eta = 0.5$; $\rho = 1$.
- In Algorithm SAL $\lambda_k = \frac{1}{k}$.
- In Algorithm SGA $\beta_k = \frac{1}{k}$, $\rho_k = 1$, $\epsilon_k = 0$, $\xi_k = 0$.
- In our algorithm: $\lambda_k = \frac{1}{k}$.

All the algorithms use the same starting points and the same stopping rule:

$$\|x^k - x^*\| \le 3.10^{-4},$$

where $x^* = (0, 0, 0, 0, 0)^T$ is the unique solution of the $EP(f, C)$. The results are presented in Table 4.

	$x^0 = (5,5,5,5,5)^T$		$x^0 = (1,1,1,1,1)^T$		$x^0 = (1,2,3,4,5)^T$		$x^0 = (-3,-5,2,-4,4)^T$	
	CPU times	Iter.	CPU times	Iter.	CPU times	Iter.	CPU times	Iter.
Alg. GEA	-	-	-	-	-	-	-	-
Alg. SGA	-	-	0.6004	14	-	-	-	-
Alg. ALS	11.1323	27	8.6821	21	11.3570	29	11.3248	26
Alg. SAL	0.4357	11	0.4470	10	0.5019	12	0.3509	10
Alg. 3-CSA	0.4604	12	0.3839	9	0.4860	11	0.5082	13

TABLE 4. Comparision of the algorithms. (-) means the algorithm does not obtain the required accuracy after 100s.

From this table we can see that, if the initial approximation x^0 is close enough to the exact solution x^*, say, $||x^0 - x^*|| \leq 7.4$, then the performance of 3-CSA is the best among four above mentioned algorithms.

Example 4.3. Consider problem $EP(f, C)$ with

$$C := \left\{x = (x_1, x_2, \ldots, x_m) \in \mathbb{R}^m : 2x_1^2 + x_2^2 + \ldots + x_m^2 \leq 1\right\}$$

and $f : C \times C \to \mathbb{R}$, defined by

$$f(x, y) := \langle Ax, y - x\rangle + y^2 - x^2 + \langle y, y - x\rangle \ \forall x, y \in C,$$

where $A = (a_{ij})$ is a $m \times m$ matrix and

$$a_{ij} = \begin{cases} 0 & \text{if } i \neq j, \\ 1.1 & \text{if } i = j. \end{cases}$$

It is easy seen that $f(x, y) + f(y, x) = -0.1\|x - y\|^2$ for all $x, y \in C$, and hence f is strongly monotone on C. All conditions of the three-component splitting algorithm (3-CSA) and the splitting sequential algorithm (SAL) (Algorithm 1 in[1]) are satisfied. We will use this problem to compare them. For 3-CSA, let

$$f_1(x, y) := \langle Ax, y - x\rangle, \ f_2(x, y) := y^2 - x^2, \ f_3(x, y) := \langle y, y - x\rangle \ \forall x, y \in C.$$

For SAL, define

$$\tilde{f}_1(x, y) := \langle Ax, y - x\rangle, \ \tilde{f}_2(x, y) := y^2 - x^2 + \langle y, y - x\rangle \ \forall x, y \in C.$$

Note that the problem has a unique solution $x^* = (0, 0, \ldots, 0)^T$. In the both algorithms, we use the same step-size $\lambda_k = \frac{1}{k}$ for all $k \geq 1$, the same stopping criteria $\|x^k - x^*\| \leq \epsilon$ and the same starting point x^0, which is randomly generated. 3-CSA now becomes

$$\begin{cases} x^0 \in C, \\ \bar{x}^k = \operatorname{argmin}\left\{\lambda_k \left\langle Ax^k, y - x\right\rangle + \frac{1}{2}\|y - x^k\|^2 : y \in C\right\}, \\ \tilde{x}^k = \operatorname{argmin}\left\{\lambda_k(\|y\|^2 - \|\bar{x}^k\|^2) + \frac{1}{2}\|y - \bar{x}^k\|^2 : y \in C\right\}, \\ x^{k+1} = \operatorname{argmin}\left\{\lambda_k \left\langle y, y - \tilde{x}^k\right\rangle + \frac{1}{2}\|y - \tilde{x}^k\|^2 : y \in C\right\}. \end{cases}$$

From the definition of $\bar{x}^k$, it implies that

$$\lambda_k Ax^k + \bar{x}^k - x^k + q = 0,$$

where q is a normal vector of C at $\bar{x}^k$. Hence,

$$\left\langle x^k - \lambda_k A x^k - \bar{x}^k, z - \bar{x}^k \right\rangle \leq 0 \ \forall z \in C.$$

It follows that $\bar{x}^k = P_C(x^k - \lambda_k A x^k)$. Similarly, we have

$$\tilde{x}^k = P_C\left(\frac{1}{1+2\lambda_k}\bar{x}^k\right).$$

Since 0 and $\bar{x}^k$ belong to C, it implies that $\frac{1}{1+2\lambda_k}\bar{x}^k \in C$, and hence, $\tilde{x}^k = \frac{1}{1+2\lambda_k}\bar{x}^k$. Analogously, we have $x^{k+1} = \frac{1+\lambda_k}{1+2\lambda_k}\tilde{x}^k$, and hence, 3-CSA has the following closed form:

$$\begin{cases} x^0 \in C, \\ x^{k+1} = \frac{1+\lambda_k}{(1+2\lambda_k)^2} P_C(x^k - \lambda_k A x^k). \end{cases}$$

Similarly, in this problem, SAL can be rewritten as

$$\begin{cases} y^0 \in C, \\ y^{k+1} = \frac{1+\lambda_k}{1+4\lambda_k} P_C(y^k - \lambda_k A y^k). \end{cases}$$

We have $\frac{1+\lambda_k}{(1+2\lambda_k)^2} < \frac{1+\lambda_k}{1+4\lambda_k}$. Hence, by induction, it is easy seen that $\|x^k - x^*\| < \|y^k - y^*\|$ for all $k \geq 1$. This means that in this problem, 3-CSA requires less iterations than SAL does. For more specific comparisons, we test these two algorithms in the problem with different m and ϵ. The results are presented in Table 5. From this table, we can see that the results of 3-CSA are better than those of SAL in terms of iterations and computational time.

		3-CSA		SAL	
m	ϵ	CPU times	Iter.	CPU times	Iter.
50	10^{-3}	0.0012	5	0.0024	8
	10^{-4}	0.0018	10	0.0032	15
	10^{-5}	0.0021	18	0.0041	27
100	10^{-3}	0.0018	6	0.0027	9
	10^{-4}	0.0022	11	0.0039	16
	10^{-5}	0.0036	20	0.0057	30
500	10^{-3}	0.3019	7	0.4019	10
	10^{-4}	0.5169	12	0.7019	19
	10^{-5}	0.9674	22	1.3214	34
2000	10^{-3}	0.5436	7	0.7503	11
	10^{-4}	0.9746	13	1.3976	20
	10^{-5}	1.8324	24	2.5864	36

Table 5. Comparision of 3-CSA and SAL

Next, we compare 3-CSA with the Subgradient Algorithm (SGA), the Armijo Line Search Algorithm (ALS), the Ergodic Algorithm (EDA) [3]. In EDA, we choose $\lambda_k = \frac{1}{k}$ for all $k \geq 1$. The parameters for the remaining algorithms are selected as in Example 4.2. The comparisons results are presented in Figure 1. As we can see, 3-CSA shows a better behavior in terms of the computational time.

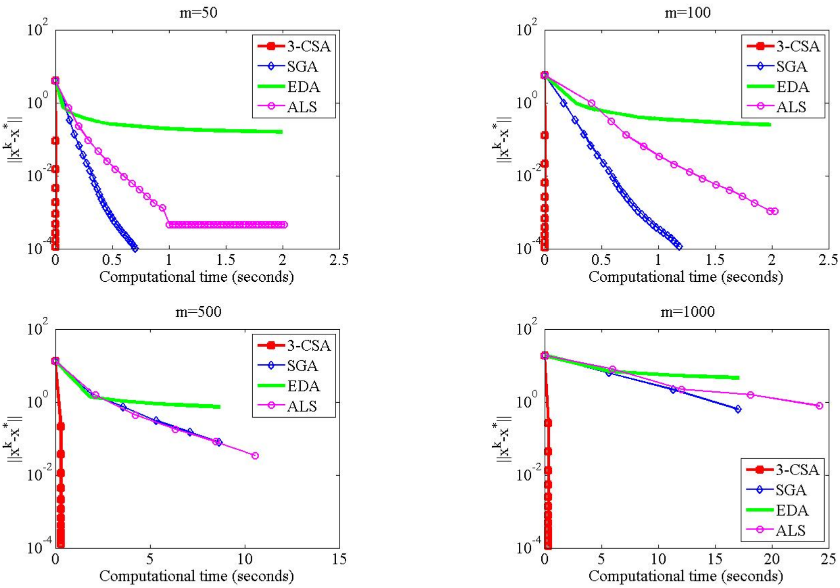

FIGURE 1. Comparisions of 3-SCA with some existing algorithms in Example 4.3

5. Conclusion

In this paper, we have proposed a three-component splitting algorithm for solving equilibrium problems in Hilbert spaces. Under the assumptions that the involving bifunction is strongly pseudomonotone and the component bifunctions satisfy suitable conditions, we have proved that the proposed algorithm strongly converges to the unique solution of the problem. Our algorithm is particularly effective when applied to equilibrium problems with complicated bifunctions, given as the sum of three components. The effectiveness of the proposed algorithm has been tested by some numerical experiments and comparisons. Also, the new algorithm has been applied to the Nash-Cournot oligopolistic equilibrium model for electricity markets with a non-convex cost function.

Acknowledgements. The authors would like to thank the anonymous referees for their remarks which helped to significantly improve the paper.

References

[1] Anh, P.K., Hai, T.N., *Splitting extragradient-like algorithms for strongly pseudomonotone equilibrium problems*, Numer. Algorithms, **76**(2017), 67-91.

[2] Anh, P.K., Hai, T.N., *A splitting algorithm for equilibrium problem given by the difference of two bifunctions*, J. Fixed. Point Theory Appl., **20**(2018), 1-15.

[3] Anh, P.N., Hai, T.N., Tuan, P.M., *On ergodic algorithms for equilibrium problems*, J. Global Optim., **64**(2016), 179-195.

[4] Bello Cruz, J.Y., Millán, R.D., *A direct splitting method for nonsmooth variational inequalities*, J. Optim. Theory Appl., **161**(2014), 729-737.

[5] Blum, E., Oettli, W., *From optimization and variational inequality to equilibrium problems*, Math. Student, **63**(1994), 127-149.

[6] Briceño-Arias, L.M., *A Douglas-Rachford splitting method for solving equilibrium problems*, Nonlinear Anal., **75**(2012), 6053-6059.

[7] Damek, D., Wotao, Y., *A Three-operator splitting scheme and its optimization applications*, Set-Valued Var. Anal., **25**(2017), 829-858.

[8] Dinh, B.V., Muu, L.D., *A projection algorithm for solving pseudomonotone equilibrium problems and it's application to a class of bilevel equilibria*, Optimization, **64**(2015), no. 3, 559-575.

[9] Dong, N.T.P., Strodiot, J.J., Van, N.T.T., Hien, N.V., *A family of extragradient methods for solving equilibrium problems*, J. Ind. Manag. Optim., **11**(2015), 619-630.

[10] Konnov, I.V., *Combined Relaxation Methods for Variational Inequalities*, Springer, Berlin, 2000.

[11] Lions, P.L., Mercier, B., *Splitting algorithms for the sum of two nonlinear operators*, SIAM J. Numer. Anal., **16**(1979), 964-979.

[12] Moudafi, A., *On the convergence of splitting proximal methods for equilibrium problems in Hilbert spaces*, J. Math. Anal. Appl., **359**(2009), 508-513.

[13] Muu, L.D., Quy, N.V., *On existence and solution methods for strongly pseudomonotone equilibrium problems*, Vietnam J. Math., **43**(2015), 229-238.

[14] Quoc, T.D., Anh, P.N., Muu, L.D., *Dual extragradient algorithms extended to equilibrium problems*, J. Global. Optim., **52**(2012), no. 1, 139-159.

[15] Quoc, T.D., Muu, L.D., Nguyen, V.H., *Extragradient algorithms extended to equilibrium problems*, Optimization, **57**(2008), no. 6, 749-776.

[16] Rockafellar, R.T., *Convex Analysis*, Princeton University Press, Princeton, New Jersey, 1970.

[17] Santos, P., Scheimberg, S., *An inexact subgradient algorithm for equilibrium problems*, Comput. Appl. Math., **30**(2011), 91-107.

[18] Xu, H.K., *Iterative algorithms for nonlinear operators*, J. Lond. Math. Soc., **66**(2002), no. 2, 240-256.

Trinh Ngoc Hai
Hanoi University of Science and Technology,
School of Applied Mathematics and Informatics,
1st, Dai Co Viet Street, Hai Ba Trung, Hanoi, Vietnam
e-mail: `hai.trinhngoc@hust.edu.vn`

Ngo Thi Thuong
HUS, VNU University of Science,
Department of Mathematics - Mechanics - Informatics,
334, Nguyen Trai Street, Thanh Xuan, HaNoi, Vietnam
e-mail: `ngothithuong.k56@hus.edu.vn`