CrossMark

Solving monotone stochastic variational inequalities and complementarity problems by progressive hedging

R. Tyrrell Rockafellar[1] · **Jie Sun**[2]

Abstract The concept of a stochastic variational inequality has recently been articulated in a new way that is able to cover, in particular, the optimality conditions for a multistage stochastic programming problem. One of the long-standing methods for solving such an optimization problem under convexity is the progressive hedging algorithm. That approach is demonstrated here to be applicable also to solving multistage stochastic variational inequality problems under monotonicity, thus increasing the range of applications for progressive hedging. Stochastic complementarity problems as a special case are explored numerically in a linear two-stage formulation.

✉ R. Tyrrell Rockafellar
rtr@uw.edu
http://www.math.washington.edu/~rtr/mypage.html

Jie Sun
jie.sun@curtin.edu.au
http://oasisapps.curtin.edu.au/staff/profile/view/Jie.Sun

1 Department of Mathematics, University of Washington, Box 354350, Seattle, WA 98195-4350, USA

2 Department of Mathematics and Statistics, Curtin University, Bentley 6102, Australia

1 Introduction

Stochastic programming problems aim at determining optimal responses to information as it becomes available. In a multistage formulation with convexity, the optimality conditions in such a problem take the form of a monotone variational inequality with stochastic structure. Interactions with statistics are rich and extensive, but also challenging in how properly to deal with emerging information and generate effective surrogates for conditional expectations and other data features important to making decisions. Problems of approximation in statistics offer, in turn, wide territory for applying techniques of stochastic optimization.

Variational inequality problems, with complementary problems as a special case, have been widely recognized as useful not only in capturing optimality conditions but also in characterizing equilibrium in game-type models of interaction, cf. [1]. When the underlying applications involve uncertainty, it is natural to think of the conditions as leading to a "stochastic" variational inequality. But the formulation of what *problem* is to be solved under that heading has gone off in completely different directions. One line of thought, the ERM approach occupying much of the literature up to now, considers not a single variational inequality but a probablistic collection of separate variational inequalities and views the associated problem as that of an element that "comes closest" to solving all of them simultaneously.[1] Contributions in that direction are seen for instance in [2–8]. The ET approach, in contrast, takes stochastic variational inequality problems to be just like deterministic variational inequality problems but with the mapping in question expressed by an expectation formula. This direction is followed for instance in [9–12]. The ERM and ET developments have mostly concentrated on single-stage models, not involving anything like a recourse decision. On the other hand, there has been work on serious applications to power systems, for example in [13–15], where multistage game-like models with stochastic features are set up as giant variational inequality problems—without attention to what, in general, might deserve to be called a "stochastic variational inequality problem."

In [16], in an effort to lay a foundation for a broader theory in the subject, Rockafellar and Wets extended the EV approach from single stage to multistage modeling in a manner capable of covering the optimality conditions in multistage stochastic programming that involve prices for information. Such conditions support stochastic decomposition, which might be valuable in particular in game-like applications such as to energy systems, although information prices have not yet been exploited in that context.

Here we take the extension in [16] as our platform and focus on how solutions to *monotone* stochastic variational inequalities of multistage type may be computed. We show that the progressive hedging algorithm in convex stochastic programming [17] carries over to this larger topic with hardly any changes except for its mode of articulation. Although we don't get into statistical issues as such, they are present for

---

[1] The analog in numerical analysis lies in the theory of overdetermined systems of equations generated by noise in measurements of parameters.

instance in the eventual need to understand how the solutions we obtain depend on how the probability space is set up—a topic for investigation elsewhere.[2]

A review of the variational inequalities we wish to solve and their counterparts in stochastic programming will furnish a starting point. To avoid distractions of infinite dimensionality and to mesh with [16], where more details can be found along with references to the history and literature of the subject, we consider here a *finite* set Ξ of scenarios $\xi = (\xi_1, \ldots, \xi_N)$ composed of elements ξ_k in sets Ξ_k that are regarded as being revealed sequentially in N stages. Each scenario ξ has a known probability $p(\xi) > 0$, and these probabilities add to 1. In this way Ξ is a probability space. Our attention is directed to mappings that designate responses to the scenarios in Ξ in the notation

$$x(\cdot) : \xi \mapsto x(\xi) = (x_1(\xi), \ldots, x_N(\xi)) \in I\!R^{n_1} \times \cdots \times I\!R^{n_N} = I\!R^n. \quad (1.1)$$

The linear space $\mathcal{L}$ consisting of all such mappings $x(\cdot)$ from Ξ to $I\!R^n$ is given the expectational inner product

$$\langle x(\cdot), w(\cdot) \rangle = E_\xi[x(\xi) \cdot w(\xi)] = \sum_{\xi \in \Xi} p(\xi) \sum_{k=1}^{N} x_k(\xi) \cdot w_k(\xi), \quad (1.2)$$

which makes it into a finite-dimensional Hilbert space. Our real interest centers though 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 ξ realized in earlier stages:

$$x(\xi) = (x_1, x_2(\xi_1), x_3(\xi_1, \xi_2), \ldots, x_N(\xi_1, \xi_2, \ldots, \xi_{N-1})) \text{ under nonanticipativity.}$$
$$(1.3)$$

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

$$\mathcal{N} = \big\{ x(\cdot) \in \mathcal{L} \,\big|\, x_k(\xi_1, \ldots, \xi_{k-1}, \xi_k, \ldots, \xi_N) \text{ doesn't depend on } \xi_k, \ldots, \xi_N \big\}.$$
$$(1.4)$$

Also important then is the subspace of $\mathcal{L}$ that is orthogonally complementary to $\mathcal{N}$ with respect to the inner product (1.2), $\mathcal{M} = \mathcal{N}^\perp$. It is given by

$$\mathcal{M} = \big\{ w(\cdot) \in \mathcal{L} \,\big|\, E_{\xi \,|\, \xi_1, \ldots, \xi_{k-1}} w_k(\xi) = 0, \ \forall k \big\}. \quad (1.5)$$

where the expectation is the conditional expectation over the remaining possibilities for the scenario ξ given that the portion $(\xi_1, \ldots, \xi_{k-1})$ is already known.

---

[2] In part, this has recently been taken up by Chen et al. in [18] in terms of discrete approximations to two-stage stochastic variational inequality problems based on "continuous" probability instead of "discrete" probability, as here.

The complementary subspaces $\mathcal{N}$ and $\mathcal{M}$ will be central in what follows whether we are dealing with multistage stochastic programming or stochastic variational inequalities more generally. So too will be response constraints of the form

$$x(\xi) \in C(\xi) \text{ for all } \xi \in \Xi \tag{1.6}$$

in which each $C(\xi)$ is assumed to be a nonempty closed convex subset of $I\!R^n$. The set of all $x(\cdot)$ satisfying (1.6) forms a nonempty closed convex subset of $\mathcal{L}$ which we denote by $\mathcal{C}$.

For the response functions $x(\cdot)$ in $\mathcal{C} \cap \mathcal{N}$ (assumed to be *nonempty*), we consider an optimization problem in convex stochastic programming and the corresponding variational inequality characterization of optimality, and then widen the picture to monotone stochastic variational inequalities more broadly.

Adhering to a context of differentiability to keep matters simple (although subgradients could be brought in instead), we consider now for each $\xi \in \Xi$ a continuously differentiable convex function $g(\cdot, \xi) : C(\xi) \rightarrow I\!R$ and get

$$\mathcal{G} : x(\cdot) \rightarrow E_\xi[g(x(\xi), \xi)] \tag{1.7}$$

as a continuously differentiable convex function on $\mathcal{C}$ with gradient mapping $\nabla\mathcal{G} : \mathcal{C} \rightarrow \mathcal{L}$, where $\nabla\mathcal{G}(x(\cdot))$ is the element of $\mathcal{L}$ that assigns to ξ the vector $\nabla_x g(x(\xi), \xi) \in I\!R^n$. The associated stochastic programming problem is to

$$\text{minimize } \mathcal{G}(x(\cdot)) \text{ over all } x(\cdot) \in \mathcal{C} \cap \mathcal{N}. \tag{1.8}$$

and the corresponding first-order condition for the optimality of $x(\cdot)$, both necessary and sufficient, is

$$-\nabla\mathcal{G}(x(\cdot)) \in N_{\mathcal{C} \cap \mathcal{N}}(x(\cdot)), \tag{1.9}$$

where $N_{\mathcal{C} \cap \mathcal{N}}(x(\cdot))$ is the normal cone to $\mathcal{C} \cap \mathcal{N}$ at $x(\cdot)$. This variational inequality in $\mathcal{L}$ over the set $\mathcal{C} \cap \mathcal{N}$ is closely related to an alternative condition,

$$x(\cdot) \in \mathcal{N} \text{ and } \exists\, w(\cdot) \in \mathcal{M} \text{ such that } -\nabla\mathcal{G}(x(\cdot)) - w(\cdot) \in N_{\mathcal{C}}(x(\cdot)), \tag{1.10}$$

in which the element $w(\cdot)$ from the complementary space $\mathcal{M}$ in (1.5) has the role of a Lagrange multiplier for the constraint of nonanticipativity. Specifically, as shown in [16], (1.10) is always sufficient for (1.9) and it is necessary under a constraint qualification on the intersection of $\mathcal{C}$ with $\mathcal{N}$. For instance it holds if there is an element $\hat{x}(\cdot)$ of $\mathcal{N}$ such that $\hat{x}(\xi) \in \text{ri}\, C(\xi)$ for all $\xi \in \Xi$, or on the other hand if the sets $C(\xi)$ in (1.4) are all polyhedral.

The progressive hedging algorithm [17] for solving (1.9) is based on solving (1.10) and taking advantage of the fact in (1.10) that

$$\begin{aligned} -\nabla\mathcal{G}(x(\cdot)) - w(\cdot) \in N_{\mathcal{C}}(x(\cdot)) \iff &\quad -\nabla_x g(x(\xi), \xi) \\ -w(\xi) \in N_{C(\xi)}(x(\xi)), \,\forall\xi. \end{aligned} \tag{1.11}$$

This decomposition into a separate "deterministic" problem for each scenario ξ is a powerful feature.

The variational inequality (1.9) is of maximal monotone type because the convexity of $\mathcal{G}$ guarantees the monotonicity of $\nabla\mathcal{G}$ while the continuity of $\nabla\mathcal{G}$ on $\mathcal{C}$, and hence on $\mathcal{C} \cap \mathcal{N}$, then makes $\nabla\mathcal{G} + N_{\mathcal{C}\cap\mathcal{N}}$ be a maximal monotone mapping from $\mathcal{L}$ into itself.

Building on this model, the concept of a multistage stochastic variational inequality in "basic form" was introduced in [16] as referring more generally to

$$- \mathcal{F}(x(\cdot)) \in N_{\mathcal{C}\cap\mathcal{N}}(x(\cdot)) \tag{1.12}$$

in the case of a mapping $\mathcal{F} : \mathcal{L} \to \mathcal{L}$ that substitutes for $\nabla\mathcal{G}$ in (1.9) by replacing the gradients $\nabla_x g(x(\xi), \xi)$ by expressions $F(x(\xi), \xi)$. In other words, instead of starting with functions $g(\cdot, \xi)$ in the sets $C(\xi)$ and passing to their gradients, the idea is to start directly with mappings

$$F(\cdot, \xi) : I\!R^n \to I\!R^n \text{ for } \xi \in \Xi \tag{1.13}$$

and get $\mathcal{F} : \mathcal{L} \to \mathcal{L}$ through defining the image $\mathcal{F}(x(\cdot))$ in $\mathcal{L}$ of an element $x(\cdot)$ of $\mathcal{L}$ under $\mathcal{F}$ by

$$\mathcal{F}(x(\cdot)) \text{ takes } \xi \in \Xi \text{ to } F(x(\xi), \xi) \in I\!R^n. \tag{1.14}$$

It is directly assumed in this, for present purposes, that $F(\cdot, \xi)$ is continuous and monotone on $C(\xi)$, so that $\mathcal{F}$ is continuous and monotone as a mapping from $\mathcal{C}$ to $\mathcal{L}$. This ensures that $\mathcal{F} + N_{\mathcal{C}}$ is maximal monotone, and the same for $\mathcal{F} + N_{\mathcal{C}\cap\mathcal{N}}$, and it puts the variational inequality (1.12) in the category we want to focus on.

Just as in the optimization case, there is an alternative condition closely related to (1.12), namely

$$x(\cdot) \in \mathcal{N} \text{ and } \exists\, w(\cdot) \in \mathcal{M} \text{ such that } - \mathcal{F}(x(\cdot)) - w(\cdot) \in N_{\mathcal{C}}(x(\cdot)). \tag{1.15}$$

Again, any solution to (1.15) is a solution to (1.12), and the converse holds under either of the constraint qualifications mentioned earlier. We refer to (1.15) as the stochastic variational inequality in "extensive form." Its special virtue is that the $w(\cdot)$-dependent variational inequality it proposes to solve over $\mathcal{C}$ instead of the original one over $\mathcal{C}\cap\mathcal{N}$ *breaks down into a separate variational inequality for each individual scenario*:

$$- F(x(\xi), \xi) - w(\xi) \in N_{C(\xi)}(x(\xi)) \text{ for all } \xi \in \Xi. \tag{1.16}$$

Our goal is to confirm that this kind of stochastic decomposition, already exploited by the progressive hedging algorithm in the stochastic programming case, can likewise work for solving the stochastic variational inequality (1.12) even when it does not stand for the optimality condition (1.9).

Special attention will be paid in this to *multistage stochastic complementarity problems*, by which we mean the case of (1.12) and (1.16) when $C(\xi) = I\!R^n_+$ for every ξ, so that

$$
\begin{aligned}
\mathcal{C} &= \big\{ x(\cdot) \in \mathcal{L} \,\big|\, x(\xi) \geq 0, \ \forall \xi \big\}, \\
-v(\cdot) &\in N_{\mathcal{C}}(x(\cdot)) \iff 0 \leq x(\xi) \perp v(\xi) \geq 0, \ \forall \xi.
\end{aligned}
\tag{1.17}
$$

Such problems have not previously been spotlighted, or for that matter even given a name.

2 Progressive hedging

We proceed now with describing the progressive hedging algorithm, first in the optimization framework of stochastic programming, second in the corresponding framework of gradient mappings, and third in the formulation we are proposing for stochastic variational inequalities in general. Throughout, we maintain the assumptions in the introduction with their consequences for convexity and monotonicity.

In all situations the projection mappings onto the subspaces $\mathcal{N}$ and $\mathcal{M}$ of $\mathcal{L}$ will serve as a key ingredient. The projection mapping $P_{\mathcal{N}}$ onto $\mathcal{N}$, which assigns to an element $\hat{x}(\cdot)$ of $\mathcal{L}$ the nearest element $x(\cdot)$ of $\mathcal{N}$, is simple to execute as follows:

$$
x(\cdot) = P_{\mathcal{N}}(\hat{x}(\cdot)) \text{ is obtained by setting } x_k(\xi) = E_{\xi \mid \xi_1, \dots, \xi_{k-1}} \hat{x}_k(\xi),
\tag{2.1}
$$

where the expectation is the conditional expectation already used in connection with the definition of $\mathcal{M}$ in (1.5). The projection mapping $P_{\mathcal{M}}$ onto $\mathcal{M}$ is equally available then because

$$
P_{\mathcal{M}} = I - P_{\mathcal{N}}.
\tag{2.2}
$$

All versions of the algorithm utilize not only $P_{\mathcal{N}}$ and $P_{\mathcal{M}}$ but also a parameter $r > 0$ which can affect performance if it is too high or too low. The choice of r is an issue that has long been familiar in stochastic programming and the experience gained there will have to be the guide in handling the new ranges of applications that we are opening up here.

We use $v = 1, 2, \dots$ to mark iterations. Iteration v starts in every case with elements $x^v(\cdot) \in \mathcal{N}$ and $w^v(\cdot) \in \mathcal{M}$ and produces the elements $x^{v+1}(\cdot) \in \mathcal{N}$ and $w^{v+1}(\cdot) \in \mathcal{M}$ that feed into the next iteration. In our notation we let $x(\xi)$ denote for each $\xi \in \Xi$ a variable in $I\!R^n$; the determination of the value of that variable for each ξ produces of course a mapping $x(\cdot) \in \mathcal{L}$.

Progressive hedging algorithm in stochastic programming: minimization mode.
Pass from having $x^v \in \mathcal{N}$ and $w^v \in \mathcal{M}$ to having $x^{v+1} \in \mathcal{N}$ and $w^{v+1} \in \mathcal{M}$ as follows in determining $\hat{x}^v(\cdot) \in \mathcal{L}$ by solving a separate problem for each scenario ξ to get $\hat{x}^v(\xi)$:

$$\hat{x}^\nu(\xi) = \mathrm{argmin}_{x(\xi) \in C(\xi)} \left\{ g(x(\xi), \xi) + w^\nu(\xi) \cdot x(\xi) + \tfrac{r}{2} ||x(\xi) - x^\nu(\xi)||^2 \right\}, \quad (2.3)$$
and then $x^{\nu+1}(\cdot) = P_\mathcal{N}(\hat{x}^\nu(\cdot))$ and $w^{\nu+1}(\cdot) = w^\nu(\cdot) + r P_\mathcal{M}(\hat{x}^\nu(\cdot))$.

Note that the vector $x(\xi)$ in (2.3) exists and is uniquely determined, because the proximal term forces the function being minimized to be strongly convex. This allows us to restate (2.3) in terms of a gradient condition for optimality instead of minimization and thereby get a form of the procedure will bridge into a version for variational inequalities more generally.

Progressive hedging algorithm in stochastic programming: gradient mode. Pass from having $x^\nu \in \mathcal{N}$ and $w^\nu \in \mathcal{M}$ to having $x^{\nu+1} \in \mathcal{N}$ and $w^{\nu+1} \in \mathcal{M}$ as follows in determining $\hat{x}^\nu(\cdot) \in \mathcal{L}$ by solving a separate problem for each scenario ξ to get $\hat{x}^\nu(\xi)$:

$$\hat{x}^\nu(\xi) = \text{the unique } x(\xi) \text{ such that}$$
$$-\nabla_x g(x(\xi), \xi) - w^\nu(\xi) - r[x(\xi) - x^\nu(\xi)] \in N_{C(\xi)}(x(\xi)), \quad (2.4)$$
and then $x^{\nu+1}(\cdot) = P_\mathcal{N}(\hat{x}^\nu(\cdot))$ and $w^{\nu+1}(\cdot) = w^\nu(\cdot) + r P_\mathcal{M}(\hat{x}^\nu(\cdot))$.

By putting $F(x(\xi), \xi)$ in the place of $\nabla_x g(x(\xi), \xi)$ we arrive at the proposed extension of the procedure as a means of solving the variational inequality problem (1.10).

Progressive hedging algorithm for stochastic variational inequalities. Pass from having $x^\nu \in \mathcal{N}$ and $w^\nu \in \mathcal{M}$ to having $x^{\nu+1} \in \mathcal{N}$ and $w^{\nu+1} \in \mathcal{M}$ as follows in determining $\hat{x}^\nu(\cdot) \in \mathcal{L}$ by solving a separate problem for each scenario ξ to get $\hat{x}^\nu(\xi)$:

$$\hat{x}^\nu(\xi) = \text{the unique } x(\xi) \text{ such that}$$
$$-F(x(\xi), \xi) - w^\nu(\xi) - r[x(\xi) - x^\nu(\xi)] \in N_{C(\xi)}(x(\xi)), \quad (2.5)$$
and then $x^{\nu+1}(\cdot) = P_\mathcal{N}(\hat{x}^\nu(\cdot))$ and $w^{\nu+1}(\cdot) = w^\nu(\cdot) + r P_\mathcal{M}(\hat{x}^\nu(\cdot))$.

In (2.5) the existence and uniqueness of the solution comes from the fact that the r-term makes the variational inequality be strongly monotone. In the optimization case this corresponds to strong convexity (with modulus r) of the function being minimized.

The progressive hedging algorithm in stochastic programming corresponds to applying the proximal point algorithm [19] to a maximal monotone mapping T derived from the gradients and normal cones in (2.4). Our key observation is that the underpinnings of that application and the convergence results they support are unaffected when $\nabla_x g(\cdot, \xi)$ is replaced by $F(\cdot, \xi)$ in our monotonicity setting.

Recall that for a maximal monotone mapping T the iterates of the proximal point algorithm with parameter $r > 0$ take the form

$$x^{\nu+1} = (I + r^{-1}T)^{-1}(x^\nu) \quad (2.6)$$

in exploiting the fact that the "resolvent" mapping $(I + r^{-1}T)^{-1}$ is single-valued and nonexpansive (Lipschitz continuous with constant 1). The fixed points of $(I + r^{-1}T)^{-1}$ are the elements of the closed convex set $T^{-1}(0)$. As long as that set is nonempty, convergence is guaranteed to a particular element $\bar{x}$ of that set, depending on the starting point, with $||x^{\nu+1} - \bar{x}|| < ||x^\nu - \bar{x}||$ unless already $x^\nu = \bar{x}$. However, as will

be important below, the algorithm can also be applied with "rescaling" in the sense of replacing T by ATA for a symmetric positive-definite and invertible linear mapping A. This preserves maximal monotonicity and leads instead to having the convergence obey

$$||A^{-1}(x^{\nu+1} - \bar{x})|| < ||A^{-1}(x^\nu - \bar{x})|| \text{ unless } x^\nu = \bar{x}. \tag{2.7}$$

Convergence at a linear rate is guaranteed in some circumstances. For instance it always holds in our finite-dimensional setting when T is piecewise polyhedral in the sense that the graph of T is the union of a finite collection of polyhedral convex sets.

The device used to obtain the crucial mapping T in our situation is that of taking a "partial inverse" of the mapping $\mathcal{F} + N_C$ in the manner o Spingarn [20] by regarding the space $\mathcal{L}$ as the direct sum of the subspaces $\mathcal{N}$ and $\mathcal{M}$. Each $x(\cdot) \in \mathcal{L}$ can be viewed uniquely as the sum of an element of $\mathcal{N}$ and an element of $\mathcal{M}$, namely $P_\mathcal{N}(x(\cdot))$ and $P_\mathcal{M}(x(\cdot))$, and these can be viewed as the two "componenents" of $x(\cdot)$ in that representation.

Theorem 1 (extended derivation of the progressive hedging algorithm) *The progressive hedging algorithm for stochastic variational inequalities corresponds to applying the proximal point algorithm, with rescaling, to determine a zero of the maximal monotone mapping $T : \mathcal{L} \rightrightarrows \mathcal{L}$ defined by*

$$z(\cdot) \in T(y(\cdot)) \iff P_\mathcal{N}(z(\cdot)) + P_\mathcal{M}(y(\cdot)) \in [\mathcal{F} + N_C](P_\mathcal{N}(y(\cdot)) + P_\mathcal{M}(z(\cdot))). \tag{2.8}$$

The rescaling replaces T by ATA for $A : u(\cdot) \mapsto P_\mathcal{N}(u(\cdot)) + r P_\mathcal{M}(u(\cdot))$.

Proof Iterations for ATA have the form $u^{\nu+1}(\cdot) = (I + r^{-1}ATA)^{-1}(u^\nu(\cdot))$, or

$$u^\nu(\cdot) \in (I + r^{-1}ATA)(u^{\nu+1}(\cdot)) = u^{\nu+1}(\cdot) + r^{-1}AT(Au^{\nu+1}(\cdot)),$$

which in the rescaling with $y^\nu(\cdot) = Au^\nu(\cdot)$ and $y^{\nu+1}(\cdot) = Au^{\nu+1}(\cdot)$ can be written as

$$A^{-1}y^\nu(\cdot) - A^{-1}y^{\nu+1}(\cdot) \in r^{-1}AT(y^{\nu+1}(\cdot)), \text{ or equivalently, } rA^{-2}[y^\nu(\cdot) \\ -y^{\nu+1}(\cdot)] \in T(y^{\nu+1}(\cdot)).$$

Because A^{-2} takes $y(\cdot)$ to $P_\mathcal{N}(y(\cdot)) + r^{-2}P_\mathcal{M}(y(\cdot))$, this has the form

$$P_\mathcal{N}(r[y^\nu(\cdot) - y^{\nu+1}(\cdot)]) + P_\mathcal{M}(r^{-1}[y^\nu(\cdot) - y^{\nu+1}(\cdot)]) \in T(P_\mathcal{N}(y^{\nu+1}(\cdot)) \\ +P_\mathcal{M}(y^{\nu+1}(\cdot)))$$

and corresponds under (2.8) to

$$P_\mathcal{N}\left(r[y^\nu(\cdot) - y^{\nu+1}(\cdot)]\right) + P_\mathcal{M}\left(y^{\nu+1}(\cdot)\right) \in [\mathcal{F} + N_C]\left(P_\mathcal{N}(y^{\nu+1}(\cdot)) \\ +P_\mathcal{M}(r^{-1}[y^\nu(\cdot) - y^{\nu+1}(\cdot)])\right). \tag{2.9}$$

In setting

$$x^\nu(\cdot) = P_\mathcal{N}(y^\nu(\cdot)), \ w^\nu(\cdot) = -P_\mathcal{M}(y^\nu(\cdot)), \ x^{\nu+1}(\cdot) = P_\mathcal{N}(y^{\nu+1}(\cdot)),$$
$$w^{\nu+1}(\cdot) = -P_\mathcal{M}(y^{\nu+1}(\cdot)),$$

we can pose the iterations expressed by (2.9) as

$$r[x^\nu(\cdot) - x^{\nu+1}(\cdot)] - w^{\nu+1}(\cdot) \in [\mathcal{F} + N_\mathcal{C}]\left(x^{\nu+1}(\cdot) + r^{-1}(w^{\nu+1} - w^\nu(\cdot)\right) \tag{2.10}$$

with respect to $x(\cdot)$-elements in $\mathcal{N}$ and $w(\cdot)$-elements in $\mathcal{M}$. If we now introduce $\hat{x}^\nu(\cdot)$ as standing for $x^{\nu+1}(\cdot) + r^{-1}(w^{\nu+1}(\cdot) - w^\nu(\cdot))$, we get

$$x^{\nu+1}(\cdot) = P_\mathcal{N}(\hat{x}^\nu(\cdot)), \qquad w^{\nu+1}(\cdot) = w^\nu(\cdot) + r P_\mathcal{M}(\hat{x}^\nu(\cdot)), \tag{2.11}$$

and thereby convert (2.10) into $r[x^\nu(\cdot) - P_\mathcal{N}(\hat{x}^\nu(\cdot))] - w^\nu(\cdot) - r P_\mathcal{M}(\hat{x}^\nu(\cdot)) \in [\mathcal{F} + N_\mathcal{C}](\hat{x}^\nu(\cdot))$, which is the same as $-w^\nu(\cdot) \in [\mathcal{F} + N_\mathcal{C}](\hat{x}^\nu(\cdot)) + r[\hat{x}^\nu(\cdot) - x^\nu(\cdot)]$ and decomposes in the scenario structure to

$$-w^\nu(\xi) \in [F(\cdot, \xi) + N_{C(\xi)}](\hat{x}^\nu(\xi)) + r[\hat{x}^\nu(\xi) - x^\nu(\xi)]. \tag{2.12}$$

Since the progressive hedging algorithm solves (2.12) for each ξ to construct $\hat{x}^\nu(\cdot)$ and then updates by (2.11), our goal has been reached. □

Theorem 2 (convergence of the algorithm) *As long as the (monotone) variational inequality (1.10) has at least one solution, the sequence of pairs $(x^\nu(\cdot), w^\nu(\cdot))$ generated by the progressive algorithm will converge to some pair $(\bar{x}(\cdot), \bar{w}(\cdot))$ satisfying (1.11) and thus furnish $\bar{x}(\cdot)$ as a solution to (1.10). In this convergence the distance expressions*

$$||x^\nu(\cdot) - \bar{x}(\cdot)||^2 + \frac{1}{r^2}||w^\nu(\cdot) - \bar{w}(\cdot)||^2 \tag{2.13}$$

will form a decreasing sequence that goes to zero. The decrease will surely be at a linear rate if, in particular, the sets $C(\xi)$ are polyhedral and the functions $F(\cdot, \xi)$ are affine.

Proof These properties translate from the corresponding ones for the proximal point algorithm by way of the derivation in Theorem 1. For the final claim, the key is the fact that T has its graph expressible as the union of finitely many convex polyhedral sets if and only if the mapping $\mathcal{F} + N_\mathcal{C}$ (for which it is a "partial inverse") has that property. When each $F(\cdot, \xi)$ is affine, $\mathcal{F}$ is affine and the question comes down to the piecewise polyhedral nature of the graph of $N_\mathcal{C}$. That property holds when $\mathcal{C}$ is a convex polyhedral set in $\mathcal{L}$ and therefore when each $C(\xi)$ is polyhedral. □

3 Application to stochastic complementarity problems

Our general concept of a multistage stochastic complementarity problem was explained at the end of Sect. 1, but now we add more detail in terms of components of the functions F_k underlying the mapping $\mathcal{F}$ with respect to IR^n being the product space $IR^{n_1} \times \cdots \times IR^{n_N}$:

$$F(x(\xi), \xi) = \Big(F_1(x_1(\xi), \ldots, x_N(\xi), \xi), \ldots, F_N(x_1(\xi), \ldots, x_N(\xi), \xi) \Big). \quad (3.1)$$

Specializing the stochastic variational inequality problem (1.15) in extensive form with its decomposition feature (1.16) in the context of the complementarity in (1.17), we arrive at the following statement of a *stochastic complementarity problem in extensive form*:

find $x(\cdot) = (x_1(\cdot), \ldots, x_N(\cdot)) \in \mathcal{N}$ and $w(\cdot) = (w_1(\cdot), \ldots, w_N(\cdot)) \in \mathcal{M}$
such that $0 \leq x_k(\xi) \perp F_k(x_1(\xi), \ldots, x_N(\xi), \xi) + w_k(\xi) \geq 0, \ \forall \xi, \ \forall k.$ $\quad (3.2)$

Because the set $C(\xi)$ is always IR^n_+ here, independent of ξ, this problem can also be stated equivalently in a *reduced form* in which the elements $w_k(\xi)$ don't appear:

determine $(x_1, x_2(\xi_1), x_2(\xi_1, \xi_2), \ldots, x_N(\xi_1, \xi_2, \ldots, \xi_N))$ such that, at each stage k,
$0 \leq x_k(\xi_1, \ldots, \xi_{k-1}) \perp E_{\xi \mid \xi_1, \ldots, \xi_{k-1}} \Big[F_k(x_1(\xi), \ldots, x_N(\xi), \xi) \Big] \geq 0, \ \forall(\xi_1, \ldots, \xi_{k-1}),$
$$(3.3)$$

where the expectation is again the conditional expectation used in the definition of the subspace $\mathcal{M}$ in (1.5); see [16, (2.20)]. However, despite the appeal of the reduced version (3.3) for understanding the nature of the problem, the version with elements $w_k(\xi)$ in (3.2) is the platform for finding a solution by progressive hedging.

The progressive hedging algorithm in this setting, like always, has at the start of each iteration some $x^\nu(\cdot) = (x_1^\nu(\cdot), \ldots, x_N^\nu(\cdot)) \in \mathcal{N}$ and $w^\nu(\cdot) = (w_1^\nu(\cdot), \ldots, w_N^\nu(\cdot)) \in \mathcal{M}$. In general it proceeds as in (2.5) with updating based on the projection mappings as described in (2.1) and (2.2), but more can be said about the special subproblems to be solved in each iteration. They are now complementarity problems themselves which can be articulated in terms of the components $\hat{x}_k^\nu(\xi)$ of $\hat{x}^\nu(\xi)$:

$\hat{x}^\nu(\xi) =$ the unique $x(\xi)$ such that $0 \leq x_k(\xi) \perp F_k^\nu(x(\xi), \xi) \geq 0$, where
$F_k^\nu(x(\xi), \xi) = F_k(x(\xi), \xi) + w_k^\nu(\xi) + r[x_k(\xi) - x_k^\nu(\xi)]$ for $k = 1, \ldots, N.$ $\quad (3.4)$

The strong monotonicity of $F_k^\nu(\cdot, \xi)$ induced by the r-term guarantees the existence and uniqueness.

To pursue this further for additional insights, let's look more closely at the case of a *stochastic linear complementarity problem*, where

$$F(x(\xi), \xi) = M(\xi)x(\xi) + b(\xi) \text{ for } M(\xi) \in IR^{n \times n} \text{ and } b(\xi) \in IR^n, \quad (3.5)$$

recalling that monotonicity in this situation corresponds to the positive semidefiniteness of the matrix $M(\xi)$, which is the same as the positive semidefiniteness of its symmetric part, $\frac{1}{2}[M(\xi) + M(\xi)^T]$, where "$T$" stands for the transpose. The subproblems solved in the progressive hedging algorithm to get $\hat{x}^\nu(\xi)$ are then likewise linear complementarity problems, one for each scenario ξ:

$$0 \leq \hat{x}^\nu(\xi) \perp M(\xi)\hat{x}^\nu(\xi) + b(\xi) + w^\nu(\xi) + r[\hat{x}^\nu(\xi) - x^\nu(\xi)] \geq 0. \quad (3.6)$$

In terms of the submatrices $M_{kl}(\xi) \in I\!R^{n_k \times n_l}$ of $M(\xi)$ for $k = 1, \ldots, N$ and $l = 1, \ldots, N$, and the corresponding components $b_k(\xi)$ of $b(\xi)$, the complementarity problem in the extensive form (3.2) in the case of the mapping in (3.5) comes out through linearity as

find $x(\cdot) = (x_1(\cdot), \ldots, x_N(\cdot)) \in \mathcal{N}$ and $w(\cdot) = (w_1(\cdot), \ldots, w_N(\cdot)) \in \mathcal{M}$ such that
$$0 \leq x_k(\xi) \perp M_{k1}(\xi)x_1(\xi) + \cdots + M_{kN}(\xi)x_N(\xi) + b_k(\xi) + w_k(\xi) \geq 0, \ \forall \xi, \forall k.$$
$$(3.7)$$

The iteration subproblems can similarly be broken down for $k = 1, \ldots, N$. Especially interesting, though, is what happens when the progressive hedging algorithm is applied, because special facts about monotone linear complementarity problems observed by Cottle, Pang and Stone [21] can also be brought in.

Theorem 3 (progressive hedging in stochastic linear complementarity) *When applied to a monotone stochastic linear complementarity problem, the progressive hedging algorithm can be executed by minimizing in each iteration a strongly convex linear-quadratic function subject to linear constraints having the special property that the unique Lagrange multiplier vector is the same as the unique vector giving the minimum.*

The sequences $\{x^\nu(\cdot)\}_{\nu=1}^\infty$ and $\{w^\nu(\cdot)\}_{\nu=1}^\infty$ thereby generated in $\mathcal{N}$ and $\mathcal{M}$ are sure to converge to a solution pair $\bar{x}(\cdot)$ and $\bar{w}(\cdot)$, moreover at a linear rate: there will exist $\theta \in (0, 1)$ such that

$$||x^{\nu+1}(\cdot) - \bar{x}(\cdot)||^2 + \frac{1}{r^2}||w^{\nu+1}(\cdot) - \bar{w}(\cdot)||^2$$
$$\leq \theta\left(||x^\nu(\cdot) - \bar{x}(\cdot)||^2 + \frac{1}{r^2}||w^\nu(\cdot) - \bar{w}(\cdot)||^2\right). \quad (3.8)$$

Proof As already seen, the subproblem to be solved in each iteration ν for each scenario ξ to get $\hat{x}^\nu(\xi)$ is the linear complementarity problem (3.6), which can also be written as

$$0 \leq \hat{x}^\nu(\xi) \perp [M(\xi) + rI]\hat{x}^\nu(\xi) + b^\nu(\xi) \geq 0, \text{ where } b^\nu(\xi) = b(\xi) + w^\nu(\xi)$$
$$- rx^\nu(\xi). \quad (3.9)$$

and is sure to be solvable. In terms of the gap function for this complementarity problem, namely the quadratic function

$$q^\nu(x, \xi) = \tfrac{1}{2}x \cdot [M(\xi) + rI]x + b^\nu(\xi) \cdot x \text{ for } x \in I\!R^n, \tag{3.10}$$

which is strongly convex, the set of solutions $\hat{x}^\nu(\xi)$ is

$$\left\{ x \in I\!R_+^n \ \middle| \ [M(\xi) + rI]x + b^\nu(\xi) \geq 0 \text{ and } q^\nu(x, \xi) = 0 \right\}.$$

But $q^\nu(x) \geq 0$ when $x \geq 0$ and $[M(\xi) + rI]x + b^\nu(\xi) \geq 0$. Hence the unique solution for each ξ is

$$\hat{x}^\nu(\xi) = \operatorname*{argmin}_x \left\{ q^\nu(x, \xi) \ \middle| \ x \geq 0, \ [M(\xi) + rI]x + b^\nu(\xi) \geq 0 \right\}. \tag{3.11}$$

Thus the iterations can be executed by solving, for each ξ, a strongly convex quadratic programming problem. This echoes a result presented in [21, Theorem 3.1.2] but with the added feature of strong convexity coming from the positive definiteness of $M(\xi) + rI$. On the basis of the characterization in [21, Lemma 3.1.1] of the relationship here between optimal solution vectors and Lagrange multiplier vectors, this strong convexity makes those vectors be unique and coincide.

The claim about convergence is justified the final assertion of Theorem 2 and the polyhedrality of the orthant $I\!R_+^n$, which makes the graph of the mapping T in the application of the proximal point algorithm behind the progressive hedging algorithm be piecewise polyhedral. □

It's worth emphasizing that the quadratic programming problems to be solved in (3.11) to obtain each $\hat{x}^\nu(\xi)$ depend only on the *symmetric* part of the matrix $M(\xi)$ and have r as a known modulus of strong convexity. Moreover these subproblems differ from iteration to iteration *only by shifts in the $b^\nu(\xi)$ vectors* in (3.9) as $x^\nu(\xi)$ and $w^\nu(\xi)$ pass to $x^{\nu+1}(\xi)$ and $w^{\nu+1}(\xi)$. This can open them up to warm-start implementations.

4 Numerical experiments

To demonstrate how the progressive hedging algorithm performs numerically in solving stochastic variational inequalities, we focus on the two-stage case of linear complementarity, where $\xi = (\xi_1, \ldots, \xi_N)$ is replaced by $\xi = \xi_1$. Then we have

$$M(\xi) = \begin{bmatrix} M_{11}(\xi) & M_{12}(\xi) \\ M_{21}(\xi) & M_{22}(\xi) \end{bmatrix}, \qquad b(\xi) = \begin{bmatrix} b_1(\xi) \\ b_2(\xi) \end{bmatrix}, \tag{4.1}$$

with $M(\xi) + M(\xi)^T$ positive semidefinite, while elements $x(\cdot)$ of $\mathcal{N}$ and $w(\cdot)$ of $\mathcal{M}$ in this context have the form

$$x(\xi) = \begin{bmatrix} x_1 \\ x_2(\xi) \end{bmatrix}, \qquad w(\xi) = \begin{bmatrix} w_1(\xi) \\ 0 \end{bmatrix} \text{ with } E_\xi[w_1(\xi)] = 0. \tag{4.2}$$

As the corresponding instance of (3.7), we are looking for a pair as in (4.2) that satisfies, for every $\xi \in \Xi$, the conditions

$$0 \leq x_1 \perp M_{11}(\xi)x_1 + M_{12}(\xi)x_2(\xi) + b_1(\xi) + w_1(\xi) \geq 0,$$
$$0 \leq x_2(\xi) \perp M_{21}(\xi)x_1 + M_{22}(\xi)x_2(\xi) + b_2(\xi) \geq 0, \tag{4.3}$$

where the first condition is equivalent to

$$0 \leq x_1 \perp E_\xi[M_{11}(\xi)]x_1 + E_\xi[M_{12}(\xi)x_2(\xi)] + E_\xi[b_1(\xi)] \geq 0. \tag{4.4}$$

In the execution of the progressive hedging algorithm in this setting, we start each iteration with some $x^\nu(\cdot) \in \mathcal{N}$ and $w^\nu(\cdot) \in \mathcal{M}$ and determine $\hat{x}^\nu(\xi) \in \mathcal{L}$ (with components $\hat{x}_1^\nu(\xi)$ and $\hat{x}_2^\nu(\xi)$ both depending on ξ) by solving for each $\xi \in \Xi$ the linear complementarity problem

$$0 \leq \hat{x}_1^\nu(\xi) \perp [M_{11}(\xi) + rI]\hat{x}_1^\nu(\xi) + M_{12}(\xi)\hat{x}_2^\nu(\xi) + b_1^\nu(\xi) \geq 0,$$
$$0 \leq \hat{x}_2^\nu(\xi) \perp M_{21}(\xi)\hat{x}_1^\nu(\xi) + [M_{22}(\xi) + rI]\hat{x}_2^\nu(\xi) + b_2^\nu(\xi) \geq 0, \tag{4.5}$$

where

$$b_1^\nu(\xi)(\xi) = b_1(\xi) - rx_1^\nu + w_1^\nu(\xi), \qquad b_2^\nu(\xi) = b_2(\xi) - rx_2^\nu(\xi), \tag{4.6}$$

and the strong monotonicity induced by r guarantees the existence of a unique solution. This corresponds to determining $(\hat{x}_1^\nu(\xi), \hat{x}_2^\nu(\xi))$ by minimizing the quadratic expression

$$\frac{1}{2}\Big(\hat{x}_1^\nu(\xi){\cdot}[M_{11}(\xi) + rI]\hat{x}_1^\nu(\xi) + \hat{x}_1^\nu(\xi){\cdot}[M_{12}(\xi) + M_{21}(\xi)]\hat{x}_2^\nu(\xi)$$
$$+ \hat{x}_2^\nu(\xi){\cdot}[M_{22}(\xi) + rI]\hat{x}_2^\nu(\xi)\Big) + b_1^\nu(\xi){\cdot}\hat{x}_1^\nu(\xi) + b_2^\nu(\xi){\cdot}\hat{x}_2^\nu(\xi), \tag{4.7}$$

subject to the constraints

$$\hat{x}_1^\nu(\xi) \geq 0, \qquad [M_{11}(\xi) + rI]\hat{x}_1^\nu(\xi) + M_{12}(\xi)\hat{x}_2^\nu(\xi) + b_1^\nu(\xi) \geq 0,$$
$$\hat{x}_2^\nu(\xi) \geq 0, \qquad M_{21}(\xi)\hat{x}_1^\nu(\xi) + [M_{22}(\xi) + rI]\hat{x}_2^\nu(\xi) + b_2^\nu(\xi) \geq 0. \tag{4.8}$$

The update is given then by

$$x_1^{\nu+1} = E_\xi[\hat{x}_1^\nu(\xi)], \quad x_2^{\nu+1}(\xi) = \hat{x}_2^\nu(\xi), \quad w_1^{\nu+1}(\xi) = w_1^\nu(\xi) - r[\hat{x}_1^\nu(\xi) - x_1^{\nu+1}]. \tag{4.9}$$

To solve linear complementary problem (4.5) for every ξ, we have several choices of methods. First, we could convert the problem into a quadratic program as just described and use some quadratic programming solver. Second, we could use a general solver for linear complementarity problems such as the PATH solver (Dirkse, Ferris and Munson http://pages.cs.wisc.edu/~ferris/path.html). Third, we could reformulate the problem as an nonsmooth equation and write a special code based on the semismooth Newton method introduced in Qi and Sun [22]. After several rounds of experiments, we finally opted for the third choice based on the following observations.

First, the speed of a quadratic solver depends on the problem type. A quadratic program can be solved fast if the quadratic terms are sparse and if an interior point approach is used; otherwise, it is generally slower than direct solvers for linear complementarity. Since our test problems are all "unstructured and dense" through the random generation scheme utilized in our preliminary efforts here, the quadratic programming approach is inferior to the other approaches. However, there is still hope that the quadratic programming approach could be more efficient if the progressive hedging scheme is used for real large scale problems with sparsity. Second, among the direct solvers for the linear complementarity problem, the Newton-based algorithms are faster and more robust. Here by "robust" we mean the algorithm is less dependent on the choice of parameters. Another motivation for selecting the semismooth Newton method is the similarity of the linear complementarity problems encountered in the progressive hedging process. This similarity can facilitate "warm starts" of the Newton's method. Namely, we can use the solutions $x^\nu(\xi)$ for each scenario in iteration ν to initiate the solution of the subproblems in iteration $\nu + 1$.

It can be readily verified that, in terms of P_+ denoting the nearest-point projection mapping from $I\!R^n$ onto its nonnegative orthant,

$$x \in I\!R^n, \ 0 \le x \perp Mx + b \ge 0 \iff 0 = h(x) := x - P_+(x - Mx - b). \quad (4.10)$$

With m indexing the iterates in solving the equation $h(x) = 0$, our test code uses the formula

$$x^{m+1} = x^m - \theta_m \bar{J} h(x^m)^{-1}(x^m) \quad (4.11)$$

to determine the next iterate for a solution to (4.10), where $\bar{J} h(x^m)$ is a matrix in the generalized Jacobian of h at x^m and θ_m is a step length. We use a Matlab code provided by Dr. Xudong Li of the National University of Singapore for this procedure.

The problem data are randomly generated with the following guidelines, based on the fact that $M(\xi)$ in our framework of monotonicity must be the sum of a symmetric positive semidefinite matrix and an arbitrary antisymetric matrix.

- Set rank $s = \lceil 3(n_1 + n_2)/4 \rceil$ and, for each scenario ξ, generate random positive numbers $a_i(\xi)$ for $i = 1, ..., s$ along with random vectors $v_i(\xi) \in I\!R^{n_1+n_2}$ for $i = 1, \cdots, s$. Obtain from these the symmetric positive semidefinite matrix $M_S(\xi) = \sum_{i=1}^s a_i(\xi)v_i(\xi)v_i(\xi)^T$, where $v_i(\xi)^T$ is the row vector transpose of the column vector $v_i(\xi)$.
- Randomly generate a nonzero antisymmetric matrix $M_A(\xi)$ for each ξ. Then let $M(\xi) = M_A(\xi) + M_S(\xi)$, which is guaranteed to be positive semidefinite but not symmetric.
- Randomly generate $b_1(\xi)$ and $b_2(\xi)$ for each ξ.
- Randomly generate the probabilities $p(\xi) > 0$ of the scenarios ξ.

The dimensions of $x_1(\xi)$ and $x_2(\xi)$ range from 15 to 500 while the number of scenarios ("sn" in the tables) ranges from 5 to 200, which we think would cover the range of quite many problems arising in practice. For each fixed dimension and number of scenarios, we generate 10 random problems and count the average number

Table 1 Numerical results while sn increases (dim = [15,15], fixed r = 1)

sn	Avg-iter	Avg-time (s)
5	60.9	0.1
10	77.1	0.2
20	108.9	0.5
50	148.1	1.5
100	179.9	3.2
150	143.0	3.8
200	193.7	6.9
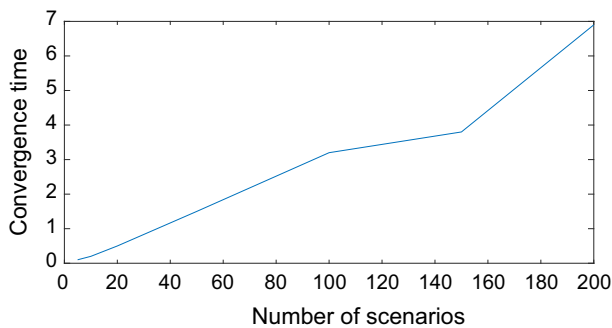
Fig. 1 Convergence time when number of scenarios increases

of iterations ("avg iter" in the tables) and the average CPU time (in seconds, "avg time (s)" in the tables). The stopping criterion we used is tied to the residual of $|h(x^\nu)|$. We set the tolerance to be 10^{-5} and the maximal iterations to be 1000, i.e., if the residual $\leq 10^{-5}$ or the iteration number ≥ 1000, the algorithm stops. The test code is written in Matlab and the experiments are done on a laptop computer running WINDOWS 10.

The numerical testing begins by fixing $r = 1$ and solving problems of size

$$[\text{dimension of } x_1, \ \text{dimension of } x_2(\xi)] = [15, 15].$$

We solve random problems with different numbers of scenarios, ranging from 5 to 200. The numerical results are shown in Table 1 and the trend of convergence time versus the number of scenarios is plotted in Fig. 1. From that we can see that the computation time grows roughly at a linear rate in response to the increase of in the number of scenarios.

We next test the effect of the parameter r on the speed of convergence of the progressive hedging algorithm. We re-solve the same problems by using a heuristic value of $r = \sqrt{n_1 + n_2}$, which yields $r = \sqrt{30}$ for problem size [15,15]. The test results in Table 2, over all the various numbers of scenarios, suggest that this value of r deserves to be explored for its possible advantages more broadly. The trend of time vs. number of scenarios is depicted in Fig. 2.

Table 2 Numerical results while sn increases (dim = [15,15])

sn	r = 1		r = $\sqrt{30}$	
	Avg-iter	Avg-time (s)	Avg-iter	Avg-time (s)
5	60.9	0.1	65.1	0.1
10	77.1	0.2	67.7	0.2
20	108.9	0.5	70.9	0.3
50	148.1	1.5	94.3	1.0
100	179.9	3.2	94.0	1.8
150	143.0	3.8	104.9	2.8
200	193.7	6.9	122.1	4.4
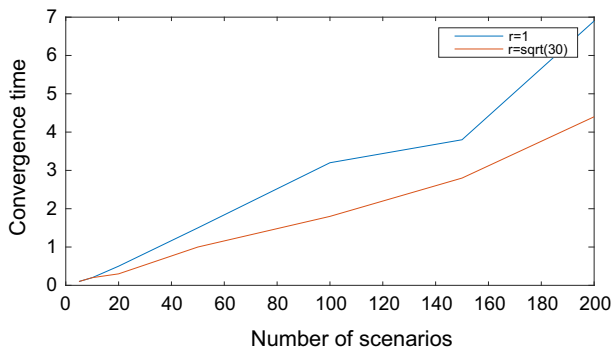
Fig. 2 Convergence time when number of scenarios increases

The third group of experiments is designed to see how efficient the progressive hedging algorithm is in response to the growth in dimension of the decision variables in the stochastic linear complementarity problem, which is an indicator for the capability of this algorithm for solving general two-stage stochastic variational inequalities. The size of the tested problems increases gradually from [15, 15] to [500,500]. Note that the progressive hedging algorithm is reasonably fast—it only takes less than fifteen minutes on average for solving a dense problem of 1000 variables (i.e., $n_1 + n_2 = 1000$) at $r = 1$. The results are in Table 3 and Fig. 3. To check the advantage of our heuristic strategy for r, we compare the results corresponding to different r in Table 4 and Fig. 4. This reveals significant savings in time if $r = \sqrt{n_1 + n_2}$ is used, at least in our limited setting. Further research will be needed for better insights into the choice of r.

In summary, our numerical experiments demonstrate that the progressive hedging algorithm can be used for solving two-stage stochastic linear complementarity problems, which include two-stage linear-quadratic stochastic programming problems and certain two-stage game problems as special cases.

A notable advantage is the "warm start" feature of the progressive hedging algorithm. That is, for each ξ, given $x^\nu(\xi)$, in order to solve the linear complementarity problem (3.9), we solve the nonsmooth equation

$$h^\nu(x) := x - P_+[x - (M(\xi) + rI)x - b^\nu(\xi)] = 0 \qquad (4.12)$$

Table 3 Numerical results while dimension increases (sn = 100, fixed r = 1)

Dimension	Avg-iter	Avg-time (s)
[15, 15]	179.9	3.2
[30, 30]	208.0	4.3
[50, 50]	148.5	4.3
[100, 100]	289.8	12.3
[200, 200]	535.3	49.8
[300, 300]	509.3	118.9
[500, 500]	876.6	709.8
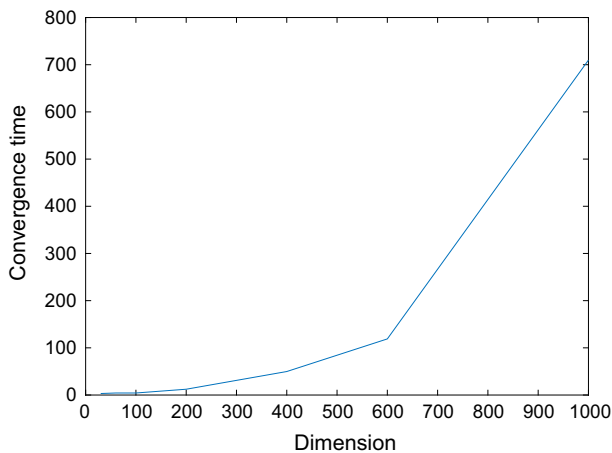
Fig. 3 Convergence time when dimension increases

Table 4 Numerical results while dimension increases (sn = 100)

Dimension	$r = 1$		$r = \sqrt{\dim}$	
	Avg-iter	Avg-time (s)	Avg-iter	Avg-time (s)
[15, 15]	179.9	3.2	94.0	1.7
[30, 30]	208.0	4.3	65.6	1.4
[50, 50]	148.5	4.3	44.3	1.5
[100, 100]	289.8	12.3	29.4	1.6
[200, 200]	535.3	49.8	23.3	3.4
[300, 300]	509.3	118.9	22.2	8.1
[500, 500]	876.6	709.8	22.7	33.1

by a semismooth Newton's method. As is well known, Newton's method is very efficient if the initial point is well chosen. In our implementation, we use $\hat{x}^{\nu}(\xi)$ as the starting point for solving (4.12). Since only $b^{\nu}(\xi)$ is "perturbed" when ν is updated, it is reasonable to use $\hat{x}^{\nu}(\xi)$ as a warm start point to find $\hat{x}^{\nu+1}(\xi)$. The advantage of such a strategy is observed in all of the tested problems. Namely, the solution time
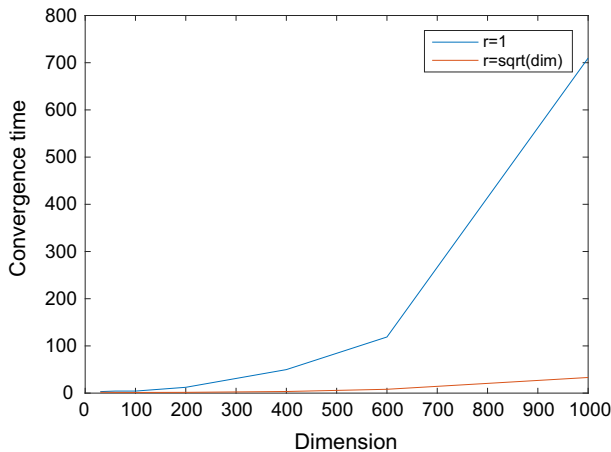
Fig. 4 Convergence time when dimension increases

for $\hat{x}^1(\xi)$, necessarily with a "cold start", was several times slower than the time for $\nu = 2, 3, \ldots$ for every ξ. This feature may be expected to persist when the problems under consideration go from two stages to three or more stages. Computationally, it appears that the progressive hedging algorithm provides a just-right platform for good performance of Newton-type methods.

Of course, the randomly generated problems in these tests are dense and do not necessarily represent the typical problems that would come up in real applications.

References

1. Facchinei, F., Pang, J.-S.: Finite-Dimensional Variational Inequalities and Complementarity Problems. Springer, New York (2003)
2. Chen, X., Fukushima, M.: Expected residual minimization method or stochastic linear complementarity problems. Math. Oper. Res. **30**, 1022–1038 (2005)
3. Chen, X., Zhang, C., Fukushima, M.: Robust solution of monotone stochastic linear complementarity problems. Math. Program. **117**, 51–80 (2009)
4. Chen, X., Pong, T.K., Wets, R.: Two-stage stochastic variational inequalities: an ERM-solution procedure. Math. Program. **165**, 71–111 (2017)
5. Fang, H., Chen, X., Fukushima, M.: Stochastic R_0 matrix linear complementarity problem. SIAM J. Optim. **18**, 482–506 (2007)
6. Jiang, H., Xu, H.: Stochastic approximation approaches to the stochastic variational inequality problem. IEEE Trans. Autom. Control **53**, 1462–1475 (2008)
7. Lu, S.: Confidence regions for stohastic variational inequalities. Math. Oper. Res. **38**, 545–568 (2013)
8. Xu, H.: Sample average approximation methods for a class of variational inequality problems. Asia Pac. J. Oper. Res. **27**, 103–119 (2010)
9. Gürkan, G., Özge, A.Y., Robinson, S.M.: Sample path solution of stochastic variational inequalities. Math. Program. **84**, 313–333 (1999)
10. Iusem, A., Jofré, A., Thompson, P.: Incremental constraint projection methods for monotone stochastic variational inequalities, Math. Oper. Res. (to appear)
11. Iusem, A., Jofré, A., Oliveira, R.I., Thompson, P.: Extragradient method with variance reduction for stochastic variational inequalities. SIAM J. Optim. **27**, 686–724 (2017)
12. Iusem, A., Jofré, A., Oliveira, R.I., Thompson, P.: Variance-based stochastic extragradient methods with line search for stochastic variational inequalies (submitted)

13. Gabriel, S.A., Zhuang, J., Egging, R.: Solving stochastic complementarity problems in energy market modeling using scenario reduction. Eur. J. Oper. Res. **197**, 1028–1040 (2009)
14. Gabriel, S.A., Fuller, J.D.: A Benders decomposition method for solving stochastic complementarity problems with an application in energy. Comput. Econ. **35**, 301–329 (2010)
15. Philpott, A.B., Ferris, M.C., Wets, R.J.-B.: Equilibrium, uncertainty and risk in hydro-thermal electricity systems. Math. Program. B **157**, 483–513 (2016)
16. Rockafellar, R.T., Wets, R.J.-B.: Stochastic variational inequalities: single-stage to multistage. Math. Program. B **165**, 291–330 (2017)
17. Rockafellar, R.T., Wets, R.J.-B.: Scenarios and policy aggregation in optimization under uncertainty. Math. Oper. Res. **16**, 119–147 (1991)
18. Chen, X., Sun, H., Xu, H.: Discrete approximation of two-stage stochastic and distributionally robust linear complementarity problems. arXiv:1796.06847
19. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. **14**, 877–898 (1976)
20. Spingarn, J.E.: Partial inverse of a monotone operator. Appl. Math. Optim. **10**, 247–265 (1983)
21. Cottle, R.W., Pang, J.-S., Stone, R.E.: The Linear Complementarity Problem. Academic Press, Cambridge (1992). (republished by SIAM in 2009)
22. Qi, Liqun, Sun, Jie: A nonsmooth version of Newton's method. Math. Program. **58**, 353–367 (1993)