Yan and Wen *Journal of Inequalities and Applications* (2016) 2016:108
DOI 10.1186/s13660-016-1047-7

Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

A new parallel splitting augmented Lagrangian-based method for a Stackelberg game

Xihong Yan* and Ruiping Wen

*Correspondence:
yanxh.tynu@gmail.com
Higher Education Key Laboratory of Engineering and Scientific Computing, Taiyuan Normal University, Taiyuan, Shanxi 030012, P.R. China

Abstract

This paper addresses a novel solution scheme for a special class of variational inequality problems which can be applied to model a Stackelberg game with one leader and three or more followers. In the scheme, the leader makes his decision first and then the followers reveal their choices simultaneously based on the information of the leader's strategy. Under mild conditions, we theoretically prove that the scheme can obtain an equilibrium. The proposed approach is applied to solve a simple game and a traffic problem. Numerical results about the performance of the new method are reported.

Keywords: Stackelberg game; variational inequality; separable structure; convergence; parallel splitting method

1 Introduction

In our work, we concentrate on typical structured variational inequality problems which are mathematically described as follows: Find a point $w^* \in \Omega$ such that

$$Q(w^*)^T(w - w^*) \geq 0, \quad \forall w \in \Omega,$$

where

$$w = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}, \qquad Q(w) = \begin{pmatrix} f_1(x_1) \\ f_2(x_2) \\ \vdots \\ f_m(x_m) \end{pmatrix}, \qquad \Omega = \left\{ (x_1, x_2, \ldots, x_m) \middle| x_i \in \mathcal{X}_i, \sum_{i=1}^{m} A_i x_i = b \right\},$$

$\mathcal{X}_i \subseteq \mathcal{R}^{n_i}$, and $f_i : \mathcal{X}_i \to \mathcal{R}^{n_i}$ $(i = 1, 2, \ldots, m)$ are mappings. In the work, we assume that $\mathcal{X}_i$ $(i = 1, 2, \ldots, m)$ are nonempty, closed, and convex sets, the mappings f_i $(i = 1, 2, \ldots, m)$ are monotone, and the solution of the problem exists. By attaching a Lagrange multiplier vector $\lambda \in R^l$ to the linear constraint $\sum_{i=1}^{m} A_i x_i = b$, the above problem can be converted to the following form: Find a point $u^* \in \mathcal{U} = \prod_{i=1}^{m} \mathcal{X}_i \times R^l$ such that

$$F(u^*)^T(u - u^*) \geq 0, \quad \forall u \in \mathcal{U}, \tag{1.1}$$

where

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \\ \lambda \end{pmatrix} \quad \text{and} \quad F(u) = \begin{pmatrix} f_1(x_1) - A_1^T\lambda \\ f_2(x_2) - A_2^T\lambda \\ \vdots \\ f_m(x_m) - A_m^T\lambda \\ \sum_{i=1}^m A_i x_i - b \end{pmatrix}. \tag{1.2}$$

The structured system (1.1)-(1.2) can be viewed as a mathematical formulation of a one-leader-m-follower Stackelberg game where ith follower controls his decision x_i, $i = 1, 2, \ldots, m$. For the special case where the game has one leader and two followers, that is, $m = 2$, there is extensive literature on numerical algorithms [1–5]. Although the one-leader-two-follower game is helpful as a baseline model in analyzing games and their equilibrium-seeking algorithms, many application problems involve three or more followers, that is, they are formulated as the problem (1.1)-(1.2) with $m \geq 3$. However, there is less work that focuses on designing algorithms for the general case $m \geq 3$ compared to a considerable number of studies on the special case $m = 2$.

For the general case of the problem (1.1)-(1.2), we can directly employ the classical augmented Lagrangian method proposed by [6]. However, the classical augmented Lagrangian method may prevent us from enjoying the separable structure of problem (1.1)-(1.2), since the subproblems obtained in the steps of the method involve coupled variables. Frequently used and powerful ideas for dealing with the difficulty are decomposition techniques. Based on the techniques, there are two most noteworthy categories of decomposition methods for the problem (1.1)-(1.2) in the literature: the alternating direction method and the parallel splitting augmented Lagrangian method. The alternating direction method has obtained recognition as a benchmark method to solve the problem (1.1)-(1.2), ever since it was proposed by Gabay and Mercier [7]. The application of the solution method has been extended in the past decade to cover a variety of areas [8–10]. The basic iterative scheme of the method for solving the problem (1.1)-(1.2) is as follows: When a decision $u^k = (x_1^k, x_2^k, \ldots, x_m^k, \lambda^k)$ is provided, the method gets $x_1^{k+1}, x_2^{k+1}, \ldots, x_m^{k+1}$ by solving the following system:

$$\begin{cases} (x_1' - x_1)^T \{f_1(x_1) - A_1^T[\lambda^k - H(A_1x_1 + \sum_{j=2}^m A_j x_j^k - b)]\} \geq 0, \quad \forall x_1' \in \mathcal{X}_1, \\ \cdots \\ (x_i' - x_i)^T \{f_i(x_i) - A_i^T[\lambda^k - H(\sum_{j=1}^{i-1} A_j x_j^{k+1} + A_i x_i + \sum_{j=i+1}^m A_j x_j^k - b)]\} \geq 0, \\ \quad \forall x_i' \in \mathcal{X}_i, \\ \cdots \\ (x_m' - x_m)^T \{f_m(x_m) - A_m^T[\lambda^k - H(\sum_{j=1}^{m-1} A_j x_j^{k+1} + A_m x_m - b)]\} \geq 0, \\ \quad \forall x_m' \in \mathcal{X}_m. \end{cases} \tag{1.3}$$

Then update λ by equation (1.4),

$$\lambda^{k+1} = \lambda^k - H\left(\sum_{i=1}^m A_i x_i^{k+1} - b\right), \tag{1.4}$$

where H is a symmetric positive definite matrix.

Note that the alternating direction method has been an attractive approach in that it successfully employs the Gauss-Seidel decomposition technique. However, in the method, each follower reveals his strategies sequentially, that is, the decision x_i^{k+1} of the follower i is revealed only when his former followers' strategies $x_1^{k+1}, \ldots, x_{i-1}^{k+1}$ are available, which is not reasonable for the case where each follower is blind with the others' strategies.

As we mentioned, the other efficient method to deal with the problem (1.1)-(1.2) is so-called the parallel splitting augmented Lagrangian method proposed by He [11] based on Jacobian decomposition. Compared to the alternating direction method, the speciality of the parallel splitting augmented Lagrangian method is that it simultaneously solves the following subproblems to obtain x_i^{k+1} $(i = 1, 2, \ldots, m)$:

$$\begin{aligned} &\left(x_i' - x_i\right)^T \left\{f_i(x_i) - A_i^T \left[\lambda^k - H\left(\sum_{j \neq i} A_j x_j^k + A_i x_i - b\right)\right]\right\} \geq 0, \\ &\forall x_i' \in \mathcal{X}_i, i = 1, 2, \ldots, m, \end{aligned} \tag{1.5}$$

which means that the followers make their choices simultaneously. However, there are few studies to address the convergence of the scheme (1.5)-(1.4), which results in improved methods where the output provided by (1.5) is corrected by a further correction step [12–15]. All these methods are designed by adding a correction step as follows:

$$u^{k+1} = u^k - \alpha_k d\left(u^k - \tilde{u}^k\right), \tag{1.6}$$

where α_k is a stepsize, $\tilde{u}^k$ is output of (1.5) which is called a predictor, and $-d(u^k - \tilde{u}^k)$ is a descent direction at u^k. The schemes may be understood in the context of a game in the following way: When the leader provides a decision $u^k = (x_1^k, \ldots, x_m^k, \lambda^k)$, all followers decide their strategies $\tilde{x}_i^k$ $(i = 1, 2, \ldots, m)$ simultaneously by solving the corresponding subproblems in (1.5), respectively. Then, based on the feedback $\tilde{x}_i^k$ $(i = 1, 2, \ldots, m)$ from the followers, the leader improves his strategy by (1.6).

It is noted that in the schemes (1.5)-(1.6), the leader controls all decision variables which is not realistic since in many practical problems the leader only has power on his own decision variables. Based on this identified research gap, the aim of this work is to devise a new method for the problem (1.1)-(1.2), that is, a mathematical formulation of a one-leader-m-follower Stackelberg game where the leader controls λ and the ith follower controls x_i. In the correction step of our method, only the leader's variable λ is improved. Furthermore, we provide insights on the convergence of our method and a computational study of its performance.

The rest of the paper is organized as follows. Section 2 gives preliminaries, such as definitions and notations, which will be useful for our analysis and ease of exposition. Section 3 presents the proposed method in detail. Section 4 conducts an analysis on the global convergence of the proposed method. In Section 5, we apply the method to solve some practical problems and report the corresponding computational results. Finally, conclusions and some future research directions are stated in Section 6.

2 Preliminaries

For the convenience of the analysis in the paper, this section provides some basic definitions and notations. An n-dimensional Euclidean space is denoted by $\mathcal{R}^n$. All vectors used in the paper mean column vectors. For ease of exposition, we use the vector $(x_1,\dots,x_m)$ to represent $(x_1^T,\dots,x_m^T)^T$, where T represents the transpose operator. $\delta_{\max}(A)$ denotes the largest eigenvalue of square matrix A. For any symmetric and positive definite matrix G, we denote by $\|x\|_G := \sqrt{x^TGx}$ its G-norm. In the work, we define

$$G = \begin{pmatrix} \alpha A_1^THA_1 & & & & \\ & \alpha A_2^THA_2 & & & \\ & & \ddots & & \\ & & & \alpha A_m^THA_m & \\ & & & & H^{-1} \end{pmatrix} \tag{2.1}$$

such that $\|u-u^*\|_G^2 := \|A_1x_1 - A_1x_1^*\|_{\alpha H}^2 + \cdots + \|A_mx_m - A_mx_m^*\|_{\alpha H}^2 + \|\lambda - \lambda^*\|_{H^{-1}}^2$.

Definition 2.1

(a) A mapping $f : \mathcal{R}^n \to \mathcal{R}^n$ is described as a monotone function, if

$$(x_1 - x_2)^T\bigl(f(x_1) - f(x_2)\bigr) \geq 0, \quad \forall x_1, x_2 \in \mathcal{R}^n.$$

(b) A mapping $f : \mathcal{R}^n \to \mathcal{R}^n$ is described as a strongly monotone function with modulus $\mu > 0$, if

$$(x_1 - x_2)^T\bigl(f(x_1) - f(x_2)\bigr) \geq \mu\|x_1 - x_2\|^2, \quad \forall x_1, x_2 \in \mathcal{R}^n.$$

In the paper, there is a basic assumption that the mappings f_i $(i = 1,2,\dots,m)$ are continuous and strongly monotone with modulus μ_{f_i}, respectively.

3 Parallel method

In this section, we formally state the procedure of the new parallel splitting method for solving the problem (1.1)-(1.2) and provide some insights on the method's properties.

Algorithm 3.1 (A new augmented Lagrangian-based parallel splitting method)

S0. Select an initial point $u^0 = (x_1^0,\dots,x_m^0,\lambda^0) \in \mathcal{U}$, $\epsilon > 0$, $\alpha > 0$, and H, and set $k = 0$.

S1. Simultaneously obtain solutions x_i^{k+1} $(i = 1,2,\dots,m)$ by solving the following variational inequalities:

$$x_i \in \mathcal{X}_i, \quad \bigl(x_i' - x_i\bigr)^T\left\{f_i(x_i) - A_i^T\left[\lambda^k - H\left(\sum_{j\neq i} A_jx_j^k + A_ix_i - b\right)\right]\right\} \geq 0,$$
$$\forall x_i' \in \mathcal{X}_i, \tag{3.1}$$

respectively. Then set

$$\tilde{\lambda}^k = \lambda^k - H\left(\sum_{i=1}^m A_ix_i^{k+1} - b\right). \tag{3.2}$$

S2. Update λ^{k+1} through equation (3.3),

$$\lambda^{k+1} = \lambda^k - \alpha\bigl(\lambda^k - \tilde{\lambda}^k\bigr). \tag{3.3}$$

S3. If

$$\max\left\{\max_i \left\|A_i x_i^k - A_i x_i^{k+1}\right\|, \left\|\lambda^k - \lambda^{k+1}\right\|\right\} \le \epsilon, \tag{3.4}$$

stop. Otherwise, set $k = k + 1$ and go to S1.

Remark 3.1 Now, we conduct some analysis on the proposed algorithm. From (3.2) and (3.3), we can deduce that $\sum_{i=1}^m A_i x_i^{k+1} = b$ if $\lambda^{k+1} = \lambda^k$. Then we have $\sum_{j\neq i} A_j x_j^k + A_i x_i^{k+1} - b = 0$ under the condition that $A_i x_i^{k+1} = A_i x_i^k$ $(i = 1,2,\ldots,m)$. Furthermore, according to (3.1), there exist the following inequalities and equation:

$$x_i^{k+1} \in \mathcal{X}_i, \quad \left(x_i' - x_i^{k+1}\right)^T \left\{f_i\left(x_i^{k+1}\right) - A_i^T \lambda^{k+1}\right\} \ge 0, \quad \forall x_i' \in \mathcal{X}_i, i = 1,2,\ldots,m, \tag{3.5}$$

and

$$\sum_{i=1}^m A_i x_i^{k+1} = b.$$

Thus, it is obvious that $(x_1^{k+1},\ldots,x_m^{k+1},\lambda^{k+1}) \in \mathcal{U}$ is a solution of the one-leader-m-follower game. Based on the analysis, we conclude that for a given small enough ϵ, the proposed method with termination condition (3.4) can obtain an approximation solution $(x_1^{k+1},\ldots,x_m^{k+1},\lambda^{k+1}) \in \mathcal{U}$ for the concerned game, that is, the stopping criterion (3.4) in the method is reasonable.

Remark 3.2 It is obvious that our proposed algorithm falls into the parallel splitting method since all the subproblems (3.1) can be solved in parallel by many existing efficient algorithms. Moreover, the proposed algorithm makes the best of the separable characteristic of the concerned problem (1.1)-(1.2) since only one function is involved in each subproblem. In addition, the proposed algorithm is a prediction-correction parallel splitting method. But the most significant difference from others is that only λ is corrected, which leads to less computational cost.

4 Convergence result

The convergence property of our parallel splitting algorithm is given in this section. First, we give a lemma that is useful for the convergence result.

Lemma 4.1 *Given λ^k by the leader and x_i^k by the ith follower $(i = 1,2,\ldots,m)$ at iteration k, the strategy $(x_1^{k+1},\ldots,x_m^{k+1},\lambda^{k+1})$ in the next iteration satisfies*

$$\begin{aligned}
&\sum_{i=1}^m \left\|A_i x_i^{k+1} - A_i x_i^*\right\|_{\alpha H}^2 + \left\|\lambda^{k+1} - \lambda^*\right\|_{H^{-1}}^2 \\
&\quad\le \sum_{i=1}^m \left\|A_i x_i^k - A_i x_i^*\right\|_{\alpha H}^2 + \left\|\lambda^k - \lambda^*\right\|_{H^{-1}}^2 - \alpha \sum_{i=1}^m \left\|\sum_{j\neq i} A_j x_j^{k+1} + A_i x_i^k - b\right\|_H^2 \\
&\qquad + \alpha \sum_{i=1}^m \left[(\alpha + m - 2)\delta_{\max}\left(A_i^T H A_i\right) - 2\mu_{f_i}\right]\left\|x_i^{k+1} - x_i^*\right\|^2.
\end{aligned}$$

Proof We assume that there exists an optimal solution $u^* = (x_1^*, \dots, x_m^*, \lambda^*) \in \mathcal{U}$. Using (1.1) with $u = u^{k+1} = (x_1^{k+1}, \dots, x_m^{k+1}, \lambda^{k+1}) \in \mathcal{U}$, we have

$$F(u^*)^T(u^{k+1} - u^*) \ge 0. \tag{4.1}$$

On the other hand, let $x_i' = x_i^*$ ($i = 1, 2, \dots, m$) in each inequality of (3.1). It is easy to obtain

$$\begin{aligned}&(x_i^* - x_i^{k+1})^T \biggl\{ f_i(x_i^{k+1}) - A_i^T\lambda^k + A_i^T H\biggl(\sum_{j\neq i} A_j x_j^k + A_i x_i^{k+1} - b\biggr)\biggr\} \ge 0,\\ &\quad i = 1, 2, \dots, m.\end{aligned} \tag{4.2}$$

From the summation of all the inequalities included in (4.1) and (4.2) and the equality $\sum_{i=1}^m A_i x_i^* = b$, we have

$$\sum_{i=1}^m (x_i^{k+1} - x_i^*)^T \biggl\{ f_i(x_i^*) - f_i(x_i^{k+1}) + A_i^T(\lambda^k - \lambda^*) - A_i^T H\biggl(\sum_{j\neq i} A_j x_j^k + A_i x_i^{k+1} - b\biggr)\biggr\} \ge 0.$$

Rearranging the above inequality and taking account of the strong monotonicity of all the functions f_i ($i = 1, 2, \dots, m$) and $\sum_{i=1}^m A_i x_i^* = b$, we obtain

$$\begin{aligned}&(\lambda^k - \lambda^*)^T \Biggl(\sum_{i=1}^m A_i x_i^{k+1} - b\Biggr)\\ &\quad\ge \sum_{i=1}^m \biggl[(A_i x_i^{k+1} - A_i x_i^*)^T H\biggl(\sum_{j\neq i} A_j x_j^k + A_i x_i^{k+1} - b\biggr)\biggr] + \sum_{i=1}^m \mu_{f_i} \|x_i^{k+1} - x_i^*\|^2.\end{aligned}$$

Note that b can be replaced by $\sum_{i=1}^m A_i x_i^*$. Then

$$\begin{aligned}&(\lambda^k - \lambda^*)^T \Biggl(\sum_{i=1}^m A_i x_i^{k+1} - b\Biggr)\\ &\quad\ge \sum_{i=1}^m \Biggl[(A_i x_i^{k+1} - A_i x_i^*)^T H\Biggl(\sum_{j\neq i} A_j x_j^k + A_i x_i^{k+1} - \sum_{j=1}^m A_j x_j^*\Biggr)\Biggr]\\ &\qquad + \sum_{i=1}^m \mu_{f_i} \|x_i^{k+1} - x_i^*\|^2\\ &\quad= \sum_{i=1}^m \biggl[(A_i x_i^{k+1} - A_i x_i^*)^T H\biggl(\sum_{j\neq i}(A_j x_j^k - A_j x_j^*) + A_i x_i^{k+1} - A_i x_i^*\biggr)\biggr]\\ &\qquad + \sum_{i=1}^m \mu_{f_i} \|x_i^{k+1} - x_i^*\|^2\\ &\quad= \sum_{i=1}^m \|A_i x_i^{k+1} - A_i x_i^*\|_H^2 + \sum_{i=1}^m \sum_{j\neq i} (A_i x_i^{k+1} - A_i x_i^*)^T H(A_j x_j^k - A_j x_j^*)\\ &\qquad + \sum_{i=1}^m \mu_{f_i} \|x_i^{k+1} - x_i^*\|^2\end{aligned}$$

$$
\begin{aligned}
&= \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 + \sum_{i=1}^{m} \sum_{j \neq i} \left(A_i x_i^{k+1} - A_i x_i^*\right)^T H \left(A_j x_j^k - A_j x_j^{k+1}\right) \\
&\quad + \sum_{i=1}^{m} \mu_{f_i} \left\| x_i^{k+1} - x_i^* \right\|^2. \qquad (4.3)
\end{aligned}
$$

Thus,

$$
\begin{aligned}
&\left\| \lambda^{k+1} - \lambda^* \right\|_{H^{-1}}^2 \\
&\quad = \left\| \lambda^k - \lambda^* - \alpha H \left(\sum_{i=1}^{m} A_i x_i^{k+1} - b \right) \right\|_{H^{-1}}^2 \\
&\quad = \left\| \lambda^k - \lambda^* \right\|_{H^{-1}}^2 - 2\alpha \left(\lambda^k - \lambda^*\right)^T \left(\sum_{i=1}^{m} A_i x_i^{k+1} - b \right) + \alpha^2 \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 \\
&\quad \leq \left\| \lambda^k - \lambda^* \right\|_{H^{-1}}^2 - 2\alpha \sum_{i=1}^{m} \mu_{f_i} \left\| x_i^{k+1} - x_i^* \right\|^2 + \alpha^2 \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 \\
&\qquad - 2\alpha \left\{ \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 + \sum_{i=1}^{m} \sum_{j \neq i} \left(A_i x_i^{k+1} - A_i x_i^*\right)^T H \left(A_j x_j^k - A_j x_j^{k+1}\right) \right\}. \qquad (4.4)
\end{aligned}
$$

Now, we focus on the terms $\|A_i x_i^{k+1} - A_i x_i^*\|_{\alpha H}^2$ $(i = 1, 2, \ldots, m)$.

Since

$$
\begin{aligned}
&\left(A_i x_i^k - A_i x_i^*\right)^T H \left(A_i x_i^k - A_i x_i^{k+1}\right) \\
&\quad = \left(A_i x_i^{k+1} - A_i x_i^*\right)^T H \left(A_i x_i^k - A_i x_i^{k+1}\right) + \left\| A_i x_i^k - A_i x_i^{k+1} \right\|_H^2, \quad i = 1, 2, \ldots, m,
\end{aligned}
$$

it follows that

$$
\begin{aligned}
&\left\| A_i x_i^{k+1} - A_i x_i^* \right\|_{\alpha H}^2 \\
&\quad = \left\| A_i x_i^k - A_i x_i^* \right\|_{\alpha H}^2 + \left\| A_i x_i^{k+1} - A_i x_i^k \right\|_{\alpha H}^2 - 2\alpha \left\| A_i x_i^k - A_i x_i^{k+1} \right\|_H^2 \\
&\qquad - 2\alpha \left(A_i x_i^{k+1} - A_i x_i^*\right)^T H \left(A_i x_i^k - A_i x_i^{k+1}\right), \quad i = 1, 2, \ldots, m. \qquad (4.5)
\end{aligned}
$$

Adding all formulas in (4.4) and (4.5), we get the following inequality:

$$
\begin{aligned}
&\sum_{i=1}^{m} \left\| A_i x_i^{k+1} - A_i x_i^* \right\|_{\alpha H}^2 + \left\| \lambda^{k+1} - \lambda^* \right\|_{H^{-1}}^2 \\
&\quad \leq \sum_{i=1}^{m} \left\| A_i x_i^k - A_i x_i^* \right\|_{\alpha H}^2 + \left\| \lambda^k - \lambda^* \right\|_{H^{-1}}^2 \\
&\qquad + \sum_{i=1}^{m} \left\| A_i x_i^{k+1} - A_i x_i^k \right\|_{\alpha H}^2 + \alpha^2 \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 \\
&\qquad - 2\alpha \left\{ \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 + \sum_{i=1}^{m} \sum_{j \neq i} \left(A_i x_i^{k+1} - A_i x_i^*\right)^T H \left(A_j x_j^k - A_j x_j^{k+1}\right)
\end{aligned}
$$

$$\begin{aligned}
&\quad + \sum_{i=1}^{m} \left(A_i x_i^{k+1} - A_i x_i^*\right)^T H \left(A_i x_i^k - A_i x_i^{k+1}\right) + \sum_{i=1}^{m} \left\| A_i x_i^k - A_i x_i^{k+1} \right\|_H^2 \Bigg\} \\
&\quad - 2\alpha \sum_{i=1}^{m} \mu_{f_i} \left\| x_i^{k+1} - x_i^* \right\|^2 \\
&= \sum_{i=1}^{m} \left\| A_i x_i^k - A_i x_i^* \right\|_{\alpha H}^2 + \left\| \lambda^k - \lambda^* \right\|_{H^{-1}}^2 + \alpha^2 \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 \\
&\quad - 2\alpha \left\{ \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 + \sum_{i=1}^{m} \sum_{j=1}^{m} \left(A_i x_i^{k+1} - A_i x_i^*\right)^T H \left(A_j x_j^k - A_j x_j^{k+1}\right) \right\} \\
&\quad - 2\alpha \sum_{i=1}^{m} \mu_{f_i} \left\| x_i^{k+1} - x_i^* \right\|^2 - \sum_{i=1}^{m} \left\| A_i x_i^{k+1} - A_i x_i^k \right\|_{\alpha H}^2 \\
&= \sum_{i=1}^{m} \left\| A_i x_i^k - A_i x_i^* \right\|_{\alpha H}^2 + \left\| \lambda^k - \lambda^* \right\|_{H^{-1}}^2 + \alpha^2 \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 \\
&\quad - 2\alpha \sum_{i=1}^{m} \mu_{f_i} \left\| x_i^{k+1} - x_i^* \right\|^2 - \alpha \Bigg\{ \sum_{i=1}^{m} \left\| A_i x_i^{k+1} - A_i x_i^k \right\|_H^2 \\
&\quad + 2 \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 + 2 \sum_{j=1}^{m} \left(\sum_{i=1}^{m} A_i x_i^{k+1} - b \right)^T H \left(A_j x_j^k - A_j x_j^{k+1}\right) \Bigg\} \\
&= \sum_{i=1}^{m} \left\| A_i x_i^k - A_i x_i^* \right\|_{\alpha H}^2 + \left\| \lambda^k - \lambda^* \right\|_{H^{-1}}^2 + \alpha(\alpha + m - 2) \left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 \\
&\quad - \alpha \sum_{i=1}^{m} \left\| \sum_{j \neq i} A_j x_j^{k+1} + A_i x_i^k - b \right\|_H^2 - 2\alpha \sum_{i=1}^{m} \mu_{f_i} \left\| x_i^{k+1} - x_i^* \right\|^2.
\end{aligned} \tag{4.6}$$

Since

$$\begin{aligned}
\left\| \sum_{i=1}^{m} A_i x_i^{k+1} - b \right\|_H^2 &\leq \sum_{i=1}^{m} \left\| A_i x_i^{k+1} - A_i x_i^* \right\|_H^2 \\
&\leq \sum_{i=1}^{m} \delta_{\max}\left(A_i^T H A_i\right) \left\| x_i^{k+1} - x_i^* \right\|^2,
\end{aligned} \tag{4.7}$$

from (4.6), we get

$$\begin{aligned}
&\sum_{i=1}^{m} \left\| A_i x_i^{k+1} - A x_i^* \right\|_{\alpha H}^2 + \left\| \lambda^{k+1} - \lambda^* \right\|_{H^{-1}}^2 \\
&\leq \sum_{i=1}^{m} \left\| A_i x_i^k - A x_i^* \right\|_{\alpha H}^2 + \left\| \lambda^k - \lambda^* \right\|_{H^{-1}}^2 - \alpha \sum_{i=1}^{m} \left\| \sum_{j \neq i} A_j x_j^{k+1} + A_i x_i^k - b \right\|_H^2 \\
&\quad + \alpha \sum_{i=1}^{m} \left[(\alpha + m - 2) \delta_{\max}\left(A_i^T H A_i\right) - 2\mu_{f_i} \right] \left\| x_i^{k+1} - x_i^* \right\|^2.
\end{aligned}$$

The proof is completed. □

Lemma 4.1 indicates that

$$\begin{aligned}\|u^{k+1}-u^*\|_G^2 &\leq \|u^k-u^*\|_G^2+\alpha\sum_{i=1}^{m}\bigl[(\alpha+m-2)\delta_{\max}\bigl(A_i^THA_i\bigr)-2\mu_{f_i}\bigr]\|x_i^{k+1}-x_i^*\|^2\\ &\quad -\alpha\sum_{i=1}^{m}\Bigl\|\sum_{j\neq i}A_jx_j^{k+1}+A_ix_i^k-b\Bigr\|_H^2,\end{aligned}\tag{4.8}$$

where G is defined by (2.1).

Based on the above analysis, the global convergence of the proposed method is presented in the following theorem.

Theorem 4.2 *Let m be the number of followers. Suppose that for each $i\in\{1,2,\dots,m\}$, $f_i(x_i)$ is continuous and strongly monotone on $\mathcal{X}_i\subseteq\mathcal{R}^{n_i}$. Moreover, if*

$$u_{f_i}>\frac{(m-2)\delta_{\max}(A_i^THA_i)}{2},\quad i=1,2,\dots,m,$$

and

$$0<\alpha\leq\min_i\left\{\frac{2\mu_{f_i}}{\delta_{\max}(A_i^THA_i)}-(m-2)\right\},$$

the sequence $\{u^k\}$ generated by the proposed method converges to an optimal solution of the problem (1.1)-(1.2).

Proof From Lemma 4.1 and (4.8), we have

$$\begin{aligned}&\|u^{k+1}-u^*\|_G^2-\|u^k-u^*\|_G^2\\ &\quad\leq-\alpha\sum_{i=1}^{m}\bigl[2\mu_{f_i}-(\alpha+m-2)\delta_{\max}\bigl(A_i^THA_i\bigr)\bigr]\|x_i^{k+1}-x_i^*\|^2\\ &\qquad-\alpha\sum_{i=1}^{m}\Bigl\|\sum_{j\neq i}A_jx_j^{k+1}+A_ix_i^k-b\Bigr\|_H^2.\end{aligned}\tag{4.9}$$

The two terms on the right side of the above inequality are negative due to the conditions of the theorem. Thus,

$$\|u^{k+1}-u^*\|_G^2\leq\|u^k-u^*\|_G^2\leq\cdots\leq\|u^0-u^*\|_G^2\leq+\infty,$$

which means that $\{u^k\}$ is a bounded sequence generated by the developed method. Consequently,

$$\begin{aligned}&\alpha\sum_{k=1}^{+\infty}\sum_{i=1}^{m}\bigl[2\mu_{f_i}-(\alpha+m-2)\delta_{\max}\bigl(A_i^THA_i\bigr)\bigr]\|x_i^{k+1}-x_i^*\|^2\\ &\quad+\alpha\sum_{k=1}^{+\infty}\sum_{i=1}^{m}\Bigl\|\sum_{j\neq i}A_jx_j^{k+1}+A_ix_i^k-b\Bigr\|_H^2<\sum_{k=1}^{+\infty}\bigl(\|u^k-u^*\|_G^2-\|u^{k+1}-u^*\|_G^2\bigr)<+\infty,\end{aligned}$$

which means that

$$\begin{aligned} &\lim_{k\to+\infty}\left\|x_i^{k+1}-x_i^*\right\|=0, \quad i=1,2,\ldots,m, \\ &\lim_{k\to+\infty}\left\|\sum_{j\neq i}A_jx_j^{k+1}+A_ix_i^k-b\right\|_H=0, \quad i=1,2,\ldots,m. \end{aligned} \tag{4.10}$$

Thus,

$$\sum_{i=1}^{m}A_ix_i^*=b. \tag{4.11}$$

On the other hand, for each $i\in\{1,2,\ldots,m\}$, x_i^{k+1} satisfies the following inequality:

$$\left(x_i'-x_i^{k+1}\right)^T\left\{f_i\left(x_i^{k+1}\right)-A_i^T\lambda^k+A_i^TH\left(\sum_{j\neq i}A_jx_j^k+A_ix_i^{k+1}-b\right)\right\}\geq 0, \quad \forall x_i'\in\mathcal{X}_i. \tag{4.12}$$

Moreover, there exist cluster points for the sequence $\{\lambda^k\}$ due to the boundedness of $\{\lambda^k\}$ implied by the boundedness of $\{u^k\}$. Let λ^* be one of cluster points such that it is a limit of a convergent subsequence $\{\lambda^{k_j}\}$. From the limit of (4.12) along this subsequence, it follows that

$$\left(x_i'-x_i^*\right)^T\left\{f_i\left(x_i^*\right)-A_i^T\lambda^*\right\}\geq 0, \quad \forall x_i'\in\mathcal{X}_i, i=1,2,\ldots,m. \tag{4.13}$$

From (4.13) and (4.11), we can assert that the sequence generated by the proposed method is globally convergent. This completes the proof. □

5 Numerical experiments

In this section, we present some numerical results by implementing our proposed algorithm in a game and a traffic problem, which demonstrate the application and performance of our proposed algorithm. All tests are performed in a MATLAB environment on a PC with Intel Core 2 Duo 3.10GHz CPU and 4GB of RAM. The section is organized as follows: First, we provide strategies for players in a game to verify the application of our algorithm. Second, we investigate the performance of the proposed algorithm by comparing it with one existing algorithm for solving a generic test problem.

Example 5.1 A simple game with one leader and three followers.

We begin the computational study by solving a one-leader-three-follower game where each follower i decides a pure strategy s_i ($i=1,2,3$). The problem is given by the following programs,

$$\begin{array}{llllll} \min & -s_1 & \min & (s_2-0.5)^2 & \min & (s_3-1.5)^2 \\ \text{s.t.} & s_1+s_2+s_3=2 & \text{s.t.} & s_1+s_2+s_3=2 & \text{s.t.} & s_1+s_2+s_3=2 \\ & s_1\geq 0 & & s_2\geq 0 & & s_3\geq 0 \end{array}$$

The game is a revised version of a game considered by [16–18]. We solve the game using the proposed algorithm to obtain a solution which matches with the analytic optimal solution, that is, $s_1=1$, $s_2=0$, $s_3=1$. Furthermore, we investigate the effect of initial points

Table 1 Computational results for Example 5.1

ϵ	$u^0 = (0,0,0,0)$		$u^0 = (1,1,1,1)$		Random point	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
10^{-4}	30	0.0078	20	0.0052	22	0.0062
10^{-5}	40	0.0090	28	0.0070	31	0.0078
10^{-6}	57	0.0120	39	0.0090	42	0.0106

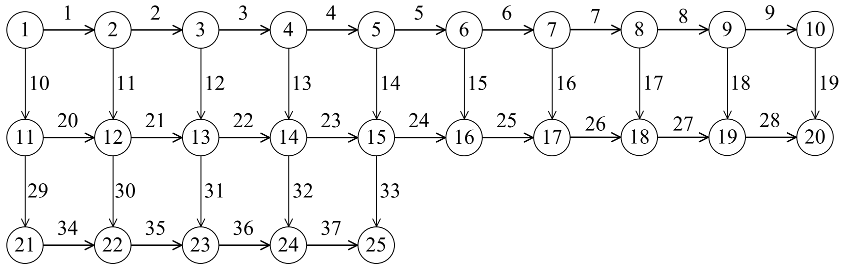

Figure 1 The network for Example 5.2.

and optimality tolerances on the performance of our algorithm. In the tests, initial points are $u_0 = (0,0,0,0)$, $u_0 = (1,1,1,1)$, or a random point in $(0,1)^4$. Optimality tolerances are $\epsilon = 10^{-4}$, $\epsilon = 10^{-5}$ or $\epsilon = 10^{-6}$. In our experiments, the parameter setting is $\alpha = 0.8$ and $H = 0.9I$. The associated numerical results are recorded in Table 1 where Iter. means the number of iterations and CPU means CPU time. The numerical results from Table 1 confirm the validity and efficiency of our method. Moreover, we observe that the proposed algorithm is robust to the initial points.

To further showcase the performance of the proposed algorithm, we use it to solve a generic test problem, a traffic equilibrium problem with fixed demand. Now, we introduce the problem briefly.

Example 5.2 A traffic equilibrium problem with fixed demand constraints.

The problem is always selected as a test case; for example, see [13, 19, 20]. Its network is shown in Figure 1 where there are 25 nodes, 37 links, and 55 paths. Other parameters and notations are summarized in Table 2.

We define variables x_p as the traffic flow of path p. Then arc flow vector f is calculated by the following formula:

$$f = A^T x, \qquad d_w = \sum_{p \in P_w} x_p, \quad \text{and} \quad d = Bx.$$

Moreover, based on the link travel cost vector denoted by $t(f) = \{t_a, a \in L\}$ whose expressions are given in Table 3, the travel cost vector θ can be formulated as follows:

$$\theta = At(f) = At\bigl(A^T x\bigr).$$

Hence, the problem is converted to a variational inequality as

$$\bigl(x - x^*\bigr)^T \theta(x) \geq 0, \quad \forall x \in S,$$

where $S = \{x \in \mathcal{R}^{55} | Bx = d, x \geq 0\}$.

Table 2 Parameter setting and notations for the network

Parameter	Description	Value
a	Link connecting two nodes	
L	Link set	
f_a	Link flow on link a	
f	Arc flow vector	
p	Path	
ω	O/D pairs	$\{\omega_1 = (1, 20), \omega_2 = (1, 25), \omega_3 = (1, 24), \omega_4 = (2, 20), \omega_5 = (3, 25), \omega_6 = (11, 25)\}$
P_ω	Set of the paths connecting O/D pair ω	
d_ω	Traffic amount between O/D pair ω	$d_1 = 10, d_2 = 20, d_3 = 20, d_4 = 55, d_5 = 100, d_6 = 30$
d	O/D pair traffic amount vector	
A	Path-arc incidence matrix	$A(i,j) = 1$ if the ith path contains link j, otherwise $A(i,j) = 0$, $A \in R^{55\times 37}$
B	Path-O/D pair incidence matrix	$B(i,j) = 1$ if the ith O/D pair contains path j, otherwise $B(i,j) = 0$, $B \in R^{6\times 55}$

Table 3 The link cost function $t_a(f)$ for Example 5.2

$t_1(f) = 5 \cdot 10^{-6} f_1^4 + 0.5f_1 + 0.2f_2 + 50$	$t_2(f) = 3 \cdot 10^{-6} f_2^4 + 0.4f_2 + 0.4f_1 + 20$
$t_3(f) = 5 \cdot 10^{-6} f_3^4 + 0.3f_3 + 0.1f_4 + 35$	$t_4(f) = 3 \cdot 10^{-6} f_4^4 + 0.6f_4 + 0.3f_5 + 40$
$t_5(f) = 6 \cdot 10^{-6} f_5^4 + 0.6f_5 + 0.4f_6 + 60$	$t_6(f) = 0.7f_6 + 0.3f_7 + 50$
$t_7(f) = 8 \cdot 10^{-6} f_7^4 + 0.8f_7 + 0.2f_8 + 40$	$t_8(f) = 4 \cdot 10^{-6} f_8^4 + 0.5f_8 + 0.2f_9 + 65$
$t_9(f) = 10^{-6} f_9^4 + 0.6f_9 + 0.2f_{10} + 70$	$t_{10}(f) = 0.4f_{10} + 0.1f_{12} + 80$
$t_{11}(f) = 7 \cdot 10^{-6} f_{11}^4 + 0.7f_{11} + 0.4f_{12} + 65$	$t_{12}(f) = 0.8f_{12} + 0.2f_{13} + 70$
$t_{13}(f) = 10^{-6} f_{13}^4 + 0.7f_{13} + 0.3f_{18} + 60$	$t_{14}(f) = 0.8f_{14} + 0.3f_{15} + 50$
$t_{15}(f) = 3 \cdot 10^{-6} f_{15}^4 + 0.9f_{15} + 0.2f_{14} + 20$	$t_{16}(f) = 0.8f_{16} + 0.5f_{12} + 30$
$t_{17}(f) = 3 \cdot 10^{-6} f_{17}^4 + 0.7f_{17} + 0.2f_{15} + 45$	$t_{18}(f) = 0.5f_{18} + 0.1f_{16} + 30$
$t_{19}(f) = 0.8f_{19} + 0.3f_{17} + 60$	$t_{20}(f) = 3 \cdot 10^{-6} f_{20}^4 + 0.6f_{20} + 0.1f_{21} + 30$
$t_{21}(f) = 4 \cdot 10^{-6} f_{21}^4 + 0.4f_{21} + 0.1f_{22} + 40$	$t_{22}(f) = 2 \cdot 10^{-6} f_{22}^4 + 0.6f_{22} + 0.1f_{23} + 50$
$t_{23}(f) = 3 \cdot 10^{-6} f_{23}^4 + 0.9f_{23} + 0.2f_{24} + 35$	$t_{24}(f) = 2 \cdot 10^{-6} f_{24}^4 + 0.8f_{24} + 0.1f_{25} + 40$
$t_{25}(f) = 3 \cdot 10^{-6} f_{25}^4 + 0.9f_{25} + 0.3f_{26} + 45$	$t_{26}(f) = 6 \cdot 10^{-6} f_{26}^4 + 0.7f_{26} + 0.8f_{27} + 30$
$t_{27}(f) = 3 \cdot 10^{-6} f_{27}^4 + 0.8f_{27} + 0.3f_{28} + 50$	$t_{28}(f) = 3 \cdot 10^{-6} f_{28}^4 + 0.7f_{28} + 65$
$t_{29}(f) = 3 \cdot 10^{-6} f_{29}^4 + 0.3f_{29} + 0.1f_{30} + 45$	$t_{30}(f) = 4 \cdot 10^{-6} f_{30}^4 + 0.7f_{30} + 0.2f_{31} + 60$
$t_{31}(f) = 3 \cdot 10^{-6} f_{31}^4 + 0.8f_{31} + 0.1f_{32} + 75$	$t_{32}(f) = 6 \cdot 10^{-6} f_{32}^4 + 0.8f_{32} + 0.3f_{33} + 65$
$t_{33}(f) = 4 \cdot 10^{-6} f_{33}^4 + 0.9f_{33} + 0.2f_{31} + 75$	$t_{34}(f) = 6 \cdot 10^{-6} f_{34}^4 + 0.7f_{34} + 0.3f_{30} + 55$
$t_{35}(f) = 3 \cdot 10^{-6} f_{35}^4 + 0.8f_{35} + 0.3f_{32} + 60$	$t_{36}(f) = 2 \cdot 10^{-6} f_{36}^4 + 0.8f_{36} + 0.4f_{31} + 75$
$t_{37}(f) = 6 \cdot 10^{-6} f_{37}^4 + 0.5f_{37} + 0.1f_{36} + 35$	

We implement our algorithm for this problem. First, the decision variable vector x is partitioned into three parts,

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix},$$

where $x_1 \in \mathcal{R}^{25}$, $x_2 \in \mathcal{R}^{15}$, and $x_3 \in \mathcal{R}^{15}$. Subsequently, matrices A, B, and θ are partitioned, respectively, as follows:

$$A = \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix}, \qquad B = (B_1 \quad B_2 \quad B_3), \quad \text{and} \quad \theta = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{pmatrix},$$

where $A_1 \in \mathcal{R}^{25\times 37}$, $A_2 \in \mathcal{R}^{15\times 37}$, $A_3 \in \mathcal{R}^{15\times 37}$, $B_1 \in \mathcal{R}^{6\times 25}$, $B_2 \in \mathcal{R}^{6\times 15}$, $B_3 \in \mathcal{R}^{6\times 15}$, and θ is partitioned in the same way as x.

Table 4 Computational results for Example 5.2

ϵ		u^0 = rand(61, 1)		u^0 = 50* ones(61, 1)		u^0 = 100* ones(61, 1)	
		Alg 3.1	Alg in [11]	Alg 3.1	Alg in [11]	Alg 3.1	Alg in [11]
10^{-4}	Iter.	122	160	146	-	153	-
	CPU (s)	0.1560	0.1872	0.1092	-	0.1248	-
10^{-5}	Iter.	153	177	168	-	175	-
	CPU (s)	0.1716	0.2340	0.1404	-	0.1560	-
10^{-6}	Iter.	183	196	190	-	197	-
	CPU (s)	0.2028	0.2496	0.1872	-	0.2028	-

Based on the above partitions, the resulting traffic problem is as follows:

$$\bigl(x_1 - x_1^*\bigr)^T \theta(x_1) \geq 0, \quad \forall x_1 \in S, \tag{5.1}$$

$$\bigl(x_2 - x_2^*\bigr)^T \theta(x_2) \geq 0, \quad \forall x_2 \in S, \tag{5.2}$$

and

$$\bigl(x_3 - x_3^*\bigr)^T \theta(x_3) \geq 0, \quad \forall x_3 \in S, \tag{5.3}$$

where $S = \{(x_1, x_2, x_3) | B_1x_1 + B_2x_2 + B_3x_3 = d, x_1 \geq 0, x_2 \geq 0, x_3 \geq 0\}$.

In order to show the efficiency and effectiveness of our algorithm, we conduct numerical experiments on the performance of the proposed algorithm and the parallel splitting augmented Lagrangian method in [11] since both methods are used for a one-leader-three-follower game. The performance of the two algorithms with different initial points (u^0 = rand(61, 1), u^0 = 50* ones(61, 1), and u^0 = 100* ones(61, 1)) and optimality tolerance ($\epsilon = 10^{-4}$, $\epsilon = 10^{-5}$, and $\epsilon = 10^{-6}$) for the traffic equilibrium problem with fixed demand constraints (Example 5.2) is stated in Table 4. Here, Alg means algorithm and '-' means failure. In these tests, the common parameters of the two methods are the same, that is, $H = \beta I$, where $\beta = 0.8$ and I is the identity matrix, and the maximum number of iterations is 5,000.

The numerical results from Table 4 demonstrate the preference of our algorithm over the algorithm in [11] since both the number of iterations and the CPU time of our algorithm are smaller than those of the algorithm in [11] for a random initial point and our algorithm can solve the problem while the algorithm in [11] fails to solve it for other initial points. The results verify the efficiency and effectiveness of the proposed algorithm again.

6 Conclusion

The system (1.1)-(1.2) can be considered as a mathematical formulation of a one-leader-m-follower Stackelberg game in which the leader constantly improves his strategy by determining the value of λ from strategy set $\mathcal{R}^l$ while the ith follower determines his plan x_i from set $\mathcal{X}_i$ based on the value of λ. Based on the characteristic, we design an augmented Lagrangian-based parallel splitting method to solve the system. In the method, each player can only control and improve his own decision. We establish the global convergence of the method under some suitable conditions. Finally, we conduct a computational study to demonstrate the validity and efficiency of our algorithm.

To improve the application of the proposed algorithm, we provide two research directions according to its limitations. First, the convergence of the method is proved under the condition that each player's utility function is strongly monotone. We plan to relax the condition such that our method can be applied to more practical problems. Second, our method only serves to solve problems with a separable structure, which sounds reasonable but may not always be the case. We should improve it to solve general problems.

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

Authors' contributions
All authors contributed equally and significantly in this paper. All authors read and approved the final manuscript.

Acknowledgements
This work is supported by grants from the NSF of Shanxi Province (2014011006-1).

Received: 2 December 2015 Accepted: 19 March 2016 Published online: 01 April 2016

References

1. Eckstein, J, Bertsekas, DP: On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Program. **55**, 293-318 (1992)
2. Fukushima, M: Application of the alternating direction method of multipliers to separable convex programming problems. Comput. Optim. Appl. **1**, 93-111 (1992)
3. Chen, G, Teboulle, M: A proximal-based decomposition method for convex minimization problems. Math. Program. **64**, 81-101 (1994)
4. Tseng, P: Alternating projection-proximal methods for convex programming and variational inequalities. SIAM J. Optim. **7**, 951-965 (1997)
5. Han, D, He, H, Yang, H, Yuan, X: A customized Douglas-Rachford splitting algorithm for separable convex minimization with linear constraints. Numer. Math. **127**, 167-200 (2014)
6. Hestenes, MR: Multiplier and gradient methods. J. Optim. Theory Appl. **4**, 303-320 (1969)
7. Gabay, D, Mercier, B: A dual algorithm for the solution of nonlinear variational problems via finite element approximations. Comput. Math. Appl. **2**, 17-40 (1976)
8. Esser, E: Applications of Lagrangian-based alternating direction methods and connections to split Bregman. CAM Report 09-31, UCLA (2009)
9. Lin, Z, Chen, M, Ma, Y: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. UILU-ENG 09-2215, UIUC (2009)
10. Yang, JF, Zhang, Y: Alternating direction algorithms for ℓ_1-problems in compressive sensing. SIAM J. Sci. Comput. **33**, 250-278 (2011)
11. He, BS: Parallel splitting augmented Lagrangian methods for monotone structured variational inequalities. Comput. Math. Appl. **42**, 195-212 (2009)
12. Tao, M: Some parallel splitting methods for separable convex programming with $O(1/t)$ convergence rate. Pac. J. Optim. **10**, 359-384 (2014)
13. Wang, K, Xu, L, Han, D: A new parallel splitting descent method for structured variational inequalities. J. Ind. Manag. Optim. **10**, 461-476 (2014)
14. Jiang, ZK, Yuan, XM: New parallel descent-like method for solving a class of variational inequalities. J. Optim. Theory Appl. **145**, 311-323 (2010)
15. Han, D, Yuan, X, Zhang, W: An augmented Lagrangian based parallel splitting method for separable convex minimization with applications to image processing. Math. Comput. **83**, 2263-2291 (2014)
16. Facchinei, F, Fischer, A, Piccialli, V: Generalized Nash equilibrium problems and Newton methods. Math. Program. **117**, 163-194 (2009)
17. Facchinei, F, Kanzow, C: Penalty methods for the solution of generalized Nash equilibrium problems. SIAM J. Optim. **20**, 2228-2253 (2010)
18. Han, D, Zhang, H, Qian, G, Xu, L: An improved two-step method for solving generalized Nash equilibrium problems. Eur. J. Oper. Res. **216**, 613-623 (2012)
19. Nagurney, A, Zhang, D: Projected Dynamical Systems and Variational Inequalities with Applications. Kluwer Academic, Boston (1996)
20. He, BS, Xu, Y, Yuan, XM: A logarithmic-quadratic proximal prediction-correction method for structured monotone variational inequalities. Comput. Optim. Appl. **35**, 19-46 (2006)