A NEW INTERPRETATION OF THE PROGRESSIVE HEDGING ALGORITHM FOR MULTISTAGE STOCHASTIC MINIMIZATION PROBLEMS

JIE SUN
School of Electric Engineering, Computing, and Mathematical Science
Curtin University, Australia, and
School of Business, National University of Singapore

HONGLEI XU AND MIN ZHANG*
School of Electric Engineering, Computing, and Mathematical Science
Curtin University, Australia

(Communicated by Wenxun Xing)

Abstract. The progressive hedging algorithm of Rockafellar and Wets for multistage stochastic programming problems could be viewed as a two-block alternating direction method of multipliers. This correspondence brings in some useful results. In particular, it provides a new proof for the convergence of the progressive hedging algorithm with a flexibility in the selection of primal and dual step lengths and it helps to develop a new progressive hedging algorithm for solving risk averse stochastic optimization problems with cross constraints.

1. Introduction. Recently Rockafellar and Wets [10] developed a theoretical framework for multi-stage stochastic variational inequality (SVI), which includes the multistage stochastic optimization (MSO) program as a special case. Their progressive hedging algorithm (PHA) [9] developed in 1991 serves as a major computational tool in this framework. In the paper of Rockafellar and Sun [8] PHA has been extended to solve the monotone SVI, where the proof of convergence depends on interpreting the PHA as a proximal point method of Moreau and Rockafellar [4, 5] applied to a partial inverse of a certain monotone mapping in the sense of Spingarn [11]. This paper provides an alternative proof for the convergence of PHA by interpreting the PHA, when applied to MSO problems, as a two-block alternating direction method of multipliers (ADMM). This interpretation provides a flexibility in choosing step lengths – particularly in the dual step – for PHA. In addition, the correspondence between PHA and ADMM could widen the applicability of PHA. For instance, if the stochastic optimization problem includes certain constraints that are not decomposable with respect to the scenarios, then the original PHA can be modified by considering an ADMM, as we shall do in Section 3 of this paper.

This paper is organized as follows. In Section 2, we establish the correspondence between PHA and a two-block ADMM after a general introduction to PHA and...
its background. In Section 3, we consider an application of PHA in risk-averse stochastic optimization and reduce this problem to a multi-stage stochastic program with cross constraints. We point out that this new problem could be solved by a new progressive hedging scheme. We achieve this goal by invoking results in ADMM to show the convergence of the new PHA. The paper is concluded in Section 4 with a few remarks.

2. PHA for multi-stage stochastic optimization. Consider a finite set $\Xi$ of scenarios $\xi = (\xi_1, \ldots, \xi_K) \in \mathbb{R}^{m_1} \times \cdots \times \mathbb{R}^{m_K} =: \mathbb{R}^m$, composed of elements $\xi_k$ that are regarded as being revealed sequentially in $K$ stages. Each scenario $\xi$ has a known probability $p(\xi) > 0$, and these probabilities add to one. In this way $\Xi$ is a probability space. Our attention is directed to mappings that designate responses to the scenarios in $\Xi$ in the notation

$$x(\cdot): \xi \mapsto x(\xi) = (x_1(\xi), \ldots, x_K(\xi)) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_K} = \mathbb{R}^n.$$ 

The linear space $\ell^2$ consisting of all such functions $x(\cdot)$ from $\Xi$ to $\mathbb{R}^n$ is given the expectation inner product

$$\langle x(\cdot), w(\cdot) \rangle = \mathbb{E}_\xi[x(\xi)^T w(\xi)] = \sum_{\xi \in \Xi} p(\xi) \sum_{k=1}^K x_k(\xi)^T w_k(\xi),$$

which makes it into a finite-dimensional Hilbert space, where “$T$” stands for the transpose. Note that by $x(\cdot)$ we mean a function (mapping) of $\Xi \rightarrow \mathbb{R}^n$, while we use $x(\xi)$ to represent the image of $x(\cdot)$ for realization $\xi$. Our real interest centers on mappings $x(\cdot)$ that are nonanticipative in the sense that the response $x_k(\xi)$ at stage $k$ depends only on the portion $(\xi_1, \ldots, \xi_{k-1})$ of the scenario $\xi$ realized in earlier stages. That is, under nonanticipativity,

$$x(\xi) = (x_1, x_2(x_1), x_3(x_1, x_2), \ldots, x_K(x_1, x_2, \ldots, x_{K-1})).$$

We capture this condition as a linear constraint by requiring $x(\cdot)$ to belong to the nonanticipativity subspace $\mathcal{N}$ of $\ell^2$,

$$\mathcal{N} = \{x(\cdot) \in \ell^2 : x_k(\xi) \text{ doesn’t depend on } \xi_k, \ldots, \xi_K\}.$$

In our modeling framework $x(\cdot)$ arises as a feasible solution to the following MSO problem of the form

$$\min_{x(\cdot)} \mathbb{E}_\xi[g(x(\xi), \xi)] \quad \text{s.t.} \quad x(\cdot) \in \mathcal{N}, \ x(\xi) \in C_\xi \ \forall \xi \in \Xi, \quad (1)$$

where $g(x(\xi), \xi)$ is a finite-valued, convex and smooth function in $x(\xi)$ for all $\xi \in \Xi$ and $C_\xi$ is the feasible set for $x(\xi)$. In order not to mix with the real feasible set of (1) imposed to $x(\cdot)$, here we call $C_\xi$ the admissible set of $x(\xi)$, which is assumed to be convex and closed for all $\xi \in \Xi$. Let us denote the feasible set of Problem (1) as $x(\cdot) \in \mathcal{C} \cap \mathcal{N}$, where

$$\mathcal{C} = \{x(\cdot) \in \ell^2 : x(\xi) \in C_\xi \ \forall \xi \in \Xi\}.$$ 

Then Problem (1) can be written as an optimization model in $\ell^2$:

$$\min_{x(\cdot)} \mathcal{G}(x(\cdot)) \quad \text{s.t.} \quad x(\cdot) \in \mathcal{C} \cap \mathcal{N}, \ \text{where} \ \mathcal{G}(x(\cdot)) := \mathbb{E}_\xi[g(x(\xi), \xi)]. \quad (2)$$

The optimality condition, both necessary and sufficient, for problem (2) can be written as an SVI in the sense of Rockafellar and Wets [10]

$$- \nabla \mathcal{G}(x(\cdot)) \in \mathcal{N}_{\mathcal{C} \cap \mathcal{N}}(x(\cdot)) \quad (3)$$

$$= \{w(\cdot) \in \ell^2 : \langle w(\cdot), z(\cdot) \rangle \leq 0, \ \forall z(\cdot) \in \mathcal{C} \cap \mathcal{N}\}.$$
where $N$ stands for the normal cone. Under certain constraint qualifications, e.g., $r_i \cap N \neq \emptyset$, or simply $C \cap N \neq \emptyset$ if the sets $C_\xi$ are all polyhedral, the following equality holds.

$$N_{C \cap N}(x) = N_C(x) + N_N(x).$$

Since $N$ is a subspace, which yields that $N_N(x) = N_{\perp} := M$. Under the constraint qualification, the SVI (3) is equivalent to the following extensive form of SVI.

$$\exists x(\cdot) \in N, w(\cdot) \in M, \text{ such that } -\nabla G(x(\cdot)) - w(\cdot) \in N_C(x(\cdot)). \quad (4)$$

The progressive hedging algorithm for problem (2) corresponds to applying the proximal point algorithm [5] to a maximal monotone mapping $T$ derived from the gradients and normal cones in (4). For details the reader is referred to Rockafellar and Sun [8]. It proceeds as follows.

**Algorithm 1. PHA for MSO**

Given $x^\nu(\cdot) \in N$ and $w^\nu(\cdot) \in M$, $r > 0$.

**Step 1.** $\hat{x}^\nu(\xi) := \arg \min_{x \in C_\xi} g(x, \xi) + \langle w^\nu(\xi), x \rangle + \frac{r}{2} \| x - x^\nu(\xi) \|^2, \forall \xi$.

**Step 2.** $x^{\nu+1}(\cdot) = P_N(\hat{x}^\nu(\cdot))$, where $P$ stands for “projection”.

**Step 3.** $w^{\nu+1}(\cdot) = w^\nu(\cdot) + rP_M(\hat{x}^\nu(\cdot))$.

Set $\nu := \nu + 1$, repeat.

Rockafellar and Wets [9] showed that the above algorithm generates a sequence converging to a solution to (2) as long as such a solution exists. There is no restriction of the value of $r$ as long as being positive. However, a suitable choice of $r$ is crucial for fast convergence of Algorithm 1, see [8].

Knowing so much for PHA, we next introduce the (2-block) ADMM. Given two proper convex functions $f_1(u)$ and $f_2(v)$ both from $\mathbb{R}^n$ to $(-\infty, +\infty]$, consider a minimization problem in the following form.

$$\min_{x,v} f_1(x) + f_2(v) \quad \text{s.t. } x = v. \quad (5)$$

Its augmented Lagrangian function is as follows:

$$L_\sigma(x, v, w) := f_1(x) + f_2(v) + \langle w, x - v \rangle + \frac{\sigma}{2} \| x - v \|^2.$$ 

The ADMM applied to solve problem (5) proceeds as follows.

**Algorithm 2. ADMM**

Given $x^\nu, v^\nu, w^\nu, \sigma > 0, \tau \in (0, \frac{\sqrt{5-1}}{2})$.

**Step 1.** $x^{\nu+1} = \arg \min_x L_\sigma(x, v^\nu, w^\nu)$.

**Step 2.** $v^{\nu+1} = \arg \min_v L_\sigma(x^{\nu+1}, v, w^\nu)$.

**Step 3.** $w^{\nu+1} = w^\nu + \tau \sigma (x^{\nu+1} - v^{\nu+1})$.

Set $\nu := \nu + 1$, repeat.

For a more general constraint of $x = Av$ in Hilbert spaces, Gabay and Mercier [3] showed that the ADMM generates a sequence $\{x^\nu, v^\nu\} \to (x^*, v^*)$, a solution to (5), as long as $f_1$ is strictly convex and $A$ is an isomorphism. Later, Fazel et al
Theorem 2.1. Let \( f_1(x(\cdot)) := \mathbb{E}_\mathcal{Z}[g(x(\xi), \xi) + \delta C_\xi(x(\xi))] \) and \( f_2(v(\cdot)) := \delta N(v(\cdot)) \), where \( \delta \) is the indicator function. Suppose that \( \text{ri} C \cap N \neq \emptyset \) or \( \text{ri} C \cap N \neq \emptyset \) with \( C \) being polyhedral. Then, starting with \( x^0(\cdot), v^0(\cdot) \in N \) and \( w^0(\cdot) \in M \), the ADMM for (5) generates the same sequence \( \{x^\nu(\cdot), w^\nu(\cdot)\} \) as the PHA for (2) if \( r = \sigma, \tau = 1 \).

Proof. Note that, \( L_\sigma(x(\cdot), v(\cdot), w(\cdot)) \) is strictly convex for \( \sigma > 0 \) with respect to \( x(\cdot) \) and \( v(\cdot) \), and the constraint qualification \( \text{ri} C \cap N \) implies \( \text{ri} C_\xi \neq \emptyset \) (or \( C_\xi \neq \emptyset \) with \( C_\xi \) being polyhedral) for every \( \xi \). Thus Step 1 and Step 2 in Algorithm 2 require \( x^{\nu+1}(\cdot) \) and \( v^{\nu+1}(\cdot) \) as necessary conditions to satisfy the following, respectively:

(i) \( 0 \in \nabla g(x^{\nu+1}(\xi), \xi) + N_{C_\xi}(x^{\nu+1}(\xi)) + w^{\nu}(\xi) + \sigma(x^{\nu+1}(\xi) - v^{\nu}(\xi)) \forall \xi \),

(ii) \( 0 \in M - w^{\nu}(\cdot) + \sigma(v^{\nu+1}(\cdot) - x^{\nu+1}(\cdot)) \).

Starting from \( w^\nu(\cdot) \in M \), one has \( x^{\nu+1}(\cdot) - v^{\nu+1}(\cdot) \in M \) by (ii), then after Step 3 of ADMM \( w^{\nu+1}(\cdot) \in M \), which implies \( w^\nu(\cdot) \in M \) for all \( \nu \).

We next deduce a formula for \( v^{\nu+1}(\cdot) \) in ADMM. Note that \( v^{\nu+1}(\cdot) \) has an explicit solution form in Step 2, namely

\[
v^{\nu+1}(\cdot) = \arg\min_{v \in N} - \langle w^{\nu}(\cdot), v(\cdot) \rangle + \frac{\sigma}{2} \|x^{\nu+1}(\cdot) - v(\cdot)\|^2 = P_N(x^{\nu+1}(\cdot)),
\]

where the last equality follows from \( w^{\nu}(\cdot) \in M \).

Hence the steps in Algorithm 2 are equivalent to

- In Step 1, solving \( x^{\nu+1}(\xi) \) via
  \[-\nabla g(x^{\nu+1}(\xi), \xi) - w^{\nu}(\xi) + \sigma(x^{\nu+1}(\xi) - v^{\nu}(\xi)) \in N_{C_\xi}(x^{\nu+1}(\xi)), \forall \xi,\]

- In Step 2, setting \( v^{\nu+1}(\cdot) = P_N(x^{\nu+1}(\cdot)) \), and

- In Step 3, setting
  \[
  w^{\nu+1}(\cdot) = w^\nu(\cdot) + \tau \sigma(x^{\nu+1}(\cdot) - v^{\nu+1}(\cdot)) = w^\nu(\cdot) + \tau \sigma P_M(x^{\nu+1}(\cdot)).
  \]

Note that these steps are exactly the same as PHA with \( r = \sigma, \tau = 1 \) and \( \dot{x}^\nu(\cdot) = x^{\nu+1}(\cdot) \).

In summary, if we regard \( v^{\nu+1}(\cdot) \) in ADMM as \( x^{\nu+1}(\cdot) \) in PHA and regard \( x^{\nu+1}(\cdot) \) in ADMM as \( \dot{x}^\nu(\cdot) \) in PHA, Algorithm 2 indeed generates the same sequence as Algorithm 1 in solving MSO. \( \square \)

Note that recent progress in ADMM allows \( \sigma \) to change with \( \nu \) as long as it is bounded away from zero, which offers more flexibility in choosing parameters in the PHA to speed up the convergence. However, we would not explore in this direction in this paper, instead, we next consider a new PHA, which can be used in risk-averse models of MSO and can not be handled by the original version of PHA.

3. PHA for MSO with cross constraints. In risk-averse applications of the MSO models, it often comes with additional constraints of the form

\[Bx(\cdot) \leq b,\]
where $B$ is a linear operator from $\ell^2$ to $\mathbb{R}^n$ and is defined through

$$Bx(\cdot) = \mathbb{E}_\xi[B(\xi)x(\xi)], \quad \text{where } B(\xi) \in \mathbb{R}^{x \times n} \forall \xi.$$

We call system (6) the cross constraint since it involves all scenarios, rather than a single one. In particular, the expectation of $x(\cdot)$ is a such linear operator from $\ell^2$ to $\mathbb{R}^n$ defined as

$$E : \ell^2 \to \mathbb{R}^n \text{ such that } Ex(\cdot) = \mathbb{E}_\xi[x(\xi)] = \sum_{\xi} p(\xi) x(\xi). \quad (7)$$

We call the following problem MSO with cross constraints (MSOCC)

$$\min_{x(\cdot)} \mathcal{G}(x(\cdot)) := \mathbb{E}_\xi[g(x(\xi),\xi)] \quad \text{s.t. } x(\cdot) \in \mathcal{N} \cap \mathcal{C} \cap \mathcal{S},$$

where $\mathcal{S} := \{x(\cdot) \in \ell^2 : Bx(\cdot) \leq b\}$.

An example arises in the recent work of Rockafellar [6] and Sun et al [12] on risk measure optimization. Suppose we would like to minimize the mean-absolute-deviation of $g(x(\xi),\xi)$, which is a risk-averse measure for $g(x(\xi),\xi)$, see Ang et al [1] for details. Then, it means that we need to solve

$$\min_{x(\cdot)} \{\mathbb{E}_\xi[g(x(\xi),\xi)] + \lambda\|g(x(\xi),\xi)\|\} \quad \text{s.t. } x(\cdot) \in \mathcal{C} \cap \mathcal{N},$$

where $0 < \lambda < 1$ and $g(x(\xi),\xi) := \max\{g(x(\xi),\xi),0\}$. This problem can be equivalently converted to (see details in [12])

$$\min_{x(\cdot),y} \left\{y + \lambda \mathbb{E}_\xi[(g(x(\xi),\xi) - y) I] + (\mathbb{E}_\xi[g(x(\xi),\xi) - y]) I\right\} \quad \text{s.t. } x(\cdot) \in \mathcal{C} \cap \mathcal{N}, y \in \mathbb{R}.$$ 

Since the objective function of this problem is not smooth, we introduce new variables $t \in \mathbb{R}$, $u(\xi) \in \mathbb{R}$ for every $\xi$, and new constraints

$$u(\xi) \geq g(x(\xi),\xi) - y, \quad u(\xi) \geq 0, \quad t \geq \mathbb{E}_\xi[g(x(\xi),\xi) - y], \quad t \geq 0, \quad (8)$$

and change the objective function to $\mathbb{E}_\xi[y + \lambda u(\xi) + t]$. Then it is not hard to see that the third constraint in (8) is a cross constraint in the enlarged space of $(y,t,x(\cdot),u(\cdot))$.

Since the new constraint $Bx(\cdot) \leq b$ is generally dependent on all $\xi$ as shown in (7), the system (6) is a barrier for the decomposability requirement of the PHA over the scenario space (see Step 1 of Algorithm 1). Thus, the current form of PHA cannot be applied. In the following, we propose a strategy which keeps the decomposability of the PHA by introducing some slack variables.

Consider an SVI with cross constraints in its general form. Find $x(\cdot) \in \ell^2$ such that

$$- \nabla \mathcal{G}(x(\cdot)) \in N_{\mathcal{C} \cap \mathcal{N} \cap \mathcal{S}}(x(\cdot)). \quad (9)$$

Note that, by introducing slack variable $y(\cdot) \in \mathbb{R}^n$, the cross system (6) can be equivalently reformulated as follows.

$$Bx(\cdot) \leq b \iff \mathbb{E}_\xi[B(\xi)x(\xi)] \leq b \iff \exists y(\xi) : B(\xi)x(\xi) - b \leq y(\xi) \forall \xi \text{ and } \mathbb{E}_\xi[y(\xi)] = 0.$$

Let 

$$z(\cdot) = \begin{pmatrix} x(\cdot) \\ y(\cdot) \end{pmatrix}, \quad \bar{C} = \{z(\cdot) \mid x(\xi) \in C_\xi, \ B(\xi)x(\xi) - b \leq y(\xi) \forall \xi\}, \text{ and} \quad \mathcal{N} = \{z(\cdot) \mid x(\cdot) \in \mathcal{N}, \ \mathbb{E}_\xi[y(\xi)] = 0\} := \mathcal{N} \times \mathcal{N}_1, \quad \text{where } \mathcal{N}_1 = \{y(\cdot) : \mathbb{E}_\xi[y(\xi)] = 0\},$$

where $\mathcal{N}_1 = \{y(\cdot) : \mathbb{E}_\xi[y(\xi)] = 0\}.$
Then MSOCC can be written as
\[
\min_{z(\cdot),v(\cdot)} \ f_1(z(\cdot)) + f_2(v(\cdot)) \quad \text{s.t.} \quad z(\cdot) = v(\cdot) \tag{10}
\]
where \( f_1(z(\cdot)) = \mathbb{E}_\xi[g(x(\xi),\xi)] + \delta_N(z(\cdot)) \), \( f_2(v(\cdot)) = \delta_N(v(\cdot)) \).

Applying ADMM (Algorithm 2) to problem (10) and using the same approach as in the proof of Theorem 2.1, we obtain the following algorithm: from \( z^{\nu}(\cdot) \), \( v^{\nu}(\cdot) \) and \( \bar{w}^{\nu}(\cdot) \),
\begin{itemize}
  \item In Step 1, solving \( z^{\nu+1}(\xi) \) for every \( \xi \) via
    \[
    \min_{z := (x,y)} \quad g(x,\xi) + \langle \bar{w}^{\nu}(\xi), z \rangle \quad \text{s.t.} \quad x \in C_\xi, \ B(\xi)x - b \leq y,
    \]
    where \( \bar{w}^{\nu}(\cdot) := (w^{\nu}(\cdot),q^{\nu}(\cdot)) \in \bar{N}^+ \).
  \item In Step 2, setting \( v^{\nu+1}(\cdot) = P_{\bar{N}}(z^{\nu+1}(\cdot)) \),
  \item In Step 3, setting \( \bar{w}^{\nu+1}(\cdot) = \bar{w}^{\nu}(\cdot) + \tau \sigma P_{\bar{N}_1}(z^{\nu+1}(\cdot)) \).
\end{itemize}

We need to explain what are the meanings of \( P_{\bar{N}}(z(\cdot)) \) and \( P_{\bar{N}_1}(z(\cdot)) \). Note that \( \bar{N} = N \times N_1 \), therefore
\[
P_{\bar{N}}(z(\cdot)) = \begin{pmatrix} P_N(x(\cdot)) \\ P_{N_1}(y(\cdot)) \end{pmatrix} \quad \text{and} \quad P_{\bar{N}_1}(y(\cdot))(\xi) = y(\xi) - \mathbb{E}_\xi[y(\xi)].
\]
Moreover,
\[
P_{\bar{N}_1}(z(\cdot)) = \begin{pmatrix} P_M(x(\cdot)) \\ P_{N_1}(y(\cdot)) \end{pmatrix} = \bar{w}(\cdot) \quad \text{and} \quad P_{\bar{N}_1}(y(\cdot))(\xi) = \mathbb{E}_\xi[y(\xi)] := q(\xi).
\]
Hence we have \( \bar{w}(\cdot) = (w(\cdot), q(\cdot)) \).

In summary, the ADMM for solving (10) can be stated as follows.

\begin{center}
\textbf{Algorithm 3. ADMM for MSOCC}
\end{center}

\begin{center}
\textbf{Given} \( x^{\nu}(\cdot) \in \bar{N}, w^{\nu}(\cdot) \in M \) and \( y^{\nu}(\cdot) \in N_1, q^{\nu}(\cdot) \in N_1^+ \).
\end{center}

\begin{center}
\textbf{Step 1.} Obtain \( (\hat{x}^{\nu}(\xi), \hat{y}^{\nu}(\xi)) \) for every \( \xi \) via
\end{center}

\[
\min_{x,y} \quad g(x,\xi) + \langle w^{\nu}(\xi), x \rangle + \langle q^{\nu}(\xi), y \rangle \quad + \frac{\sigma}{2} \| x - x^{\nu}(\xi) \|^2 + \frac{\sigma}{2} \| y - y^{\nu}(\xi) \|^2,
\quad \text{s.t.} \quad x \in C_\xi, \ B(\xi)x - b \leq y.
\]

\begin{center}
\textbf{Step 2.} \( x^{\nu+1}(\cdot) = P_N(\hat{x}^{\nu}(\cdot)), \ y^{\nu+1}(\xi) = \hat{y}^{\nu}(\xi) - \mathbb{E}_\xi[\hat{y}^{\nu}(\xi)] \).
\end{center}

\begin{center}
\textbf{Step 3.} \( w^{\nu+1}(\cdot) = w^{\nu}(\cdot) + \tau \sigma P_M(\hat{x}^{\nu}(\cdot)), \ q^{\nu+1}(\xi) = q^{\nu}(\xi) + \tau \sigma \mathbb{E}_\xi[\hat{y}^{\nu}(\xi)] \).
\end{center}

Set \( \nu := \nu + 1 \), repeat.

Based on the convergence result of ADMM, it would be a straightforward matter to show the convergence of Algorithm 3. We summarize the result in a theorem.

\begin{theorem}
Suppose that \( g(x(\xi),\xi) \) is convex and smooth for all \( \xi \) and problem (9) satisfies constraint qualification, i.e., one of the following two conditions holds.
\begin{itemize}
  \item Both \( C \) and \( S \) are convex polyhedra and \( C \cap N \cap S \neq \emptyset \);
  \item \( r_i(C) \cap N \cap r_i(S) \neq \emptyset \).
\end{itemize}
If the solution set to problem (9) is nonempty, then the sequence generated by Algorithm 3 converges to a point in this set, namely
\[
\{(x^{\nu}(\cdot), y^{\nu}(\cdot), w^{\nu}(\cdot), q^{\nu}(\cdot))\} \to \{(x^*(\cdot), y^*(\cdot), w^*(\cdot), q^*(\cdot))\} \quad \text{when} \ \nu \to +\infty,
\]
where \( x^*(\cdot) \) and \( w^*(\cdot) \) are a primal and a dual optimal solution to (9), respectively.

It should be noted that Theorem 3.1 can be proved by viewing Algorithm 3 as a more general PHA with respect to $z(\cdot)$ and $\bar{N}$, and employing the Rockafellar-Sun approach in [8], i.e., taking Algorithm 3 for $\tau = 1, \sigma = r$ and treat it as a special proximal point method, then Theorem 3.1 is true and we can obtain stronger results on the rate of convergence as follows.

**Theorem 3.2.** Under the conditions of Theorem 3.1, if $g(x(\xi), \xi)$ is convex quadratic and both $C$ and $S$ are convex polyhedra, then the sequence converges at $q$-linear rate with respect to

$$
\|x^{\nu}(\cdot) - x^*(\cdot)\|^2 + \|y^{\nu}(\cdot) - y^*(\cdot)\|^2 + \frac{1}{r^2} \|w^{\nu}(\cdot) - w^*(\cdot)\|^2 + \frac{1}{r^2} \|q^{\nu}(\cdot) - q^*(\cdot)\|^2.
$$

The proof is totally similar to the one used in [8]. For brevity, we omit it.

4. **Conclusions.** The progressive hedging algorithm for MSO generates the same convergence sequence as the ADMM if both methods start from the same initial point and take $\tau = 1$ and $\sigma = r$ in the ADMM. This fact, on one hand, offers flexibility in selecting step lengths for PHA, and, on the other hand, strengthens the convergence result of ADMM by providing a $q$-linear rate for the linear-quadratic MSO problems. The connection between these two methods can results in new algorithms for MSO with cross constraints. It is our belief that this idea can be further used in developing more PHA algorithms for more complex MSO problems.

**Acknowledgments.** This research was partially supported by Australian Research Council under Grant DP160102819.

**REFERENCES**

[1] M. Ang, J. Sun and Q. Yao, On the dual representation of coherent risk measures, *Ann. Oper. Res.*, 262 (2018), 29–46.
[2] M. Fazel, T. K. Pong, D. Sun and P. Tseng, Hankel matrix rank minimization with applications to system identification and realization, *SIAM J. Matrix Anal. Appl.*, 34 (2013), 946–977.
[3] D. Gabay and B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximations, *Comput. Math. Appl.*, 2 (1976), 17–40.
[4] J. J. Moreau, Proximité et dualité dans un espace Hilbertien, *Bull. Soc. Math. de France*, 93 (1975), 273–299.
[5] R. T. Rockafellar, Monotone operators and the proximal point algorithm, *SIAM J. Control Optim.*, 14 (1976), 877–898.
[6] R. T. Rockafellar, Solving stochastic programming problems with risk measures by progressive hedging, *Set-Valued Var. Anal.*, 26 (2018), 759–768.
[7] R. T. Rockafellar, Progressive decoupling of linkages in monotone variational inequalities and convex optimization, in *Proceedings of the Conference on Nonlinear Analysis and Convex Analysis*, Chitose, Japan, (2017).
[8] R. T. Rockafellar and J. Sun, Solving monotone stochastic variational inequalities and complementarity problems by progressive hedging, *Math. Program.*, (2018).
[9] R. T. Rockafellar and R. J. B. Wets, Scenario and policy aggregation in optimization under uncertainty, *Math. Oper. Res.*, 16 (1991), 119–147.
[10] R. T. Rockafellar and R. J. B. Wets, Stochastic variational inequalities: Single-stage to multistage, *Math. Program.*, 135 (2016), 331–360.
[11] J. E. Spingarn, Applications of the method of partial inverses to convex programming: decomposition, *Math. Program.*, 32 (1985), 199–223.
[12] J. Sun, X. M. Yang, Q. Yao and M. Zhang, Risk minimization, regret minimization and progressive hedging algorithms, work in process.
Received for publication August 2018.

E-mail address: jsun@nus.edu.sg
E-mail address: h.xu@curtin.edu.au
E-mail address: min.zhang2@curtin.edu.au