A Primal-Dual Proximal Point Algorithm for Variational Inequality Problems

Kouichiro IWAOKA[1], Masao FUKUSHIMA[2] and Toshihide IBARAKI[1]

[1] Department of Applied Mathematics and Physics
Faculty of Engineering, Kyoto University
Kyoto 606, Japan

[2] Graduate School of Information Science
Nara Institute of Science and Technology
Ikoma, Nara 630-01, Japan
e-mail: fuku@is.aist-nara.ac.jp

February 1994

Abstract: The proximal point algorithm is an iterative method for finding a zero of a maximal monotone set-valued mapping, and has served as a fundamental method for solving variational inequality problems and, in particular, convex optimization problems. In this paper, we propose a new variant of the primal-dual proximal point algorithm for solving monotone variational inequality problems. The proposed method adopts a Gauss-Seidel-like procedure to solve a subproblem in each iteration, which updates primal variables and dual variables alternatively. We prove the convergence of the proposed method and show that our method is particularly efficient for variational inequality problems with certain separable structure, such as traffic assignment problems. Finally, we report some computational results for the proposed algorithm with the traffic assignment problem.

Key Words: Variational inequality problem, primal-dual proximal point algorithm, parallel computation, traffic assignment.

1. Introduction

Let F be a maximal monotone mapping from R^N into itself and X be a nonempty closed convex subset of R^N. The variational inequality problem can be stated as follows:

$$\text{[VI]} \qquad \text{Find } x^* \in X \text{ such that } \langle F(x^*),\, x - x^* \rangle \geq 0 \quad \text{for all } x \in X,$$

where $\langle \cdot\,, \cdot \rangle$ denotes the inner product in R^N. We suppose that X is defined by

$$X = \Big\{ x \in R^N \;\Big|\; g(x) \leq 0,\ h(x) = 0 \Big\} \cap S, \tag{1.1}$$

where $g(x) = [g_1(x), g_2(x), \cdots, g_M(x)]^T$ with $g_i : R^N \to R$ being continuously differentiable and convex, $h(x) = [h_1(x), h_2(x), \cdots, h_P(x)]^T$ with $h_j : R^N \to R$ being affine, and S denotes a closed polyhedral convex subset of R^N.

Over the years, various solution methods for problem VI have been developed [7]. In this paper, we propose a method for solving problem VI based on the proximal point algorithm [2, 14]. Our method turns out to be particularly efficient for problem VI with certain separable structure, such as traffic assignment problems.

The proximal point algorithm is a method for finding a zero of a maximal monotone set-valued mapping T :

$$\text{Find} \quad v^* \quad \text{such that} \quad 0 \in T(v^*).$$

This method is an iterative method such that in iteration k one solves the equation

$$0 \ \in \ T(v) \ + \frac{1}{c^{(k)}}(v - v^{(k)}),$$

where $v^{(k)}$ is the current iterate and $c^{(k)}$ is a positive parameter. The proximal point algorithm can be applied to problem VI in the following manner. Let $\lambda \in R^M$ and $\pi \in R^P$ be Lagrange multipliers vectors associated with the inequality and equality constraints in (1.1), respectively, and define the mapping T by

$$T(x, \lambda, \pi) = \Big\{ (\xi, \zeta, \eta) \in R^{N+M+P} \;\Big|\; \begin{array}{l} \xi \ \in F(x) + \nabla g(x)\lambda + \nabla h(x)\pi + N_S(x), \\ \zeta \ \in -g(x) + N_{R^M_+}(\lambda),\ \eta = -h(x) \ \Big\}, \end{array} \tag{1.2}$$

where R^M_+ denotes the nonnegative orthant of R^M, $N_S(x)$ is the normal cone to S at x, and $N_{R^M_+}(\lambda)$ is the normal cone to R^M_+ at λ. It is known [15, 16] that the mapping T is maximal monotone. Also, it is shown in [15, 16] that any zero of the mapping T is a solution of problem

VI if the Slater's constraint qualification is satisfied, i.e., there is an $x \in X$ satisfying $g(x) < 0$. Then, the proximal point algorithm for the mapping T is to solve at each iteration the following system of equations:

$$0 \in F(x) + \nabla g(x)\lambda + \nabla h(x)\pi + N_S(x) + \frac{1}{c^{(k)}}(x - x^{(k)}), \tag{1.3}$$

$$0 \in -g(x) + N_{R^M_+}(\lambda) + \frac{1}{c^{(k)}}(\lambda - \lambda^{(k)}), \tag{1.4}$$

$$0 = -h(x) + \frac{1}{c^{(k)}}(\pi - \pi^{(k)}). \tag{1.5}$$

Note that λ and π satisfying (1.4) and (1.5) are expressed as

$$\lambda = \Lambda(x, \lambda^{(k)}, c^{(k)}), \tag{1.6}$$

$$\pi = \Pi(x, \pi^{(k)}, c^{(k)}), \tag{1.7}$$

where $\Lambda(x, \lambda, c)$ and $\Pi(x, \pi, c)$ are the vectors which have the following functions as the components, respectively,

$$\Lambda_i(x, \lambda, c) = \max\{\lambda_i + c\, g_i(x),\, 0\}, \qquad i = 1, \cdots M, \tag{1.8}$$

$$\Pi_j(x, \pi, c) = \pi_j + c\, h_j(x), \qquad j = 1, \cdots, P. \tag{1.9}$$

To solve problem VI, Rockafellar [15, 16] has proposed an iterative procedure called the proximal method of multipliers. The proximal method of multipliers is a modification of the method of multipliers or the augmented Lagrangian method developed for nonlinear optimization [1, 8, 12, 13] and variational inequalities [5, 10]. This method belongs to a class of primal-dual proximal point algorithms, which have been applied to convex programming in various manners [9, 15, 16, 18]. In the proximal method of multipliers, One solves approximately (1.3), with λ and π expressed in terms of x from (1.6) and (1.7). In other words, one finds x satisfying the equation

$$0 \in F(x) + \nabla g(x)\, \Lambda(x, \lambda^{(k)}, c^{(k)}) + \nabla h(x)\, \Pi(x, \pi^{(k)}, c^{(k)}) + N_S(x) + \frac{1}{c^{(k)}}(x - x^{(k)}). \tag{1.10}$$

Unfortunately the equation (1.10) is not separable even though F, g, h and S are separable, since $\nabla g(x)\, \Lambda(x, \lambda^{(k)}, c^{(k)})$ and $\nabla h(x)\, \Pi(x, \pi^{(k)}, c^{(k)})$ usually involve cross terms. This difficulty is common to the method of multipliers in nonlinear optimization, for which a number of modifications have been proposed in order to restore the separability [4, 9, 11, 17]. In this paper, bearing in mind the applicability to separable problems, we propose an approach to solve subproblems (1.3)-(1.5) of the primal-dual proximal point algorithm in a way different from [15, 16].

This approach, roughly speaking, adopts a Gauss-Seidel-like procedure to solve approximately (1.3)-(1.5). In other words, one first updates x for fixed λ and π, then one improves λ and π by using the new data of x. This procedure thereby enables us to make the most of a separable structure the given problem may have.

This paper is organized as follows. In Section 2, we state a new variant of the primal-dual proximal point algorithm. In Section 3, we prove the convergence of the proposed method. In Section 4, we apply the method to traffic assignment problems. Section 5 concludes the paper.

2. Algorithm

In this section, we propose a variant of the primal-dual proximal point method for solving problem VI with the constraints defined by (1.1). The method is in essence the proximal point algorithm for the mapping T defined by (1.2). Let $\mathrm{SP}(x, \lambda, \pi; c)$ denote the subproblem to be solved for the current iteration point (x, λ, π) (see (1.3)-(1.5)).

$[\mathbf{SP}(x, \lambda, \pi; c)]$ Find $(\overline{x}, \overline{\lambda}, \overline{\pi})$ such that

$$
\begin{aligned}
0 &\in F(\overline{x}) + \nabla g(\overline{x})\,\overline{\lambda} + \nabla h(\overline{x})\,\overline{\pi} + N_S(\overline{x}) + \frac{1}{c}(\overline{x} - x), \\
0 &\in -g(\overline{x}) + N_{R^M_+}(\overline{\lambda}) + \frac{1}{c}(\overline{\lambda} - \lambda), \\
0 &= -h(\overline{x}) + \frac{1}{c}(\overline{\pi} - \pi).
\end{aligned}
$$

Note that the solution $(\overline{x}, \overline{\lambda}, \overline{\pi})$ of subproblem $\mathrm{SP}(x, \lambda, \pi; c)$ is uniquely determined. In fact, using the mapping T defined by (1.2), $(\overline{x}, \overline{\lambda}, \overline{\pi})$ can be expressed as $(\overline{x}, \overline{\lambda}, \overline{\pi}) = (I + cT)^{-1}(x, \lambda, \pi)$. We first state a procedure, called $\mathrm{SSP}(x, \lambda, \pi; c, \epsilon)$, to find an approximate solution $(\overline{z}, \overline{\mu}, \overline{\nu})$ of subproblem $\mathrm{SP}(x, \lambda, \pi; c)$.

Procedure **SSP**$(x, \lambda, \pi; c, \epsilon)$**.**
Input: a current iteration point (x, λ, π) and positive scalars c and ϵ.
Output: an approximate solution $(\hat{z}, \hat{\mu}, \hat{\nu})$ of subproblem $\mathrm{SP}(x, \lambda, \pi; c)$.
Step 1: $z^{(0)} := x$; $\mu^{(0)} := \Lambda(x, \lambda, c)$; $\nu^{(0)} := \Pi(x, \pi, c)$;
$l := 0$.
Step 2: Find $z^{(l+1)}$ such that

$$
0 \in F(z^{(l+1)}) + \nabla g(z^{(l+1)})\,\mu^{(l)} + \nabla h(z^{(l+1)})\,\nu^{(l)} + N_S(z^{(l+1)}) + \frac{1}{c}(z^{(l+1)} - x).
$$

Step 3: If the stopping criterion

$$\| z^{(l+1)} - z^{(l)} \| \leq \epsilon$$

is satisfied, then go to Step 5.

Step 4: $\mu^{(l+1)} := \Lambda(z^{(l+1)}, \lambda, c)$; $\nu^{(l+1)} := \Pi(z^{(l+1)}, \pi, c)$;

$l := l + 1$;

go to Step 2.

Step 5: $\hat{z} := z^{(l+1)}$;

$\hat{\mu} := \mu^{(l)}$; $\hat{\nu} := \nu^{(l)}$;

exit.

Note that the functions Λ and Π that appear in Step 4 are defined by (1.8) and (1.9) in the previous section. Using this procedure, a primal-dual proximal point algorithm for problem VI can be stated as follows. This algorithm uses two sequences of positive parameters $\{c^{(k)}\}$ and $\{\epsilon^{(k)}\}$ such that

$$\inf_k c^{(k)} > 0, \quad \sum_{k=0}^{\infty} \epsilon^{(k)} < \infty .$$

Algorithm PDPPA.

Step 1: Choose a starting point $(x^{(0)}, \lambda^{(0)}, \pi^{(0)})$;

$k := 0$.

Step 2: Obtain an approximate solution $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)})$ of subproblem

$\mathrm{SP}(x^{(k)}, \lambda^{(k)}, \pi^{(k)}; c^{(k)})$ by executing procedure $\mathrm{SSP}(x^{(k)}, \lambda^{(k)}, \pi^{(k)}; c^{(k)}, \epsilon^{(k)})$.

Step 3: If $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)}) = (x^{(k)}, \lambda^{(k)}, \pi^{(k)})$, then stop. Else

$k := k + 1$;

go to Step 2.

The termination condition $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)}) = (x^{(k)}, \lambda^{(k)}, \pi^{(k)})$ in Step 3 guarantees that $x^{(k+1)}$ is a solution of problem VI. This fact will be proved in the next section.

Now let us consider a variational inequality problem with certain separable structure. Specifically, suppose that

$$S = S_0 \times \cdots \times S_q \times \cdots \times S_Q,$$

where S_q is a closed polyhedral subset of R^{N_q}, $\sum_{q=0}^{Q} N_q = N$ and $\times$ denotes the Cartesian

product. Moreover, suppose that

$$x = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_Q \end{bmatrix}, \quad F(x) = \begin{bmatrix} F_0(x_0) \\ F_1(x_1) \\ \vdots \\ F_Q(x_Q) \end{bmatrix}, \quad g(x) = \sum_{q=0}^{Q} g_q(x_q), \quad h(x) = \sum_{q=0}^{Q} A_q x_q - b,$$

where $F_q \colon R^{N_q} \to R^{N_q}$ are maximal monotone mappings, $g_q \colon R^{N_q} \to R^M$ are mappings whose components are continuously differentiable convex functions, and A_q and b are $P \times N_q$ matrices and an P-vector, respectively. Problem VI with the above structure will be called a separable variational inequality problem (SVI). For problem SVI, the equation to be solved in Step 2 of procedure SSP$(x, \lambda, \pi;\ c,\ \epsilon)$ can be decomposed into Q smaller equations as follows:

For each $q = 0, \cdots, Q$, find $z_q^{(l+1)}$ such that

$$0 \in F_q(z_q^{(l+1)}) \ + \nabla g_q(z_q^{(l+1)})\, \mu^{(l)} \ + A_q^T\, \nu^{(l)} + N_{S_q}(z_q^{(l+1)}) \ + \frac{1}{c}(z_q^{(l+1)} - x_q)\ .$$

Note that these Q equations can be solved independently of each other. Therefore, Algorithm PDPPA can take full advantage of the separable structure of problem SVI.

3. Convergence

In this section, we prove the convergence of the algorithm PDPPA proposed in the previous section. First we establish a finite termination property of procedure SSP$(x, \lambda, \pi; c, \epsilon)$.

Proposition 1. *Let h be represented as $h(x) = Ax - b$. Suppose that g is Lipschitz continuous with constant $L_g > 0$, that is, $\parallel g(x) - g(x') \parallel \leq L_g \parallel x - x' \parallel$ for all x and x'. If c is chosen smaller than $(L_g^2 + \rho(A^T A))^{-\frac{1}{2}}$, then procedure* SSP$(x, \lambda, \pi; c, \epsilon)$ *halts finitely, where $\rho(A^T A)$ denotes the spectral radius of $A^T A$.*

Proof. Let $\{z^{(l)}\}, \{\mu^{(l)}\}$ and $\{\nu^{(l)}\}$ be the sequences generated by procedure SSP$(x, \lambda, \pi;\ c, \epsilon)$. Since $z^{(l+1)}$ is the unique solution of the equation in Step 2 and since $\mu^{(l)}$ and $\nu^{(l)}$ equal $\Lambda(z^{(l)}, \lambda, c)$ and $\Pi(z^{(l)}, \pi, c)$ as determined in Step 4 on the previous iteration, we have for all $z \in S$

$$\langle F(z^{(l+1)}) + \nabla g(z^{(l+1)})\, \Lambda(z^{(l)}, \lambda, c) \ + \nabla h(z^{(l+1)})\, \Pi(z^{(l)}, \pi, c) \ + \frac{1}{c}(z^{(l+1)} - x),\ z - z^{(l+1)} \rangle \ \geq \ 0.$$

Let $(\overline{x}, \overline{\lambda}, \overline{\pi})$ denote the unique exact solution of subproblem SP(x,λ, π; c). Then, since $\overline{\lambda} = \Lambda(\overline{x}, \lambda, c)$ and $\overline{\pi} = \Pi(\overline{x}, \pi, c)$, we have for all $z \in S$

$$\langle F(\overline{x}) \ + \nabla g(\overline{x}) \ \Lambda(\overline{x}, \lambda, c) \ + \nabla h(\overline{x}) \ \Pi(\overline{x}, \pi, c) \ + \frac{1}{c}(\overline{x} - x), \ z \ - \overline{x} \rangle \geq 0.$$

Substituting $z = \overline{x}$ in the first inequality and $z = z^{(l+1)}$ in the second inequality, and then adding the two inequalities, we have

$$\begin{aligned} \frac{1}{c} \parallel \ z^{(l+1)} - \ \overline{x} \ \parallel^2 \ \leq \ & -\langle \ F(z^{(l+1)}) - F(\overline{x}), \ z^{(l+1)} - \overline{x} \ \rangle \\ & -\langle \ \nabla g(z^{(l+1)}) \ \Lambda(z^{(l)}, \lambda, c) - \nabla g(\overline{x}) \ \Lambda(\overline{x}, \lambda, c), \ z^{(l+1)} - \overline{x} \ \rangle \\ & -\langle \ \nabla h(z^{(l+1)}) \ \Pi(z^{(l)}, \pi, c) - \nabla h(\overline{x}) \ \Pi(\overline{x}, \pi, c), \ z^{(l+1)} - \overline{x} \ \rangle. \end{aligned} \quad (3.1)$$

By the monotonicity of F, we have

$$\langle \ F(z^{(l+1)}) - F(\overline{x}), z^{(l+1)} - \overline{x} \ \rangle \geq 0. \quad (3.2)$$

By the differentiability and convexity of g and the nonnegativity of Λ, we have

$$\langle \ \Lambda(z^{(l)}, \lambda, c), g(\overline{x}) - g(z^{(l+1)}) \ \rangle \geq \langle \ \Lambda(z^{(l)}, \lambda, c), \nabla g(z^{(l+1)})^T (\overline{x} - z^{(l+1)}) \ \rangle \quad (3.3)$$

and

$$\langle \ \Lambda(\overline{x}, \lambda, c), g(z^{(l+1)}) - g(\overline{x}) \ \rangle \geq \langle \ \Lambda(\overline{x}, \lambda, c), \nabla g(\overline{x})^T (z^{(l+1)} - \overline{x}) \ \rangle. \quad (3.4)$$

Since $h(x) = Ax - b$, we have

$$\nabla h(x) = A^T. \quad (3.5)$$

It follows from (3.1)-(3.5) that

$$\begin{aligned} \parallel z^{(l+1)} - \overline{x} \parallel^2 \ \leq \ & c \ \langle \ \Lambda(\overline{x}, \lambda, c) - \ \Lambda(z^{(l)}, \ \lambda, c), \ g(z^{(l+1)}) - g(\overline{x}) \ \rangle \\ & + c \langle \ \Pi(\overline{x}, \pi, c) - \Pi(z^{(l)}, \pi, c), \ A \ (z^{(l+1)} - \overline{x}) \ \rangle. \end{aligned}$$

From the Cauchy-Schwarz inequality and the definitions of Π and h, we obtain

$$\begin{aligned} \parallel z^{(l+1)} - \overline{x} \parallel^2 \ \leq \ & c \ \parallel \Lambda(\overline{x}, \lambda, c) - \Lambda(z^{(l)}, \lambda, c) \parallel \cdot \parallel g(z^{(l+1)}) - g(\overline{x}) \parallel \\ & + c^2 \, \rho(A^T A) \parallel \overline{x} - z^{(l)} \parallel \cdot \parallel z^{(l+1)} - \overline{x} \parallel . \end{aligned} \quad (3.6)$$

On the other hand, it can be shown from the definition of Λ that

$$\parallel \ \Lambda(\overline{x}, \lambda, c) \ - \Lambda(z^{(l)}, \lambda, c) \ \parallel \leq \ c \ \parallel \ g(\overline{x}) - g(z^{(l)}) \ \parallel . \quad (3.7)$$

To see (3.7), let

$$\begin{aligned}\Delta\Lambda_i &= \Lambda_i(\overline{x},\lambda,c) - \Lambda_i(z^{(l)},\lambda,c)\\ &= \max\{\lambda_i + cg_i(\overline{x}),0\} - \max\{\lambda_i + cg_i(z^{(l)}),0\},\end{aligned}$$

and consider the following four cases:

(i) if $\lambda_i + cg_i(\overline{x}) \geq 0, \lambda_i + cg_i(z^{(l)}) \geq 0$, then $\Delta\Lambda_i = c(g_i(\overline{x}) - g_i(z^{(l)}))$;

(ii) if $\lambda_i + cg_i(\overline{x}) \geq 0, \lambda_i + cg_i(z^{(l)}) < 0$, then $0 \leq \Delta\Lambda_i \leq c(g_i(\overline{x}) - g_i(z^{(l)}))$;

(iii) if $\lambda_i + cg_i(\overline{x}) < 0, \lambda_i + cg_i(z^{(l)}) \geq 0$, then $0 \geq \Delta\Lambda_i > c(g_i(\overline{x}) - g_i(z^{(l)}))$;

(iv) if $\lambda_i + cg_i(\overline{x}) < 0, \lambda_i + cg_i(z^{(l)}) < 0$, then $\Delta\Lambda_i = 0$.

Thus, in each case, we have $|\Delta\Lambda_i| \leq c|g_i(\overline{x}) - g_i(z^{(l)})|$, which implies (3.7). By (3.6), (3.7) and the Lipschitz continuity of g, we have

$$\| z^{(l+1)} - \overline{x} \| \leq c^2(L_g^2 + \rho(A^TA)) \| z^{(l)} - \overline{x} \| . \tag{3.8}$$

Since $c^2(L_g^2 + \rho(A^TA)) < 1$ by assumption, the inequality (3.8) implies that $\{z^{(l)}\}$ converges to $\overline{x}$ linearly. Therefore, the stopping criterion in Step 3 of procedure SSP$(x,\lambda,\pi;c,\epsilon)$ must be satisfied in a finite number of iterations. □

Remark. The inequality (3.8) indicates that the sequence $\{z^{(l)}\}$ generated by procedure SSP$(x,\lambda,\pi;c,\epsilon)$ converges to an exact solution $\overline{x}$ at a linear rate.

The theorem below justifies the termination condition at Step 3 of algorithm PDPPA.

Theorem 1. *Let the assumptions in Propositions* 1 *be satisfied. Suppose that* $c^{(k)}$ *are chosen smaller than* $(L_g^2 + \rho(A^TA))^{-\frac{1}{2}}$ *for all* k. *If algorithm* PDPPA *terminates at Step* 3, *then the last iterate* $x^{(k+1)}$ *is a solution of problem* VI.

Proof. By the termination condition of algorithm PDPPA, we have

$$x^{(k+1)} = x^{(k)}. \tag{3.9}$$

Suppose that procedure SSP$(x^{(k)},\lambda^{(k)},\pi^{(k)};c^{(k)},\epsilon^{(k)})$ halts at iteration $\hat{l}$. That is to say, we have

$$x^{(k+1)} = \hat{z} = z^{(\hat{l}+1)}. \tag{3.10}$$

Also we have

$$z^{(0)} = x^{(k)} \tag{3.11}$$

from Step 1 of procedure SSP$(x^{(k)}, \lambda^{(k)}, \pi^{(k)}; c^{(k)}, \epsilon^{(k)})$. It then follows from (3.9)-(3.11) that

$$z^{(0)} = z^{(\hat{l}+1)}. \tag{3.12}$$

Now let $(\overline{x}, \overline{\lambda}, \overline{\pi})$ denote the exact solution of subproblem SP$(x^{(k)}, \lambda^{(k)}, \pi^{(k)}; c^{(k)})$. By (3.8), if $z^{(0)} \neq \overline{x}$, then

$$\| z^{(\hat{l}+1)} - \overline{x} \| < \| z^{(\hat{l})} - \overline{x} \| < \cdots < \| z^{(0)} - \overline{x} \| .$$

Therefore, in order for (3.12) to hold, we must have

$$\overline{x} = z^{(0)}, \tag{3.13}$$

which together with (3.9) and (3.11) implies that

$$x^{(k+1)} = x^{(k)} = \overline{x}. \tag{3.14}$$

Moreover, by (3.8), (3.13) implies $\overline{x} = z^{(1)}$. In other words, procedure SSP$(x^{(k)}, \lambda^{(k)}, \pi^{(k)}; c^{(k)}, \epsilon^{(k)})$ necessarily terminates at the first iteration, i.e., $\hat{l} = 0$. Since $\lambda^{(k+1)} = \hat{\mu}$ and $\hat{\mu} = \mu^{(\hat{l})} = \mu^{(0)}$, it follows from (3.14) that

$$\lambda^{(k+1)} = \mu^{(0)} = \Lambda(x^{(k)}, \lambda^{(k)}, c^{(k)}) = \Lambda(x^{(k+1)}, \lambda^{(k)}, c^{(k)}). \tag{3.15}$$

This implies that $\lambda^{(k+1)}$ satisfies (1.4) with $x = x^{(k+1)}$. In a similar way, we obtain

$$\pi^{(k+1)} = \nu^{(0)} = \Pi(x^{(k)}, \pi^{(k)}, c^{(k)}) = \Pi(x^{(k+1)}, \pi^{(k)}, c^{(k)}), \tag{3.16}$$

which implies that $\pi^{(k+1)}$ satisfies (1.5) with $x = x^{(k+1)}$. Moreover, from Step 2 of procedure SSP$(x^{(k)}, \lambda^{(k)}, \pi^{(k)}; c^{(k)}, \epsilon^{(k)})$, we have

$$0 \in F(z^{(1)}) + \nabla g(z^{(1)})\mu^{(0)} + \nabla h(z^{(1)})\nu^{(0)} + N_S(z^{(1)}) + \frac{1}{c^{(k)}}(z^{(1)} - x^{(k)}). \tag{3.17}$$

Since $\hat{l} = 0$ implies $z^{(1)} = x^{(k+1)}, \mu^{(0)} = \lambda^{(k+1)}$ and $\nu^{(0)} = \pi^{(k+1)}$, it follows from (3.17) that $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)})$ satisfies (1.3). Consequently, if $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)}) = (x^{(k)}, \lambda^{(k)}, \pi^{(k)})$ holds, it follows from (1.3)-(1.5) that $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)})$ is a zero of the mapping T, i.e., $x^{(k+1)}$ is a solution of problem VI. □

Now we show that algorithm PDPPA converges to a solution of problem VI, when it generates an infinite sequence.

Theorem 2. *Let the assumptions in Proposition* 1 *be satisfied. Suppose that* $c^{(k)}$ *are chosen smaller than* $(L_g^2 + \rho(A^T A))^{-\frac{1}{2}}$ *for all* k. *If problem* VI *has at least one solution, then the sequence* $\{(x^{(k)}, \lambda^{(k)}, \pi^{(k)})\}$ *generated by algorithm* PDPPA *with an arbitrary starting point* $(x^{(0)}, \lambda^{(0)}, \pi^{(0)})$ *converges to a particular solution* (x^*, λ^*, π^*) *of problem* VI.

Proof. Consider an arbitrary iteration k of algorithm PDPPA. Suppose that procedure SSP($x^{(k)}$, $\lambda^{(k)}, \pi^{(k)}; c^{(k)}, \epsilon^{(k)}$) halts at iteration $\hat{l}$. Namely, $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)})$ is determined as $(z^{(\hat{l}+1)}$, $\mu^{(\hat{l})}, \nu^{(\hat{l})})$ satisfying the stopping criterion

$$\| z^{(\hat{l}+1)} - z^{(\hat{l})} \| \leq \epsilon^{(k)}. \tag{3.18}$$

For simplicity of notation, we shall write $(x^{(k)}, \lambda^{(k)}, \pi^{(k)})$, $(x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)})$, $c^{(k)}$ and $\epsilon^{(k)}$ as (x, λ, π), (x^+, λ^+, π^+), c and ϵ, respectively. Moreover, let z^+, z, μ and ν stand for $z^{(\hat{l}+1)}$, $z^{(\hat{l})}$, $\mu^{(\hat{l})}$ and $\nu^{(\hat{l})}$, respectively. Then, by the definition (1.2) of mapping T, we have

$$\begin{aligned}
(I + cT)\,(x^+, \lambda^+, \pi^+) &= (I + cT)\,(z^+, \mu, \nu) \\
&= (\, z^+ + cF(z^+) + c\nabla g(z^+)\,\mu + c\nabla h(z^+)\,\nu + N_S(z^+), \\
&\qquad \mu - cg(z^+) + N_{R_+^M}(\mu),\ \nu - ch(z^+)\,).
\end{aligned} \tag{3.19}$$

From Step 2 and Step 4 of procedure SSP($x, \lambda, \pi;\ c,\ \epsilon$), we obtain

$$x \in z^+ + cF(z^+) + c\nabla g(z^+)\,\mu\ + c\nabla h(z^+)\,\nu\ + N_S(z^+), \tag{3.20}$$

$$\lambda \in \mu - cg(z) + N_{R_+^M}(\mu), \tag{3.21}$$

$$\pi = \nu - ch(z). \tag{3.22}$$

It then follows from (3.19)-(3.22) that

$$(\, x,\ \lambda - cg(z^+) + cg(z),\ \pi - ch(z^+) + ch(z)\,) \in (I + cT)\,(x^+, \lambda^+, \pi^+)\,,$$

namely,

$$(x^+, \lambda^+, \pi^+) = J_T^c\,(\, x, \lambda - cg(z^+) + cg(z)\,,\ \pi - ch(z^+) + ch(z)\,)\,, \tag{3.23}$$

where J_T^c denotes the resolvent of T, i.e., $J_T^c = (I + cT)^{-1}$. On the other hand, we have

$$\begin{aligned}
&\| (x^+, \lambda^+, \pi^+) - J_T^c\,(x, \lambda, \pi) \| \\
&\quad = \| J_T^c\,(\, x, \lambda - cg(z^+) + cg(z), \pi - ch(z^+) + ch(z)\,) - J_T^c\,(x, \lambda, \pi) \| \\
&\quad \leq c \,\| \,(\,0,\ g(z) - g(z^+),\ h(z) - h(z^+)\,) \,\| \\
&\quad = c\,(\,\| g(z) - g(z^+) \|^2 + \| h(z) - h(z^+) \|^2\,)^{\frac{1}{2}} \\
&\quad \leq c\,(\,L_g^2 + \rho(A^T A))^{\frac{1}{2}}\ \| z^+ - z \|,
\end{aligned} \tag{3.24}$$

where the first inequality follows from the nonexpansiveness of J_T^c and the last inequality follows from the Lipschitz continuity of g and the linearity of h. From (3.18) and (3.24), we have

$$\| \ (x^+, \lambda^+, \pi^+) - \ J_T^c \ (x, \lambda, \pi) \ \| \leq \ c(L_g^2 + \rho(A^T A))^{\frac{1}{2}} \epsilon,$$

i.e.,

$$\begin{aligned} \| \ (x^{(k+1)}, \lambda^{(k+1)}, \pi^{(k+1)}) - J_T^{c^{(k)}}(x^{(k)}, \lambda^{(k)}, \pi^{(k)}) \ \| \quad &\leq \quad c^{(k)}(L_g^2 + \rho(A^T A))^{\frac{1}{2}} \epsilon^{(k)} \\ &\leq \quad \epsilon^{(k)}, \end{aligned}$$

where the last inequality follows from the assumption on $c^{(k)}$. Then, by Theorem 1 in Rockafellar [14], the sequence $\{(x^{(k)}, \lambda^{(k)}, \pi^{(k)})\}$ converges to a particular solution of problem VI. □

4. An Application to Traffic Assignment Problems

As mentioned in Section 2, algorithm PDPPA will particularly be effective for problem VI with separable structure such as problem SVI. In this section, we consider an application of algorithm PDPPA to traffic assignment problems [3]. First we introduce the notation for the problem.

- $\mathcal{G} = (\mathcal{N}, \mathcal{A})$: transportation network
- $q = 1, \cdots, Q$: commodities (OD pairs)
- $x_q \in R^{|\mathcal{A}|}$: arc flow vector for commodity q
- $x_0 \in R^{|\mathcal{A}|}$: vector of total flow $x_0 = \sum_{q=1}^{Q} x_q$
- E: node-arc incidence matrix of network $\mathcal{G}$
- $d_q \in R^{|\mathcal{N}|}$: travel demand vector for commodity q
- $S_q \subset R^{|\mathcal{A}|}$: set of feasible flows for commodity q,

$$S_q = \left\{ \ x_q \in R^{|\mathcal{N}|} \ \middle| \ Ex_q = d_q, x_q \geq 0 \right\}$$

- $X_0 \subset R^{|\mathcal{A}|}$: set of feasible total flows,

$$X_0 = \left\{ \ x_0 \in R^{|\mathcal{A}|} \ \middle| \ x_0 = \sum_{q=1}^{Q} x_q, \ x_q \in S_q, \ \ q = 1, \cdots, Q \right\}$$

- C: travel cost function from $R^{|\mathcal{A}|}$ into itself

We assume that the travel cost function C is maximal monotone. Then we can formulate the traffic assignment problem as the variational inequality problem:

[TA] Find $x_0^* \in X_0$ such that $\langle C(x_0^*), x_0 - x_0^* \rangle \geq 0$ for all $x_0 \in X_0$.

A solution x_0^* of problem TA is called a user-optimal flow in the sense that no user can decrease his/her travel cost by only changing his/her route. Clearly, problem TA can be reduced to problem SVI with the following identifications:

$$S = R^{|\mathcal{A}|} \times S_1 \times \cdots \times S_Q,$$

$$x = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_Q \end{bmatrix} \in R^{(Q+1)|\mathcal{A}|}, \quad F(x) = \begin{bmatrix} C(x_0) \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad h(x) = A\,x,$$

where A is a $|\mathcal{A}| \times (Q+1)|\mathcal{A}|$ matrix defined by

$$A = \begin{bmatrix} I & -I & \cdots & -I \end{bmatrix}$$

and I denotes the $|\mathcal{A}| \times |\mathcal{A}|$ identity matrix. Note that $g(x)$ and λ need not be considered since there are no nonlinear inequality constraints in problem TA. Then we may apply algorithm PDPPA to problem TA. Specifically, procedure SSP$(x, \pi; c, \epsilon)$ specialized to problem TA, which we call SSPTA$(x, \pi; c, \epsilon)$, is stated as follows.

Procedure **SSPTA**$(x, \pi; c, \epsilon)$

Input: a current iterate (x, π) and positive scalars c and ϵ.

Output: an approximate solution $(\hat{z}, \hat{\nu})$ of subproblem SP$(x, \pi; c)$

Step 1: $z_q^{(0)} := x_q$, $q = 0, 1, \cdots, Q$;

$\nu^{(0)} := \pi + c(x_0 - \sum_{q=1}^{Q} x_q)$;

$l := 0$.

Step 2: Find $z_0^{(l+1)}$ such that

$$C(z_0^{(l+1)}) + \nu^{(l)} + \frac{1}{c}(z_0^{(l+1)} - x_0) = 0.$$

Step 3: For each commodity $q = 1, \cdots, Q$, find $z_q^{(l+1)}$ such that

$$0 \in \frac{1}{c}(z_q^{(l+1)} - x_q) - \nu^{(l)} + N_{S_q}(z_q^{(l+1)}).$$

Step 4: If the stopping criterion

$$\sum_{q=0}^{Q} \parallel z_q^{(l+1)} - z_q^{(l)} \parallel \leq \epsilon$$

is satisfied, then go to Step 6.

Step 5: $\nu^{(l+1)} := \pi + c\left(z_0^{(l+1)} - \sum_{q=1}^{Q} z_q^{(l+1)}\right)$;

$l := l + 1$;

go to Step 2.

Step 6: $\hat{z}_q := z_q^{(l+1)}$, $q = 0, 1, \cdots, Q$;

$\hat{\nu} := \nu^{(l)}$;

exit.

Step 2 is solving an ordinary system of strongly monotone nonlinear equations with respect to total flows. This problem may be solved by using any standard method for nonlinear equations such as Newton or quasi-Newton methods. Step 3 is solving symmetric linear variational inequality problems associated with each commodity, which may alternatively be written as

$$\min_{z_q} \quad \frac{1}{2c} \parallel z_q - x_q \parallel^2 - \langle z_q\,,\, \nu^{(l)} \rangle \qquad \text{s.t.} \quad z_q \in S_q.$$

These are single commodity minimum cost flow problems with separable quadratic costs. Since these Q problems are independent of each other, they can be dealt with in parallel. Note that in order to guarantee the convergence of SSPTA$(x, \pi; c, \epsilon)$, the parameter c has to satisfy $c < \rho(A^T A)^{\frac{1}{2}}$ (see Proposition 1). Since $\rho(A^T A) = \rho(AA^T)$ by [6, p.285] and since $AA^T = (Q+1)I$, we may conclude that procedure SSPTA$(x, \pi; c, \epsilon)$ finitely terminates, provided that c is chosen such that $c < 1/\sqrt{Q+1}$.

5. Conclusion

We have proposed a new method for solving monotone variational inequality problems. The proposed method is a variant of the primal-dual proximal point algorithm, which uses the Gauss-Seidel-like procedure SSP to solve subproblem SP at each major iteration. In particular, when applied to the separable variational inequality problem SVI, procedure SSP efficiently takes advantage of the separable structure of the problem and hence is suited for parallel implementation. As an example, we have shown that the proposed method yields a decomposition algorithm for solving the traffic assignment problem.

References

[1] D.P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[2] D.P. Bertsekas and J.N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice Hall, New Jersey, 1989.

[3] S. Dafermos, "Traffic equilibrium and variational inequalities," *Transportation Science* 14 (1980), pp. 42-54.

[4] M. Fukushima, "Application of the alternating direction method of multipliers to separable convex programming problems," *Computational Optimization and Applications* 1 (1992), pp. 93-111.

[5] D. Gabay, "Application of the method of multipliers to variational inequalities, " in: M. Fortin and R. Gorwinski eds., *Augmented Lagrangian Methods: Applications to the Numerical Solutions of Boundary-Value Problems*, North-Holland, Amsterdam, 1983, pp. 299-331.

[6] G.H. Golub and C.F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, 1983.

[7] P.T. Harker and J.S. Pang, "Finite–dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications," *Mathematical Programming* 48 (1990), pp. 161-220.

[8] M. Hestenes, "Multiplier and gradient methods," *Journal of Optimization Theory and Applications* 4 (1969), pp. 303-320.

[9] S. Ibaraki, M. Fukushima and T. Ibaraki, "Primal-dual proximal point algorithm for linearly constrained convex programming problems," *Computational Optimization and Applications* 1 (1992), pp. 207-226.

[10] T. Ito, M. Fukushima and T. Ibaraki, "An iterative method for variational inequalities with application to traffic equilibrium problems," *Journal of the Operations Research Society of Japan* 31 (1988), pp. 82-104.

[11] J.M. Mulvey and A. Ruszczyński, "A diagonal quadratic approximation method for large scale linear programs," *Operations Research Letters* 12 (1992), pp. 205-215.

[12] M.J.D. Powell, "A method for nonlinear constraints in minimization problems," in: R. Fletcher eds., *Optimization*, Academic Press, London, 1969, pp. 283-298.

[13] R.T. Rockafellar, "Augmented Lagrange multiplier functions and duality in non-convex programming," *SIAM Journal on Control* 12 (1974), pp. 268-285.

[14] R.T. Rockafellar, "Monotone operators and the proximal point algorithm," *SIAM Journal on Control and Optimization* 14 (1976), pp. 877-898.

[15] R.T. Rockafellar, "Monotone operators and augmented Lagrangian methods in nonlinear programming," in: O.L. Mangasarian, R.R. Meyer and S.M. Robinson, eds., *Nonlinear Programming 3*, Academic Press, New York, 1978, pp. 1-25.

[16] R.T. Rockafellar, "Lagrange multipliers and variational inequalities," in: R.W. Cottle, F. Giannessi, and J.L. Lions eds., *Variational Inequality and Complementarity Problems: Theory and Applications* , Wiley, New York, 1980, pp. 330-322.

[17] P. Tseng, "Application of a splitting algorithm to decomposition in convex programming and variational inequalities," *SIAM Journal on Control and Optimization* 29 (1991), pp. 119-138.

[18] C. Zhu, "Modified proximal point algorithm for extended linear-quadratic programming," *Computational Optimization and Applications* 1 (1992), pp. 185-205.