Citation
Sun, J. and Zhang, M. 2021. The Elicited Progressive Decoupling Algorithm: A Note on the Rate of Convergence and a Preliminary Numerical Experiment on the Choice of Parameters. Set-Valued and Variational Analysis. 29 (4): pp. 997-1018. http://doi.org/10.1007/s11228-021-00613-0

The Elicited Progressive Decoupling Algorithm: A Note on the Rate of Convergence and a Preliminary Numerical Experiment on the Choice of Parameters

Jie Sun*and Min Zhang†

Abstract The paper studies the progressive decoupling algorithm (PDA) of Rockafellar and focuses on the elicited version of the algorithm. Based on a generalized Yosida-regularization of Spingarn's partial inverse of an elicitable operator, it is shown that the elicited progressive decoupling algorithm (EPDA), in a particular nonmonotone setting, linearly converges at a rate that could be viewed as the rate of a rescaled PDA, which may provide certain guidance to the selection of the parameters in computational practice. A preliminary numerical experiment shows that the choice of the elicitation constant has an impact on the efficiency of the EPDA. It is also observed that the influence of the elicitation constant is generally weaker than the proximal constant in the algorithm.

Key words Proximal point algorithm, progressive decoupling algorithm, stochastic variational inequality
MSC classification 90C15, 90C30, 90C33, 91B51

1 Introduction

Rockafellar recently introduced the notion of elicitable monotonicity [9] in an attempt to extend the proximal point algorithm (PPA for short) and its varieties from monotone mappings to certain nonmonotone mappings. Roughly speaking, for a given subspace $\mathcal{M}$ in a Hilbert space $\mathcal{H}$, a set-valued mapping $T : \mathcal{H} \rightrightarrows \mathcal{H}$ is e-elicitable monotone at level $e \geq 0$ in a neighborhood if $T + eP_{\mathcal{M}}$ is monotone in that neighborhood, where $P_{\mathcal{M}}$ is the projection mapping on $\mathcal{M}$.

Our main interest here is a variety of the PPA, called the progressive decoupling algorithm (PDA for short), for the linkage problem. A prototype of the PDA was first developed by Rockafellar and Wets [12] for stochastic programming under the name of progressive hedging algorithm (PHA for short) and later was transplanted to stochastic variational inequality (SVI for short) and Lagrangian variational inequality by Rockafellar and Sun [10, 11]. PHA has been numerically tested for various SVI problems, e.g.,

*Department of Analytics and Operations, School of Business, National University of Singapore, Singapore 119245 and School of EECMS, Curtin University, Australia 6102. Email: jsun@nus.edu.sg

†Corresponding author. Xinjiang Institute of Ecology and Geography, Chinese Academy of Sciences, Urumqi 830011, China and University of Chinese Academy of Sciences, Beijing 100049, China. Email: zhangmin1206@ms.xjb.ac.cn

[2, 14, 15] and has been shown promising to become a major tool in solving SVIs of ordinary-size (i.e., hundreds of variables and scenarios).

In particular, the convergence rate of the PDA has been shown to be globally linear for monotone linear SVI [8]. The rate of convergence depends on a proximal constant r that is inherited from the analysis of the PPA [7]. An elicited version of PDA has been developed both in global and local senses for e-elicitable monotone variational inequality and optimization problems [9]. Specifically, linear rate of convergence has been established for strongly local and global e-elicitable monotone problems. Since this linear rate of convergence is related to both r and e in the elicitable case, it is conceivable that the performance of the elicited progressive decoupling algorithm (EPDA for short) will depend on the joint selection of them, rather than the individual selection of r or e only. Therefore, some sort of theoretical guidance on the choice of the (r, e) pair would be practically helpful, which motivates this paper.

The main contribution of this paper is two-fold.

- Theoretically, we provide an interpretation of the EPDA as the iteration scheme

$$z^{k+1} = (I + cA(T + eP_{\mathcal{M}})_{\mathcal{N}}A)^{-1}(z^k),$$

  where $c = \frac{\sqrt{4r^2+e^2}-e}{2r^2}$, $A : u \mapsto P_{\mathcal{N}}(u) + rP_{\mathcal{M}}(u)$ is a self-adjoint linear operator and $\mathcal{N}$ is the orthogonal complement space of $\mathcal{M}$.[1] From this rescaling interpretation, we conclude that if the elicitable monotonicity is strong with modulus $\sigma > 0$, then the EPDA will converge at rate $\frac{c^{-1}}{c^{-1}+\sigma}$ with respect to the r-norm (defined in Section 2). A bigger e leads to a smaller c, which makes the rate closer to 1 and the speed of convergence would therefore tend to decrease. In addition, we show that the mapping $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ can be regarded as a generalized Yosida-regularization of operator $T_{\mathcal{N}}$, which sheds some light for understanding the connection between the EPDA and Pennanen's nonmonotone proximal point algorithm [4].

- Computationally, we design and test a sequence of numerical problems of both monotone and nonmonotone but elicitable monotone types. The numerical results confirm our theoretical analysis. Specifically, it is observed that, as expected, the elicited version of PDA is generally slower than the non-elicited version in both number of iterations and computational time if they are used to blindly solve the same problem, either monotone or elicitable monotone. Furthermore, the elicited version becomes much slower if the elicitation constant e is set too large. It is also observed that the influence of e is generally weaker than the influence of r in the algorithm.

The rest of this paper is organized as follows. Section 2 provides some background materials on PPA and PDA. Section 3 investigates properties of $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ and provides the aforementioned rescaling interpretation of the EPDA. Section 4 is devoted to presentation of the numerical results and Section 5 concludes this paper.

[1] For any operator $S : \mathcal{H} \rightrightarrows \mathcal{H}$, $S_{\mathcal{N}}$ stands for Spingarn's partial inverse of S [6] (detailed definition will be given in next section). Therefore, $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ means the partial inverse of $T + eP_{\mathcal{M}}$.

2 Preliminaries

We first provide definitions about several types of monotonicity. Then we briefly introduce the PPA for finding a root of a maximal monotone mapping and a nonmonotone mapping, respectively. After that, we describe the PDA and EPDA with corresponding convergence results.

Definition 2.1 *Let $T : \mathcal{H} \rightrightarrows \mathcal{H}$ be a set-valued mapping. The mapping T is called*

- ***monotone**, if*

$$\langle u - u', v - v' \rangle \geq 0 \quad \forall\ v \in T(u),\ v' \in T(u').$$

  *Moreover, a monotone mapping T is **maximal** if, in addition,*

$$\nexists \text{ monotone } T' : \mathcal{H} \rightrightarrows \mathcal{H} \text{ with } \operatorname{grh} T \subset \operatorname{gph} T' \text{ but } \operatorname{gph} T \neq \operatorname{gph} T',$$

  where $gphT := \{(u, v) | v \in T(u)\}$;

- ***strongly monotone with modulus** $\sigma > 0$, if*

$$\langle u - u', v - v' \rangle \geq \sigma \|u - u'\|^2 \quad \forall\ v \in T(u),\ v' \in T(u');$$

- ***globally (or locally around** $(\bar{u}, \bar{v})$**) e-elicitable monotone at level** $e \geq 0$, if $T + eP_{\mathcal{M}}$ is monotone over $\mathcal{H}$ (or locally around $(\bar{u}, \bar{v})$), where $P_{\mathcal{M}}$ is the projection mapping to a certain given subspace $\mathcal{M}$ of $\mathcal{H}$.*

Minty [3] discovered a useful property of a maximal monotone mapping T: For every $c > 0$ and $z \in \mathcal{H}$, there exists a unique $u \in \mathcal{H}$ such that $z - u \in cT(u)$. Equivalently, $J_{cT} := (I + cT)^{-1}$ is a single-valued function, which is called the resolvent of mapping T. It implies that finding a zero point of T is equivalent to finding a fixed point of its resolvent J_{cT}. Since J_{cT} is nonexpansive when T is maximal monotone, one can in principle use the following scheme to obtain a solution to $0 \in T(z)$:

$$z^{k+1} = J_{c_k T}(z^k) := (I + c_k T)^{-1}(z^k), \tag{1}$$

where the sequence $\{c_k\}$ consists of positive scalars satisfying $\inf c_k > 0$, which is the basic scheme of PPA. Moreover, Eckstein and Bertsekas [1] developed a relaxed PPA for maximal monotone mapping T with the following scheme and established its convergence.

$$z^{k+1} = \gamma_k J_{c_k T}(z^k) + (1 - \gamma_k) z^k, \tag{2}$$

$$\text{with } 0 < \inf \gamma_k \leq \sup \gamma_k < 2 \text{ and } \inf c_k > 0.$$

Although these results are based on monotonicity of the underlying mapping, it is shown by Pennanen [4] that they eventually lead to some results for the nonmonotone case. When T is no longer maximal monotone but its Yosida-regularization $T_e = (T^{-1} + eI)^{-1}$ is maximal monotone for some $e > 0$, Pennanen demonstrated that the sequence generated by PPA (1) still converges if the parameter c_k is well chosen. In fact, due to the equivalence between problem $0 \in T(z)$ and $0 \in T_e(z)$, the key idea of Pennanen's

work is to apply the relaxed PPA (2) to the maximal monotone T_e instead of the original (possibly nonmonotone) operator T, i.e.

$$z^{k+1} = \gamma_k J_{c'_k T_e}(z^k) + (1-\gamma_k) z^k,$$

$$\text{where } 0 < \inf \gamma_k \leq \sup \gamma_k < 2 \text{ and } \inf c'_k > 0.$$

Based on the following relationship between the resolvents of T_e and T:

$$(I + cT_e)^{-1} = \frac{e}{c+e} I + \frac{c}{c+e} \left[I + (c+e)T \right]^{-1}, \tag{3}$$

the relaxed PPA (3) on T_e turns out to be a rescaled PPA of form (1) on T by choosing

$$c'_k = c_k - e, \ \gamma_k = \frac{c_k}{c_k - e}.$$

Following the convergence results of the relaxed PPA for maximal monotone mappings, Pennanen proved that the choice of c_k satisfying $\inf c_k > 2e$ can guarantee the convergence of PPA (1) for such nonmonotone T.

An important application of the PPA is the PDA [8, 9] in solving the so-called linkage problem. The notion of linkage problem is popularized by Rockafellar [8, 9]. Let $\mathcal{H}$ be a Hilbert space and consider a set-valued mapping $T : \mathcal{H} \rightrightarrows \mathcal{H}$. Let $\mathcal{N}$ be a subspace of $\mathcal{H}$ and let $\mathcal{M}$ be its orthogonal complement. The linkage problem for $\mathcal{N}$ and T is to

$$\text{find } x \in \mathcal{N} \text{ and } w \in \mathcal{M} \text{ such that } w \in T(x), \tag{4}$$

where the condition $x \in \mathcal{N}$ represents a "linkage" constraint. The iteration procedure of the PDA is, in a nutshell, as follows.

Algorithm 1. The Progressive Decoupling Algorithm

Having $x^k \in \mathcal{N}$, $w^k \in \mathcal{M}$ and $r > 0$,

Step 1. obtain $\hat{x}^k$ via solving $w^k - r(x - x^k) \in T(x)$,

Step 2. update $x^{k+1} = P_{\mathcal{N}}(\hat{x}^k)$, $w^{k+1} = w^k - r(\hat{x}^k - x^{k+1})$.

$k := k + 1$. **repeat**.

When $x = (x_1, x_2, \cdots, x_K) \in \mathcal{H}_1 \times \mathcal{H}_2 \times \cdots \times \mathcal{H}_K$ and $T(x) = (T_1(x_1), \cdots, T_K(x_K))$, such as in SVIs [10, 11, 13], Step 1 of Algorithm 1 can be decomposed in term of scenarios, namely, obtaing $\hat{x}_i^k$ by separately solving

$$w_i^k - r(x_i - x_i^k) \in T_i(x_i) \quad \forall \ i = 1, \cdots, K,$$

where w_i^k is the component of w_k corresponding to x_i. That is why it is called decoupling.

In the following, we illustrate the connection between the PDA and the PPA. We start with the case of T being monotone. Based on Spingarn's partial inverse mapping $T_{\mathcal{N}}$ [6], which is defined by

$$\text{gph}\, T_{\mathcal{N}} := \{(P_{\mathcal{N}}(u) + P_{\mathcal{M}}(v), P_{\mathcal{M}}(u) + P_{\mathcal{N}}(v)) \mid (u, v) \in \text{gph}\, T\}, \tag{5}$$

one can equivalently reformulate the linkage problem (4) as a generalized equation problem, namely,

$$\text{find } z \text{ such that } 0 \in T_{\mathcal{N}}(z), \text{ then set } x = P_{\mathcal{N}}(z),\ w = P_{\mathcal{M}}(z). \tag{6}$$

Problem (6) is equivalent to

$$\text{find } z \text{ such that } 0 \in AT_{\mathcal{N}}A(z), \text{ then set } x = P_{\mathcal{N}}(Az),\ w = P_{\mathcal{M}}(Az), \tag{7}$$

where A is the invertible linear mapping $A : z \mapsto P_{\mathcal{N}}(z) + rP_{\mathcal{M}}(z)$. Since $T_{\mathcal{N}}$, as well as $AT_{\mathcal{N}}A$, is maximal monotone if and only if T is maximal monotone, the PDA can be interpreted as a special version of PPA applied to the mapping $AT_{\mathcal{N}}A$. More concretely, Algorithm 1 is equivalent to the following scheme

$$z^{k+1} = (I + r^{-1}AT_{\mathcal{N}}A)^{-1}(z^k), \text{ with } x^{k+1} = P_{\mathcal{N}}(Az^{k+1}),\ w^{k+1} = P_{\mathcal{M}}(Az^{k+1}).$$

See a detailed proof in [10]. Following the convergence results of PPA [7], the above iteration scheme generates a convergent sequence to a solution of problem (4), as long as (4) has a solution. Moreover, if in addition the graph of T is the union of a finite collection of polyhedral convex sets, the rate of convergence is linear with respect to certain norm of z, i.e., $\|A^{-1}(x + w)\|$, which is the so-called r-norm of pair (x, w):

$$\|(x, w)\|_r := (\|x\|^2 + r^{-2}\|w\|^2)^{1/2}.$$

Recently, Rockafellar [9] developed a theoretical framework of elicitable monotonicity to go beyond the assumption of monotonicity and designed the EPDA to handle problem (4), where the mapping T is possibly nonmonotone, but elicitable monotone. The iteration scheme is as follows.

Algorithm 2. The Elicited Progressive Decoupling Algorithm

Having $x^k \in \mathcal{N}$, $w^k \in \mathcal{M}$ and $r > e \geq 0$,

Step 1. obtain $\hat{x}^k$ via solving $w^k - r(x - x^k) \in T(x)$,

Step 2. update $x^{k+1} = P_{\mathcal{N}}(\hat{x}^k)$, $w^{k+1} = w^k - (r - e)(\hat{x}^k - x^{k+1})$.

$k := k + 1$. **repeat**.

Similar to the monotone case, Rockafellar proved that for an elicitable monotone T, as long as a solution to the linkage problem (4) exists, Algorithm 2 generates a convergent sequence of pairs (x^k, w^k) to a solution (x^*, w^*) of the linkage problem in the manner that the so-called (r, e)-norm of the pair (x, w)

$$\|(x, w)\|_{r,e} := \left(\|x\|^2 + \frac{1}{r(r - e)}\|w\|^2\right)^{1/2}$$

keeps decreasing. Furthermore, if $T + eP_{\mathcal{N}}$ is strongly monotone with modulus $\sigma > 0$, then the convergence will follow the pattern that

$$\|x^{k+1} - x^*\| \leq \|(x^{k+1}, w^{k+1}) - (x^*, w^k)\|_{r,e} \leq \frac{r}{r + \sigma}\|(x^k, w^k) - (x^*, w^*)\|_{r,e}. \tag{8}$$

Since both r and e show up in the convergence rate estimate of (8) with an implicit requirement of $r > e$, it appears complicated to theoretically compare the convergence speed for different (r, e). However, by following the idea in the proof of Theorem 1(c) in [9], we can identify the rate of convergence of EPDA with the rate of a rescaled PDA under the r-norm, which may provide a clearer guidance for selecting ideal e and r in practice. We start this job with the following theorem.

Theorem 2.1 *Suppose that the linkage problem* (4) *is solvable and the set-valued mapping $T : \mathcal{H} \rightrightarrows \mathcal{H}$ is maximal strongly monotone of modulus $\sigma > 0$. Let $T_{\mathcal{N}}$ be the partial inverse of T and A be the linear mapping defined as $Az = x + w$ where $x = P_{\mathcal{N}}(z)$ and $w = rP_{\mathcal{M}}(z)$ (r is a positive constant). Let $0 < c \leq r^{-1}$. Then the sequences $\{(x^k, w^k)\}$ and $\{z^k\}$ generated by the procedure*

$$\begin{aligned} z^{k+1} &= (I + cAT_{\mathcal{N}}A)^{-1}(z^k), \text{ and} \\ x^{k+1} &= P_{\mathcal{N}}(Az^{k+1}),\ w^{k+1} = P_{\mathcal{M}}(Az^{k+1}), \end{aligned} \tag{9}$$

will converge to a solution (x^*, w^*) of Problem (4)*, and $z^* = x^* + r^{-1}w^*$, respectively, in the following pattern.*

$$\|x^{k+1} - x^*\| \leq \|(x^{k+1}, w^{k+1}) - (x^*, w^k)\|_r \leq \frac{c^{-1}}{c^{-1} + \sigma}\|(x^k, w^k) - (x^*, w^*)\|_r. \tag{10}$$

Proof. Following exactly the same argument as Theorem 1 in [10] (or cf. [9, Theorem 1(c)] for a coincise proof) , we can prove $(x^k, w^k) \to (x^*, w^*)$, a solution of (4). Thus, we only have to show $z^* = x^* + r^{-1}w^*$ and (10). Since $Az^* = x^* + w^*$ and $A^{-1}(z) = P_{\mathcal{N}}(z) + r^{-1}P_{\mathcal{M}}(z) = x + r^{-1}w$, we have $z^* = A^{-1}(x^* + w^*) = x^* + r^{-1}w^*$. We next prove (10). The first inequality in (10) is obvious due to the definition of the r-norm. For the second inequality of (10), note that the first formula in (9) means that

$$c^{-1}(z^k - z^{k+1}) \in AT_{\mathcal{N}}A(z^{k+1}),$$

which is equivalent to

$$\begin{aligned} & c^{-1}A^{-2}(Az^k - Az^{k+1}) \in T_{\mathcal{N}}(Az^{k+1}) \\ \Leftrightarrow\ & c^{-1}A^{-2}((x^k - x^{k+1}) + (w^k - w^{k+1})) \in T_{\mathcal{N}}(Az^{k+1}). \end{aligned} \tag{11}$$

The definition of A implies that $A^{-2}(z) = P_{\mathcal{N}}(z) + r^{-2}P_{\mathcal{M}}(z)$, therefore (11) becomes

$$c^{-1}(x^k - x^{k+1} + r^{-2}(w^k - w^{k+1})) \in T_{\mathcal{N}}(x^{k+1} + w^{k+1}). \tag{12}$$

Let $y^{k+1} = c^{-1}r^{-2}(w^k - w^{k+1})$. From the definition of $T_{\mathcal{N}}$ (set $u = x^{k+1} + y^{k+1}$ and $v = c^{-1}(x^k - x^{k+1}) + w^{k+1}$ in (5)), we see that (12) is equivalent to

$$c^{-1}(x^k - x^{k+1}) + w^{k+1} \in T(x^{k+1} + y^{k+1}).$$

Since (x^*, w^*) is a solution to problem (4), it holds $w^* \in T(x^*)$. Based on strong monotonicity of T, we obtain

$$\langle x^{k+1} + y^{k+1} - x^*, c^{-1}(x^k - x^{k+1}) + w^{k+1} - w^* \rangle \geq \sigma\|x^{k+1} + y^{k+1} - x^*\|^2. \tag{13}$$

The left-hand side of (13) is equal to

$$\begin{aligned}
& c^{-1}\langle x^{k+1}-x^*, x^k-x^{k+1}\rangle + \langle y^{k+1}, w^{k+1}-w^*\rangle \\
= \;& -c^{-1}\|x^{k+1}-x^*\|^2 + c^{-1}\langle x^{k+1}-x^*, x^k-x^*\rangle - cr^2\|y^{k+1}\|^2 + \langle y^{k+1}, w^k-w^*\rangle \\
= \;& c^{-1}\langle x^{k+1}-x^*, x^k-x^*\rangle + c^{-1}\langle cry^{k+1}, r^{-1}(w^k-w^*)\rangle \\
& -c^{-1}\|x^{k+1}-x^*\|^2 - cr^2\|y^{k+1}\|^2 \\
= \;& c^{-1}\langle x^{k+1}-x^*+cry^{k+1}, x^k-x^*+r^{-1}w^k-r^{-1}w^*\rangle \\
& -c^{-1}\|x^{k+1}-x^*\|^2 - cr^2\|y^{k+1}\|^2, \qquad (14)
\end{aligned}$$

while the right-hand side of (13) is equal to

$$\sigma\|x^{k+1}-x^*\|^2 + \sigma\|y^{k+1}\|^2. \qquad (15)$$

Combining (14) and (15), we have

$$\begin{aligned}
& (c^{-1}+\sigma)\left\|x^{k+1}-x^*+\sqrt{\frac{\sigma+cr^2}{\sigma+c^{-1}}}y^{k+1}\right\|^2 \\
\le \;& c^{-1}\langle x^{k+1}-x^*+cry^{k+1}, x^k-x^*+r^{-1}w^k-r^{-1}w^*\rangle \\
\le \;& c^{-1}\|x^{k+1}-x^*+cry^{k+1}\|\,\|x^k-x^*+r^{-1}w^k-r^{-1}w^*\|, \qquad (16)
\end{aligned}$$

where the second inequality follows from the Cauchy-Schwartz inequality. Since $\sigma > 0$ and $0 < cr < 1$, it holds $\sqrt{\frac{\sigma+cr^2}{\sigma+c^{-1}}} \ge cr$. Hence we have

$$\left\|x^{k+1}-x^*+\sqrt{\frac{\sigma+cr^2}{\sigma+c^{-1}}}y^{k+1}\right\|^2 \ge \left\|x^{k+1}-x^*+cr\,y^{k+1}\right\|^2.$$

Therefore, (16) implies that

$$\|x^{k+1}-x^*+cry^{k+1}\| \le \frac{c^{-1}}{c^{-1}+\sigma}\|x^k-x^*+r^{-1}w^k-r^{-1}w^*\|,$$

which is equivalent to

$$\|(x^{k+1},w^{k+1})-(x^*,w^k)\|_r \le \frac{c^{-1}}{c^{-1}+\sigma}\|(x^k,w^k)-(x^*,w^*)\|_r.$$

The proof is complete. □

To summarize, our target is to interpret Algorithm 2 as in the format (9) with appropriate definitions of T and r, then apply Theorem 2.1 in order to provide a clearer explanation on how the convergence rate of the EPDA depends on (r, e). Moreover, since PDA has a very close connection to PPA in the monotone case, one may naturally wonder whether the EPDA has some connection with the PPA in the nonmonotone case as well, especially with the work of Pennanen on nonmonotone PPA [4]. This question is also mentioned in Rockafellar's paper [9]. Motivated by these considerations, we attempt to present a rescaling interpretation of the EPDA through Theorem 2.1. The analysis in the next section also reveals that mapping $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ can be viewed as a generalized Yosida-regularization of $T_{\mathcal{N}}$, which provides a clue on on the connection between PDA and Pennanen's nonmonotone PPA. It also explains why the Yosida-regularization T_e is not a suitable tool for transplanting PDA into the nonmonotone setting and why one needs instead go along the line of elicited monotonicity by utilizing mapping $(T+eP_{\mathcal{M}})_{\mathcal{N}}$.

3 A Rescaling Interpretation of the EPDA

We first derive a relation between $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ and $T_{\mathcal{N}}$, then establish the connection between their resolvents. Based on such connections, we present a rescaling interpretation for the EPDA in the end of this section.

The next proposition shows the relationship between Spingarn's partial inverses of operators $T + eP_{\mathcal{M}}$ and T.

Proposition 3.1 *Let $T : \mathcal{H} \rightrightarrows \mathcal{H}$ be a set-valued mapping, $e \geq 0$ and $T_{\mathcal{N}}$ be Spingarn's partial inverse of T. Then,*

$$(T + eP_{\mathcal{M}})_{\mathcal{N}} = (T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}})^{-1}. \tag{17}$$

Moreover, $T + eP_{\mathcal{M}}$ is maximal monotone iff $T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}}$ is maximal monotone.

Proof.

$$\begin{aligned} v \in (T + eP_{\mathcal{M}})_{\mathcal{N}}(u) &\Leftrightarrow P_{\mathcal{N}}(v) + P_{\mathcal{M}}(u) \in (T + eP_{\mathcal{M}})(P_{\mathcal{N}}(u) + P_{\mathcal{M}}(v)) \\ &\Leftrightarrow P_{\mathcal{N}}(v) + P_{\mathcal{M}}(u) - eP_{\mathcal{M}}(v) \in T(P_{\mathcal{N}}(u) + P_{\mathcal{M}}(v)) \\ &\Leftrightarrow v \in T_{\mathcal{N}}(u - eP_{\mathcal{M}}(v)) \\ &\Leftrightarrow u \in T_{\mathcal{N}}^{-1}(v) + eP_{\mathcal{M}}(v) \\ &\Leftrightarrow v \in (T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}})^{-1}(u). \end{aligned}$$

Since $T_{\mathcal{N}}$ is maximal monotone if and only if T is maximal monotone, we have

$$\begin{aligned} &T + eP_{\mathcal{M}} \text{ is maximal monotone } \Leftrightarrow \ (T + eP_{\mathcal{M}})_{\mathcal{N}} \text{ is maximal monotone } \Leftrightarrow \\ &(T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}})^{-1} \text{ is maximal monotone } \Leftrightarrow \ T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}} \text{ is maximal monotone.} \end{aligned}$$

□

Remark 3.1

1. *Recall that the Yosida-regularization of mapping T is $T_e = (T^{-1} + eI)^{-1}$. Notice that the equation* (17) *discloses a similar connection between $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ and $T_{\mathcal{N}}$, which implies that $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ may be viewed as something similar to the Yosida-regularization of $T_{\mathcal{N}}$;*

2. *Recall that*

$$0 \in T(z) \ \Leftrightarrow \ 0 \in T_e(z),$$

where T is possibly nonmonotone but T_e is monotone for some $e > 0$. The adoption of T_e enables to find a root of nonmonotone T by applying the relaxed PPA on the monotone T_e. Back to the linkage problem (4)*, since*

$$0 \in T_{\mathcal{N}}(z) \ \Leftrightarrow \ 0 \in (T + eP_{\mathcal{M}})_{\mathcal{N}}(z),$$

where T is possibly nonmonotone but T is e-elicitable monotone for some $e > 0$. Thus, the adoption of $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ plays a similar role on the transformation of PDA to the e-elicitable monotone case.

3. *Specifically, When $\mathcal{N} = \{0\}$, we have $T_{\mathcal{N}} = T^{-1}$. Proposition 3.1 shows that $(T + eP_{\mathcal{M}})_{\mathcal{N}} = (T^{-1})_e$. In this special case, $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ is exactly the Yosida-regularization of $T_{\mathcal{N}}$.*

In [4], Pennanen explored the relation between the resolvents of T and its Yosida-regularization T_e and derived a nonmonotone PPA from it. Hence it is natural to investigate if there exists any similar relation between the resolvents of $T_{\mathcal{N}}$ and $(T + eP_{\mathcal{M}})_{\mathcal{N}}$. We next establish a connection between the resolvents of $DT_{\mathcal{N}}D$ and $D(T + eP_{\mathcal{M}})_{\mathcal{N}}D$, where an invertible linear mapping D, which is an important tool for the analysis on the convergence rate of the EPDA.

Proposition 3.2 *Let D be defined as $D : u \mapsto \alpha P_{\mathcal{N}}(u) + \beta P_{\mathcal{M}}(u)$ with positive α and β and let $T : \mathcal{H} \rightrightarrows \mathcal{H}$ be a maximal elicitable monotone mapping at level e. Then we have*

$$(I + cD(T + eP_{\mathcal{M}})_{\mathcal{N}}D)^{-1} = (I - R)I + R(I + cR^{-1}DT_{\mathcal{N}}D)^{-1}, \tag{18}$$

where $c > 0$ and $R = (I + \frac{e}{c}D^{-1}P_{\mathcal{M}}D^{-1})^{-1}$. Moreover, one has

$$R(u) = P_{\mathcal{N}}(u) + \frac{c\beta^2}{c\beta^2 + e}P_{\mathcal{M}}(u), \tag{19}$$

which infers that R is a single-valued and invertible linear mapping.

Proof. Based on Proposition 3.1, we have $(T + eP_{\mathcal{M}})_{\mathcal{N}} = (T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}})^{-1}$. Therefore,

$$\begin{aligned}
& v = (I + cD(T + eP_{\mathcal{M}})_{\mathcal{N}}D)^{-1}(u) \\
\Leftrightarrow\ & D^{-1}\left(\frac{u - v}{c}\right) \in (T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}})^{-1}(Dv) \\
\Leftrightarrow\ & Dv \in (T_{\mathcal{N}}^{-1} + eP_{\mathcal{M}})\left(D^{-1}\left(\frac{u - v}{c}\right)\right) \\
\Leftrightarrow\ & Dv - \frac{e}{c}P_{\mathcal{M}}D^{-1}(u - v) \in T_{\mathcal{N}}^{-1}\left(D^{-1}\left(\frac{u - v}{c}\right)\right) \\
\Leftrightarrow\ & D^{-1}\left(\frac{u - v}{c}\right) \in T_{\mathcal{N}}D\left(v - \frac{e}{c}D^{-1}P_{\mathcal{M}}D^{-1}(u - v)\right).
\end{aligned} \tag{20}$$

From the definition of D and $P_{\mathcal{M}}$, we have $D^{-1}P_{\mathcal{M}}D^{-1}(u) = \beta^{-2}P_{\mathcal{M}}(u)$, which leads to

$$\left(I + \frac{e}{c}D^{-1}P_{\mathcal{M}}D^{-1}\right)(u) = P_{\mathcal{N}}(u) + \frac{c\beta^2 + e}{c\beta^2}P_{\mathcal{M}}(u).$$

Thus, given the definition of R as $R = (I + \frac{e}{c}D^{-1}P_{\mathcal{M}}D^{-1})^{-1}$, (19) holds, which infers that R is a single-valued invertible linear mapping. Therefore, (20) is equivalent to

$$\begin{aligned}
& u - v \in cDT_{\mathcal{N}}D\big(R^{-1}v - (R^{-1} - I)u\big) \\
\Leftrightarrow\ & R(u - (R^{-1}v - (R^{-1} - I)u)) \in cDT_{\mathcal{N}}D\left(R^{-1}v - (R^{-1} - I)u\right) \\
\Leftrightarrow\ & u - (R^{-1}v - (R^{-1} - I)u) \in cR^{-1}DT_{\mathcal{N}}D\left(R^{-1}v - (R^{-1} - I)u\right) \\
\Leftrightarrow\ & u \in (I + cR^{-1}DT_{\mathcal{N}}D)(R^{-1}v - (R^{-1} - I)u) \\
\Leftrightarrow\ & R^{-1}v - (R^{-1} - I)u = (I + cR^{-1}DT_{\mathcal{N}}D)^{-1}(u) \\
\Leftrightarrow\ & v = \Big[(I - R) + R(I + cR^{-1}DT_{\mathcal{N}}D)^{-1}\Big](u).
\end{aligned}$$

□

Remark 3.2 *When $D = I$, we have $R = \frac{c}{c+e} I$. Then, equation* (18) *becomes*

$$\left(I + c(T + eP_{\mathcal{M}})_{\mathcal{N}}\right)^{-1} = \frac{e}{c+e} I + \frac{c}{c+e}(I + (c+e)T_{\mathcal{N}})^{-1}.$$

It reveals a similar relation between $T_{\mathcal{N}}$ and $(T+eP_{\mathcal{M}})_{\mathcal{N}}$ to the relation between mapping T and its Yosida-regularization T_e in Pennanen's Lemma 8 of [4]. This finding provides another evidence for viewing mapping $(T+eP_{\mathcal{M}})_{\mathcal{N}}$ as a generalized Yosida-regularization of the partial inverse mapping $T_{\mathcal{N}}$.

In addition, when $\mathcal{N} = \{0\}$, we have $T_{\mathcal{N}} = T^{-1}$ and $(T + eP_{\mathcal{M}})_{\mathcal{N}} = (T^{-1})_e$, equation (18) *reduces to relation* (3) *for T^{-1}.*

Propositions 3.1 and 3.2 provide a basis for the rescaling interpretation of the EPDA.

Theorem 3.1 *Suppose that set-valued mapping $T : \mathcal{H} \rightrightarrows \mathcal{H}$ is elicitable maximal monotone at level $e \geq 0$. Then Algorithm 2 is equivalent to PPA for mapping $A(T+eP_{\mathcal{M}})_{\mathcal{N}}A$, namely*

$$z^{k+1} = (I + cA(T + eP_{\mathcal{M}})_{\mathcal{N}}A)^{-1}(z^k), \tag{21}$$

where $c = \frac{\sqrt{e^2+4r^2}-e}{2r^2}$ and A is a non-singular linear operator defined as $A : u \mapsto P_{\mathcal{N}}(u) + rP_{\mathcal{M}}(u)$.

Proof. By the definition of elicitable monotonicity, T is maximal elicitable monotone for some $e \geq 0$ iff $(T+eP_{\mathcal{M}})$ is maximal monotone, which leads to the maximal monotonicity of $(T + eP_{\mathcal{M}})_{\mathcal{N}}$. From Proposition 3.2, the iteration scheme (21) is equivalent to

$$z^{k+1} = (I - R)(z^k) + R(I + cR^{-1}AT_{\mathcal{N}}A)^{-1}(z^k), \tag{22}$$

with $R : u \mapsto P_{\mathcal{N}}(u) + \frac{cr^2}{cr^2+e}P_{\mathcal{M}}(u)$. Then, from the iteration scheme (22), we have

$$\begin{aligned}
& R^{-1}\left(z^{k+1} - z^k\right) + z^k = (I + cR^{-1}AT_{\mathcal{N}}A)^{-1}(z^k) \\
\Leftrightarrow\ & z^k \in (I + cR^{-1}AT_{\mathcal{N}}A)\left(R^{-1}\left(z^{k+1} - z^k\right) + z^k\right) \\
\Leftrightarrow\ & R^{-1}\left(z^k - z^{k+1}\right) \in cR^{-1}AT_{\mathcal{N}}A\left(R^{-1}\left(z^{k+1} - z^k\right) + z^k\right) \\
\Leftrightarrow\ & A^{-1}\left(\frac{z^k - z^{k+1}}{c}\right) \in T_{\mathcal{N}}\left(AR^{-1}\left(z^{k+1} - z^k\right) + A(z^k)\right)
\end{aligned} \tag{23}$$

Based on the definitions of linear mapping A and R, we have

$$A^{-1} : u \mapsto P_{\mathcal{N}}(u) + r^{-1}P_{\mathcal{M}}(u) \text{ and } AR^{-1} : u \mapsto P_{\mathcal{N}}(u) + \frac{cr^2+e}{cr}P_{\mathcal{M}}(u).$$

Substituting them into (23), we obtain

$$\begin{aligned}
& \frac{P_{\mathcal{N}}(z^k) - P_{\mathcal{N}}(z^{k+1})}{c} + \frac{P_{\mathcal{M}}(z^k) - P_{\mathcal{M}}(z^{k+1})}{cr} \\
\in\ & T_{\mathcal{N}}\left((P_{\mathcal{N}}(z^{k+1}) - P_{\mathcal{N}}(z^k)) + \frac{cr^2+e}{cr}\left(P_{\mathcal{M}}(z^{k+1}) - P_{\mathcal{M}}(z^k)\right) + P_{\mathcal{N}}(z^k) + rP_{\mathcal{M}}(z^k)\right).
\end{aligned}$$

Based on the definition of partial inverse $T_{\mathcal{N}}$, it follows that

$$\begin{aligned} & \frac{P_{\mathcal{N}}(z^k) - P_{\mathcal{N}}(z^{k+1})}{c} + \frac{cr^2 + e}{cr}\left(P_{\mathcal{M}}(z^{k+1}) - P_{\mathcal{M}}(z^k)\right) + rP_{\mathcal{M}}(z^k) \\ \in \quad & T\left(P_{\mathcal{N}}(z^{k+1}) + \frac{P_{\mathcal{M}}(z^k) - P_{\mathcal{M}}(z^{k+1})}{cr}\right). \end{aligned} \tag{24}$$

Let $\hat{x} = P_{\mathcal{N}}(z^{k+1}) + \frac{P_{\mathcal{M}}(z^k) - P_{\mathcal{M}}(z^{k+1})}{cr}$. Then we have

$$\begin{cases} P_{\mathcal{N}}(\hat{x}) = P_{\mathcal{N}}(z^{k+1}), \\ P_{\mathcal{M}}(\hat{x}) = \frac{P_{\mathcal{M}}(z^k) - P_{\mathcal{M}}(z^{k+1})}{cr}, \end{cases} \Leftrightarrow \begin{cases} P_{\mathcal{N}}(z^{k+1}) = P_{\mathcal{N}}(\hat{x}), \\ P_{\mathcal{M}}(z^{k+1}) = P_{\mathcal{M}}(z^k) - crP_{\mathcal{M}}(\hat{x}). \end{cases} \tag{25}$$

Substituting (25) into (24), we have

$$\frac{P_{\mathcal{N}}(z^k)}{c} - \left(\frac{P_{\mathcal{N}}(\hat{x})}{c} + (cr^2 + e)P_{\mathcal{M}}(\hat{x})\right) + rP_{\mathcal{M}}(z^k) \in T(\hat{x}). \tag{26}$$

Moreover, setting $c = \frac{\sqrt{e^2+4r^2}-e}{2r^2}$, i.e., $c^{-1} = cr^2 + e$, then (26) becomes

$$rP_{\mathcal{M}}(z^k) - c^{-1}(\hat{x} - P_{\mathcal{N}}(z^k)) \in T(\hat{x}). \tag{27}$$

Let $x^k = P_{\mathcal{N}}(z^k)$, $w^k = rP_{\mathcal{M}}(z^k)$. Together with (25), it follows that

$$\begin{cases} x^{k+1} = P_{\mathcal{N}}(z^{k+1}) = P_{\mathcal{N}}(\hat{x}), \\ w^{k+1} = rP_{\mathcal{M}}(z^{k+1}) = r(P_{\mathcal{M}}(z^k) - crP_{\mathcal{M}}(\hat{x})) = w^k - (c^{-1} - e)(\hat{x} - x^{k+1}). \end{cases} \tag{28}$$

Putting (27) and (28) together, the scheme (21) is equivalent to the following iteration scheme.
Having $x^k \in \mathcal{N}$, $w^k \in \mathcal{M}$ and $c^{-1} > e \geq 0$,
Step 1: obtain $\hat{x}$ via solving $w^k - c^{-1}(\hat{x} - x^k) \in T(\hat{x})$,
Step 2: update $x^{k+1} = \hat{x}_{\mathcal{N}}$ and $w^{k+1} = w^k - (c^{-1} - e)(\hat{x} - x^{k+1})$,
which is exactly the EPDA (8). □

Since

$$c = \frac{\sqrt{e^2 + 4r^2} - e}{2r^2} \text{ and } c^{-1} = \frac{\sqrt{e^2 + 4r^2} + e}{2} =: d,$$

based on Theorems 2.1 and 3.1, if $T + eP_{\mathcal{M}}$ is strongly maximal monotone with modulus $\sigma > 0$ for some $e \geq 0$, then convergence pattern (10) holds for algorithm (21).Therefore, we can expect that the rate of convergence

$$\delta := \frac{c^{-1}}{c^{-1} + \sigma} = \frac{d}{d + \sigma}$$

will deteriorate when either parameter e or r grows since δ is an increasing function of e and r. Moreover, it can be observed that δ could be insensitive to the change of (r, e) if r (and therefore e) is large. Hence we will concentrate on relatively small r in our numerical experiments.

4 Numerical Experiments

Since the stochastic linear complementarity problem (SLCP) [10] is an important field of applications of PDA, we design a sequence of experiments on solving two-stage SLCPs and observe the performance of EPDA under different values of parameter r and e. In all statements of the problems and algorithms in this section, the support Ξ of the random vector ξ is finite, which is a common hypothesis in applications of the SLCP.

Given matrices $M_{ij}(\xi), i, j = 1, 2$ and vectors $q_i(\xi)$, the two-stage SLCP aims to find $x(\cdot)$, $y(\cdot)$ and $w(\cdot)$ such that $x(\xi) \equiv x \ \forall \ \xi \in \Xi$ (nonanticipativity), $\mathbb{E}_\xi[w(\xi)] = 0$ (orthogonality) and

$$0 \le \begin{pmatrix} x(\xi) \\ y(\xi) \end{pmatrix} \ \perp \ \begin{pmatrix} M_{11}(\xi) & M_{12}(\xi) \\ M_{21}(\xi) & M_{22}(\xi) \end{pmatrix} \begin{pmatrix} x(\xi) \\ y(\xi) \end{pmatrix} + \begin{pmatrix} q_1(\xi) \\ q_2(\xi) \end{pmatrix} + \begin{pmatrix} w(\xi) \\ 0 \end{pmatrix} \ge 0, \ \forall \ \xi \in \Xi, \quad (29)$$

where $x(\cdot) : \xi \mapsto x(\xi) \in \mathbb{R}^{n_1}$ is the first-stage response function and $y(\cdot) \in \mathbb{R}^{n_2}$ is the second-stage response function. Let $\mathcal{H}$ be the Hilbert space comprised of all response functions $z(\cdot) := (x(\cdot), y(\cdot))$ from Ξ to $\mathbb{R}^n, n = n_1 + n_2$, equipped with the inner product

$$\langle z(\cdot), u(\cdot) \rangle := \mathbb{E}_\xi[z(\xi)^T u(\xi)] := \sum_{\xi \in \Xi} p(\xi) z(\xi)^T u(\xi), \quad (30)$$

where $p(\xi) > 0$ is the probability of scenario ξ and all such probabilities add up to one.

The EPDA for solving the two-stage SLCP (29) has the following specific iteration scheme [9, 10, 15].

Algorithm 3. The EPDA for two-stage SLCP

Given $x^k(\xi) \equiv x \ \forall \ \xi$, and $y^k(\cdot)$ and $w^k(\cdot)$ such that $\mathbb{E}_\xi[w^k(\xi)] = 0$; $r > e \ge 0$.

Step 1. Obtain $(\hat{x}^k(\xi), \hat{y}^k(\xi))$ by solving the following linear complementarity problem for each ξ

$$0 \le \begin{pmatrix} x \\ y \end{pmatrix} \ \perp \ \begin{pmatrix} M_{11}(\xi) + rI & M_{12}(\xi) \\ M_{21}(\xi) & M_{22}(\xi) + rI \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} q_1(\xi) + w^k(\xi) - rx^k(\xi) \\ q_2(\xi) - ry^k(\xi) \end{pmatrix} \ge 0.$$

Step 2. Set $x^{k+1}(\xi) = \mathbb{E}_\xi[\hat{x}^k(\xi)]$, $y^{k+1}(\xi) = \hat{y}^k(\xi)$ and

$$w^{k+1}(\xi) = w^k(\xi) - (r - e)(\hat{x}^k(\xi) - x^{k+1}(\xi)) \ \forall \ \xi.$$

Set $k := k + 1$, repeat.

Set

$$\text{rel.err}_1 = \frac{\| x - \prod_{\ge 0} \left[x - \left(\mathbb{E}_\xi[M_{11}(\xi)]x + \mathbb{E}_\xi[M_{12}(\xi)y(\xi)] + \mathbb{E}_\xi[q_1(\xi)] \right) \right] \|}{1 + \|x\|},$$

$$\text{rel.err}_2 = \max_\xi \left\{ \frac{\| y(\xi) - \prod_{\ge 0}[y(\xi) - (M_{21}(\xi)x + M_{22}(\xi)y(\xi) + q_2(\xi))] \|}{1 + \|y(\xi)\|} \right\},$$

where $(\prod_{\ge 0}(a))_j = \max\{a_j, 0\}$, and let $\text{rel.err} = \max\{\text{rel.err}_1, \text{rel.err}_2\}$. Algorithm 3 stops if $\text{rel.err} \le 10^{-5}$ or the iteration number ≥ 5000. In solving the LCP subproblems in Step 1 of Algorithm 3, we reformulate the complementarity problem to a nonsmooth

equation and solve the resulting equation by the semismooth Newton method introduced by Qi and Sun [5]. The code is adopted from [10], which is available in public domain[2]. All experiments are implemented in Matlab R2015b under Windows 7 operating system on a desktop with an Intel(R) Core i5-6600, 3.30GHz processor and 8GB of RAM.

4.1 Data Generation

Note that, a two-stage response function $z(\cdot) : \Xi \to \mathbb{R}^n$ in a finite discrete distribution setting can be equivalently written as a vector in $\mathbb{R}^{nK}$, which is defined as

$$z := (x(\xi^1), \cdots, x(\xi^K), y(\xi^1), \cdots, y(\xi^K)),$$

where K is the number of scenarios. Let $\mathcal{F}(z(\cdot)) : \mathcal{H} \to \mathcal{H}$ be the linear function of $z(\cdot)$ defined by $\mathcal{F}(z(\xi)) = M(\xi)z(\xi) + q(\xi)$ for every ξ with

$$M(\xi) = \begin{pmatrix} M_{11}(\xi) & M_{12}(\xi) \\ M_{21}(\xi) & M_{22}(\xi) \end{pmatrix} \text{ and } q(\xi) = \begin{pmatrix} q_1(\xi) \\ q_2(\xi) \end{pmatrix}.$$

From definition (30) of the inner product in the space $\mathcal{H}$, we have

$$\langle z(\cdot) - z'(\cdot), (\mathcal{F} + e\mathcal{P}_{\mathcal{M}})(z(\cdot)) - (\mathcal{F} + e\mathcal{P}_{\mathcal{M}})(z'(\cdot))\rangle = v^T DMDv + ev^T DPDv,$$

where v is the vector equivalent to the response function $z(\cdot) - z'(\cdot)$, and D, M, P are matrices respectively defined as

$$D = \begin{pmatrix} \bar{D} & \\ & \bar{D} \end{pmatrix}, M = \begin{pmatrix} \bar{M}_{11} & \bar{M}_{12} \\ \bar{M}_{21} & \bar{M}_{22} \end{pmatrix}, P = \begin{pmatrix} I_{n_1 K} - \bar{P} & 0 \\ 0 & 0 \end{pmatrix},$$

with $\bar{D} = \text{Diag}(\sqrt{p_i} I_{n_1})$, $\bar{M}_{jk} = \text{Diag}(M_{jk}(\xi^i)) \ \forall \ j, k = 1, 2$, and

$$\bar{P} = \bar{p}\bar{p}^T = \begin{pmatrix} p_1 I_{n_1} & \cdots & \sqrt{p_1 p_K} I_{n_1} \\ \vdots & & \vdots \\ \sqrt{p_k p_1} I_{n_1} & \cdots & p_K I_{n_1} \end{pmatrix}.$$

We have the following observations.

- The 2-stage SLCP is monotone if and only if matrix M is positive semi-definite, i.e., $(M(\xi) + M(\xi)^T)/2$ is symmetric positive semi-definite for every ξ.
- The 2-stage SLCP is e-elicitable monotone if and only if matrix $M + eP$ is positive semi-definite.
- Based on Theorem 5 in [9], a sufficient condition for the 2-stage SLCP being elicitable monotone is that there exists $\alpha > 0$ such that $\langle z(\cdot), \mathcal{F}(z(\cdot))\rangle > \alpha \|z(\cdot)\|^2$

[2]The user guide and the MATLAB code are in testing stage and are available upon request; they will be soon moved to the authors' websites.

$\forall z(\cdot) \in \mathcal{N}$. This condition is equivalent to the following matrix being positive definite:

$$\begin{pmatrix} \sum_{i=1}^{K} p_i M_{11}(\xi^i) & \sqrt{p_1} M_{12}(\xi^1) & \cdots & \sqrt{p_K} M_{12}(\xi^K) \\ \sqrt{p_1} M_{21}(\xi^1) & M_{22}(\xi^1) & & \\ \vdots & & \ddots & \\ \sqrt{p_K} M_{21}(\xi^K) & & & M_{22}(\xi^K) \end{pmatrix}.$$

Moreover, if $M(\xi)$ is symmetric for every ξ, then the above matrix is positive definite if $M_{22}(\xi)$ is positive definite and $\sum_{i=1}^{K} p_i \left(M_{11}(\xi^i) - M_{12}(\xi^i) M_{22}^{-1}(\xi^i) M_{21}(\xi^i)\right)$ is positive definite.

Based on these observations, we generate monotone and elicitable monotone SLCPs by the following rules.

- **Monotone SLCPs.** Matrix $M(\xi)$ for each ξ is generated to be a symmetric positive semi-definite matrix. For each ξ, we first generate a matrix $M_{tmp} \in \mathbb{R}^{n \times n}$ composed of entries uniformly distributed in (0,1), then set $M(\xi) = M_{tmp}^T M_{tmp}$. We also generate vector $x_{tmp} \in \mathbb{R}^{n_1}$ composed of entries uniformly distributed in (0,1) and generate vectors $y_{tmp}(\xi) \in \mathbb{R}^{n_2}, q_{tmp}(\xi) \in \mathbb{R}^n$ composed of entries uniformly distributed in (0,1) for every ξ, then set $q(\xi) = -M(\xi)[x_{tmp}^T, y_{tmp}^T(\xi)]^T - q_{tmp}(\xi)$. Probability $p(\xi)$ is randomly generated as $p(\xi) > 0$ and $\sum p(\xi) = 1$.

- **Elicitable monotone SLCPs.** Matrix $M_{22}(\xi)$ for each ξ is generated to be a symmetric positive definite matrix. In detail, for each ξ, we first generate M_{tmp} composed of entries uniformly distributed in (0,1), then set $M_{22}(\xi) = M_{tmp}^T M_{tmp} + 0.1 I_{n_1}$. For each ξ, $M_{12}(\xi)$ is generated to be matrices composed of entries uniformly distributed in (0,1), and $M_{21}(\xi) = M_{12}^T(\xi)$. For each ξ, $M_{11}(\xi)$ is generated to be a symmetric matrix by

$$M_{11}(\xi^i) = \begin{cases} M_{12}(\xi^i) M_{22}^{-1}(\xi^i) M_{21}(\xi^i) + I_{n_1}, & \text{when } i = 1, \cdots, K-1, \\ M_{12}(\xi^i) M_{22}^{-1}(\xi^i) M_{21}(\xi^i) - 0.995 I_{n_1}, & \text{when } i = K. \end{cases}$$

  The way to generate $q(\xi)$, $p(\xi)$ is the same with the rule for monotone case.

4.2 Experiment Design

The purpose of the experiment is to provide empirical guidelines for the choice of r and e in Algorithm 3, based on the analysis in Section 3. We also would like to see the sizes of the problems that Algorithm 3 is capable to solve. Recall that the rate of convergence is denoted by δ and we have

$$\delta = \delta(d) = \frac{d}{d+\sigma}, \tag{31}$$

$$d = d(r,e) = \frac{\sqrt{e^2 + 4r^2} + e}{2}. \tag{32}$$

Since $\delta(d)$ is strictly increasing in d and $d(\cdot, e)$ and $d(r, \cdot)$ are strictly increasing in r and e, respectively, we should generally keep both r and e as small as possible in the experiment, say $e < r < \sqrt{n}$ if possible, where $\sqrt{n}$ was the heuristic value used in [10].

We generate two groups of test examples for both monotone and elicitable monotone SLCPs.

Group 1. Fix the dimension of first and second stage variables as dim=[10, 10] and increase the number of scenarios (sn for short) as 5, 10, 25, 50, 100. For each setting, 10 problems are randomly generated and solved by Algorithm 3 under different values of parameters r and e, and the average number of iterations (iter for short) and CPU time (in second, time(s) for short) are recorded.

Group 2. Fix sn=25 and increase the dimension to [20,20], [30,30], [40,40], [50,50] and [60,60]. For each setting, 10 problems are randomly generated and solved by Algorithm 3 under different values of parameters r and e, and the average number of iterations and CPU time are recorded.

4.3 Numerical Results for the Monotone Case

For Group 1, we apply Algorithm 3 to solve the randomly generated monotone problems with parameter $r = 1$, $e = 0$, 0.25, 0.5, 0.75. For each combination of r, e and sn, ten random problems are solved. The average number of iteration and the average computational time of the ten problems are listed in Table 1 and drawn in Figure 1. It can be seen that when the number of scenarios increases, the convergence follows the same trend presented in references [10, 15], which takes almost constant number of iterations and more computing time when the number of scenarios rises. In addition, for the fixed value of $r = 1$, the convergence of Algorithm 3 becomes slower when parameter e grows.

Table 1: Monotone results while sn increases (dim=[10,10], $r = 1$)

sn	$e = 0$		$e = 0.25$		$e = 0.5$		$e = 0.75$	
	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)
5	62	0.1	69	0.1	96	0.1	207	0.2
10	67	0.1	99	0.2	157	0.3	329	0.5
25	68	0.3	92	0.4	146	0.5	307	1.1
50	76	0.6	91	0.7	134	1.0	278	2.0
100	74	1.1	93	1.4	146	2.1	303	4.2

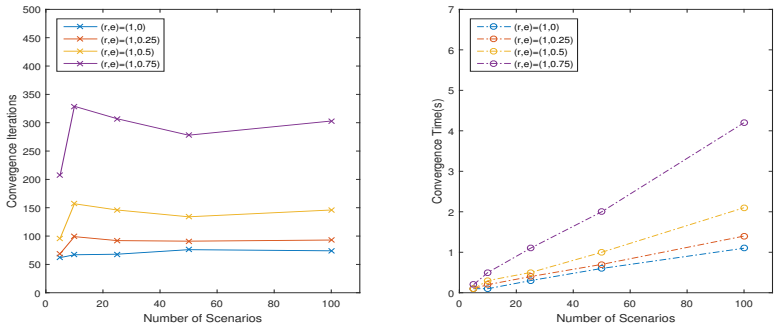

Figure 1: Results for monotone case when sn increases (dim=[10,10])

For Group 2, we apply Algorithm 3 to solve the randomly generated monotone problems under the same choice of parameters $r = 1$, $e = 0$, 0.25, 0.5 and 0.75, respectively.

For each combination of r, e and sn, ten random problems are solved. The average number of iteration and the average time of computation of the ten problems are presented in Table 2 and Figure 2. It is shown that both the number of iterations and time for convergence increase when the dimension grows, which agrees with the results of [10, 15]. Moreover, the speed will generally decline when the parameter e increases.

Table 2: Monotone results while dim increases (sn=25, $r = 1$)

dim	$e = 0$		$e = 0.25$		$e = 0.5$		$e = 0.75$	
	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)
[20,20]	122	0.5	168	0.7	251	1.0	514	2.2
[30,30]	160	0.8	216	1.0	329	1.5	665	3.3
[40,40]	266	1.7	358	2.3	540	3.4	1088	8.3
[50,50]	265	2.3	352	3.0	527	4.4	1090	10.7
[60,60]	336	3.4	454	4.3	685	7.1	1377	14.5

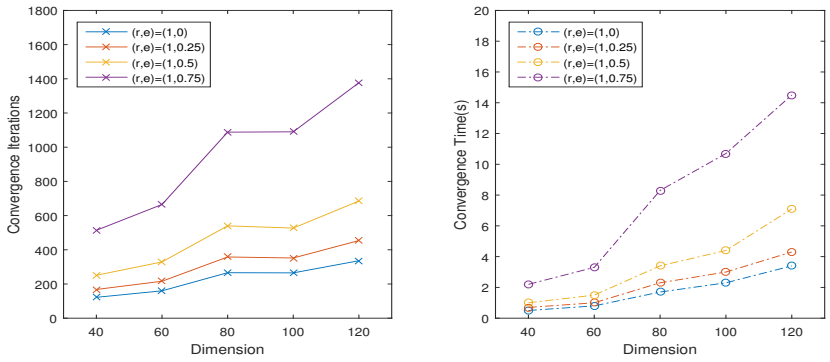

Figure 2: Results for monotone case when dim increases (sn=25)

4.4 Numerical Results for the Elicitable Monotone Case

Before applying Algorithm 3 to solve the test problems in the two groups of the nonmonotone case, we examine whether the matrix $M + eP$ is positive semi-definite under $e = 0, 1, 2, \cdots$ to reveal the elicitable monotonicity for all the problems in Group 1 and Group 2. Table 3 and Table 4 present the number of problems being e-elicited monotone at level e in Group 1 and Group 2, respectively. It can be observed that in this particular experiment, all the test problems in the two groups are nonmonotone at $e = 0$, but they will be e-elicited monotone at level $e \geq 5$ in Group 1 and at level $e \geq 2$ in Group 2. Moreover, the number of e-elicitable monotone problems increases when parameter e grows.

We then conduct Algorithm 3 to solve the test problems in the two groups with $e = 1, 2, \cdots, 10$ and $r = e + 1$. It can be observed that all the test problems can be successfully solved by Algorithm 3 under such choices of r and e, even for problems without elicited monotonicity under a small e, such as $e = 1$. In addition, it can be seen that, similar to the results of the monotone case, under the same (r, e), it takes

Table 3: Number of e-elicitable monotone problems in Group 1

e	sn					sum
	5	10	25	50	100	
0	0	0	0	0	0	0
1	0	0	0	1	1	2
2	7	10	10	10	10	47
3	9	10	10	10	10	49
4	9	10	10	10	10	49
5	10	10	10	10	10	50

Table 4: Number of e-elicitable monotone problems in Group 2

e	dim					sum
	[20,20]	[30,30]	[40,40]	[50,50]	[60,60]	
0	0	0	0	0	0	0
1	1	1	0	1	0	3
2	10	10	10	10	10	50

Table 5: Elicitable monotone results while sn increases (dim=[10,10], $r = e + 1$)

e	sn=5		sn=10		sn=25		sn=50		sn=100	
	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)
1	186	0.2	164	0.3	189	0.7	216	1.6	268	3.8
2	264	0.3	217	0.4	221	0.9	232	1.8	300	4.3
3	341	0.3	292	0.5	290	1.2	278	2.1	323	4.7
4	430	0.4	370	0.6	370	1.4	357	2.6	368	5.2
5	519	0.5	448	0.7	451	1.7	437	3.1	434	6.0
6	609	0.5	528	0.9	534	2.0	518	3.7	506	7.1
7	697	0.6	608	1.0	618	2.3	599	4.3	585	8.3
8	786	0.7	688	1.1	702	2.6	681	4.7	665	9.3
9	874	0.8	768	1.4	785	3.1	762	5.7	745	10.6
10	963	0.9	848	1.4	869	3.2	843	5.9	826	11.8

Table 6: Elicitable monotone results while dim increases (sn=25, $r = e + 1$)

e	dim=[20,20]		dim=[30,30]		dim=[40,40]		dim=[50,50]		dim=[60,60]	
	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)
1	290	1.3	342	1.7	416	3.6	508	5.7	660	8.3
2	310	1.4	371	2.0	428	3.9	515	6.6	667	8.3
3	342	1.6	393	2.2	460	4.1	537	6.6	674	9.3
4	385	1.8	418	2.3	480	4.3	567	6.6	691	9.0
5	444	2.1	456	2.5	507	4.6	600	7.1	719	9.8
6	522	2.3	506	2.6	544	5.2	615	7.4	733	10.0
7	603	2.8	570	3.1	585	5.6	646	8.0	765	10.1
8	686	3.2	639	3.5	646	6.1	686	8.7	798	11.2
9	769	3.6	706	3.8	723	6.8	732	8.5	843	11.0
10	852	4.0	784	4.3	803	7.4	769	8.9	889	11.4

almost constant number of iterations and more time for convergence when the number of scenarios increases, while more iterations and more time are necessary for convergence when the dimension grows. In addition, both the number of iterations and the time become larger when the value of e rises. The detailed results under $e \geq 1, r = e + 1$ of Group 1 and Group 2 are presented in Table 5 and Table 6, respectively.

Figure 3 shows the trend of convergence when the value of e grows in the 50 test problems in Group 1 and Group 2, respectively. It should be pointed out that parameter r is increasing together with the growth of e in the above experiments. Note that, the problems in Group 1 are e-elicited monotone when $e \geq 5$. Due to the requirement of $r > e$, the choice of r is at least larger than 5. Table 5 shows that the computing time is around twice under $r = 6$, $e = 5$ than that under $r = 2$, $e = 1$. Similar observations can be obtained in Group 2. In addition, we can see from Figure 3 that increasing e leads to increasing r and decreasing speed of convergence. It should be also pointed out that Algorithm 3 fails under the choice of $e = 0, r = 1$ in this particular experiment, but Algorithm 3 works for all the test problems under $e = 0$ and $r \geq 2$.

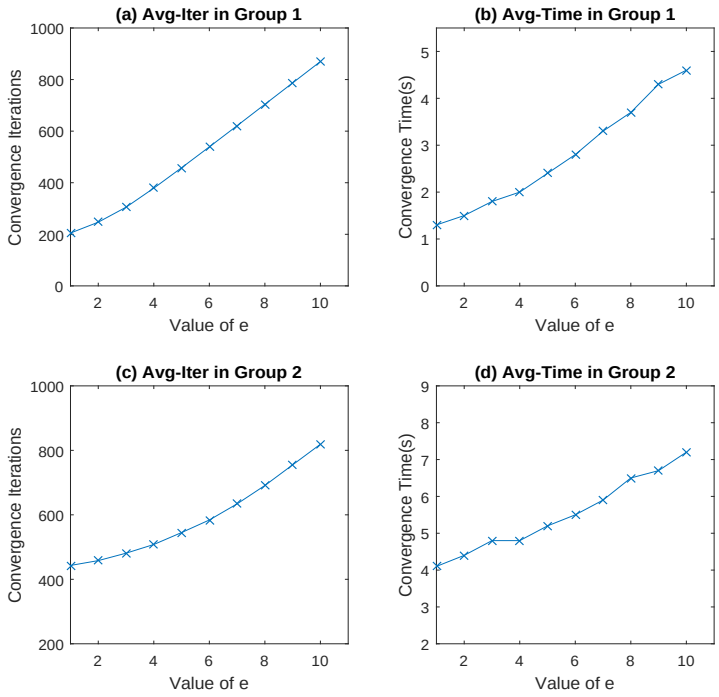

Figure 3: Results for elicitable monotone case under different e

In the following, we observe the impact of e under fixed r. For Group 1, we conduct Algorithm 3 under the choices of $r = 2, 4$ and $e = 0, r - 1$, respectively. The results are provided in Table 7 and Figure 4. It can be observed that $(r, e) = (2, 0)$ leads to the best performance. In addition, the influence of e seems less significant than the influence of r, which is observed from that, for fixed r, both the number of iterations and the computing time under different values of e are only slightly changed.

Similarly to Group 1, we conduct Algorithm 3 under the choices of $r = 2, 4$ and $e = 0, r - 1$, respectively, for Group 2. Corresponding results are presented in Table 8

Table 7: Elicitable monotone results while sn increases (dim=[10,10])

sn	$r=2$				$r=4$			
	$e=0$		$e=1$		$e=0$		$e=3$	
	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)
5	173	0.2	186	0.2	354	0.3	341	0.3
10	148	0.3	164	0.3	317	0.5	292	0.5
25	161	0.6	189	0.7	326	1.2	290	1.2
50	166	1.2	216	1.6	317	2.3	278	2.1
100	187	2.7	268	3.8	312	4.4	323	4.7

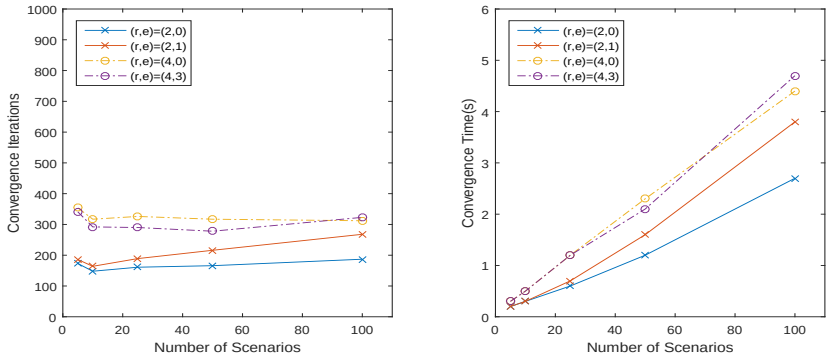

Figure 4: Results for elicitable monotone case when sn increases (dim=[10,10])

Table 8: Elicitable monotone results while dim increases (sn=25)

dim	$r=2$				$r=4$			
	$e=0$		$e=1$		$e=0$		$e=3$	
	iter	time(s)	iter	time(s)	iter	time(s)	iter	time(s)
[20,20]	216	1.0	290	1.3	322	1.4	342	1.6
[30,30]	242	1.2	342	1.7	303	1.6	393	2.2
[40,40]	245	1.9	416	3.6	308	2.5	460	4.1
[50,50]	290	3.3	508	5.7	277	2.9	537	6.6
[60,60]	335	4.2	660	8.3	301	4.1	674	9.3

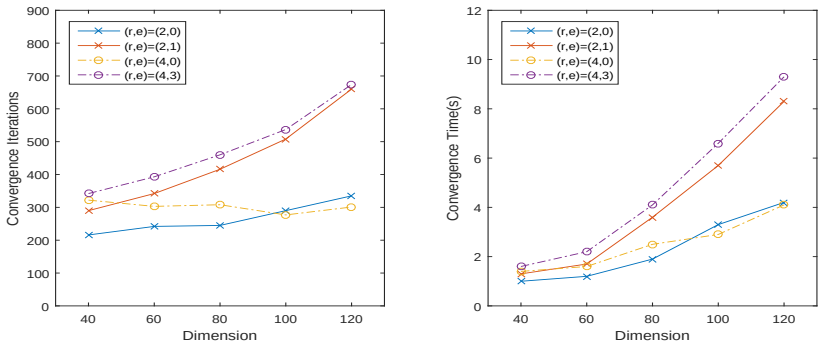

Figure 5: Results for elicitable monotone case when dim increases (sn=25)

and Figure 5. The parameter r seems to impact the convergence speed more than the parameter e. In this experiment, we achieve the best performance under the choice of $(r, e) = (2, 0)$ for the problems of smaller dimension and achieve the best performance under $(r, e) = (4, 0)$ for the problems of larger dimension. However, the convergence will become slower when choosing a larger e under a fixed r.

In summary, for fixed r, as e increases, the algorithm tends to slowdown and the performance of the EPDA seems more dependent on the choice of r. As a rule of thumb, for SLCPs, it looks that smaller e leads to better rate of convergence and the choice of r is more important than the choice of e. Overall, the computational results are consistent to our theoretical results (31) and (32).

A plausible theoretical explanation to why the choice of r is more important than the choice of e can be made from the following fact: We have $d'_r(r, e) > d'_e(r, e) > 0$ in the area $r > e$, which can be derived from (32).

5 Conclusions

1. It is shown that the mapping $(T + eP_{\mathcal{M}})_{\mathcal{N}}$ can be viewed as a generalized Yosida-regularization of Spingarn's partial inverse operator $T_{\mathcal{N}}$, which provides a rationale for the application of the PPA to elicitable monotone SVIs.

2. Under strong elicitable monotonicity of mapping T with modulus $\sigma > 0$, the EPDA converges at rate $\frac{c^{-1}}{c^{-1}+\sigma}$ with respect to the r-norm of pair (x, w), where $c = \frac{\sqrt{e^2+4r^2}-e}{2r^2} \leq r^{-1}$. When $e = 0$ (the monotone case), this result reproduces the result in [10], the rate of convergence for monotone PDA under the strongly monotonicity of mapping T with modulus $\sigma > 0$.

3. The above result explains why the convergence speed of the EPDA is generally slower than the monotone PDA and why the rate of convergence is closer to 1 when parameter e grows. It also gives possible explanations why smaller r and e are generally preferred and why the choice of r is more important than the choice of e in the numerical experiment.

4. The preliminary numerical experiment confirms the theoretical result of Sections 2 and 3. In particular, it is observed that the elicited version of PDA are generally slower than the non-elicited version in both number of iterations and computational time. Furthermore, the elicited version becomes much slower if the elicitation constant e is set too large. To avoid this, the numerical results appear to suggest to use relatively small e and r for a heuristic start in practice and only increase the value of e if the EPDA does not converge well.

References

[1] Eckstein, J., Bertsekas, D.P.: On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Program. 55(3), 293-318 (1992)

[2] Lu, Y., Sun, J., Zhang, M., Zhang, Y.: A stochastic variational inequality approach to the Nash equilibrium model of a manufacturer-supplier game under uncertainty. preprint. Department of Analytics and Operations, National University of Singapore (2020)

[3] Minty, G.J.: Monotone (nonlinear) operators in Hilbert space. Duke Math. J. 29(3), 341-346 (1962)

[4] Pennanen, T.: Local convergence of the proximal point algorithm and multiplier methods without monotonicity. Math. Oper. Res. 27(1), 170-191 (2002)

[5] Qi, L., Sun, J.: A nonsmooth version of Newton's method. Math. Program. 58, 353-367 (1993)

[6] Spingarn, J.E.: Partial inverse of a monotone operator. Appl. Math. Optim. 10(3), 247-265 (1983)

[7] Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. 14(5), 877-898 (1976)

[8] Rockafellar, R.T.: Progressive decoupling of linkages in monotone variational inequalities and convex optimization. Proceedings of the 10th International Conference on Nonlinear Analysis and Convex Analysis (Chitose, Japan), 1-21 (2017)

[9] Rockafellar, R.T.: Progressive decoupling of linkages in optimization and variational inequalities with elicitable convexity or monotonicity. Set-Valued Var. Anal. 27, 863-893 (2019)

[10] Rockafellar, R.T., Sun, J.: Solving monotone stochastic variational inequalities and complementarity problems by progressive hedging. Math. Program. 174, 453-471 (2019)

[11] Rockafellar, R.T., Sun, J.: Solving Lagrangian variational inequalities with applications to stochastic programming. Math. Program. 181, 435-451 (2020)

[12] Rockafellar, R.T., Wets, R.-J.B.: Scenarios and policy aggregation in optimization under uncertainty. Math. Oper. Res. 16(1), 119-147 (1991)

[13] Rockafellar, R.T., Wets, R.-J.B.: Stochastic variational inequalities: single-stage to multistage. Math. Program. 165(1), 331-360 (2017)

[14] Zhang, M., Hou, L.S., Sun, J., Yan, A.L.: A model of multistage risk-averse stochastic optimization and its solution by scenario-based decomposition algorithms. Asia-Pacific J. Oper. Res. 37(4), 2040004, doi:10.1142/S0217595920400047 (2020)

[15] Zhang, M., Sun, J., Xu, H.: Two-stage quadratic games under uncertainty and their solution by progressive hedging algorithms. SIAM J. Optim. 29(3), 1799-1818 (2019)