OPTIMAL PRIMAL-DUAL METHODS FOR A CLASS OF SADDLE POINT PROBLEMS

YUNMEI CHEN*, GUANGHUI LAN†, AND YUYUAN OUYANG‡

Abstract. We present a novel accelerated primal-dual (APD) method for solving a class of deterministic and stochastic saddle point problems (SPP). The basic idea of this algorithm is to incorporate a multi-step acceleration scheme into the primal-dual method without smoothing the objective function. For deterministic SPP, the APD method achieves the same optimal rate of convergence as Nesterov's smoothing technique. Our stochastic APD method exhibits an optimal rate of convergence for stochastic SPP not only in terms of its dependence on the number of the iteration, but also on a variety of problem parameters. To the best of our knowledge, this is the first time that such an optimal algorithm has been developed for stochastic SPP in the literature. Furthermore, for both deterministic and stochastic SPP, the developed APD algorithms can deal with the situation when the feasible region is unbounded, as long as a saddle point exists. In the unbounded case, we incorporate the modified termination criterion introduced by Monteiro and Svaiter in solving SPP problem posed as monotone inclusion, and demonstrate that the rate of convergence of the APD method depends on the distance from the initial point to the set of optimal solutions. Some preliminary numerical results of the APD method for solving both deterministic and stochastic SPPs are also included.

Keywords: saddle point problem, optimal methods, stochastic approximation, stochastic programming, complexity, large deviation

1. Introduction. Let $\mathcal{X}$ and $\mathcal{Y}$ denote the finite-dimensional vector spaces equipped with an inner product $\langle \cdot, \cdot \rangle$ and norm $\| \cdot \|$, and $X \subseteq \mathcal{X}$, $Y \subseteq \mathcal{Y}$ be given closed convex sets. The basic problem of interest in this paper is the saddle-point problem (SPP) given in the form of:

$$\min_{x \in X} \left\{ f(x) := \max_{y \in Y} G(x) + \langle Kx, y \rangle - J(y) \right\}. \tag{1.1}$$

Here, $G(x)$ is a general smooth convex function and K is a linear operator such that, for some $L_G, L_K \geq 0$,

$$G(u) - G(x) - \langle \nabla G(x), u - x \rangle \leq \frac{L_G}{2} \|u - x\|^2 \text{ and } \|Ku - Kx\|_* \leq L_K \|u - x\|, \quad \forall x, u \in X, \tag{1.2}$$

and $J : Y \to \mathbb{R}$ is a relatively simple, proper, convex, lower semi-continuous (l.s.c.) function (i.e., problem (2.5) is easy to solve). In particular, if J is the convex conjugate of some convex function F and $Y \equiv \mathcal{Y}$, then (1.1) is equivalent to the primal problem:

$$\min_{x \in X} G(x) + F(Kx). \tag{1.3}$$

Problems of these types have recently found many applications in data analysis, especially in imaging processing and machine learning. In many of these applications, $G(x)$ is a convex data fidelity term, while $F(Kx)$ is a certain regularization, e.g., total variation [47], low rank tensor [22, 50], overlapped group lasso [19, 30], and graph regularization [19, 49].

This paper focuses on first-order methods for solving both deterministic SPP, where exact first-order information on f is available, and stochastic SPP, where we only have access to inexact information about f. Let us start by reviewing a few existing first-order methods in both cases.

*Department of Mathematics, University of Florida (yun@math.ufl.edu). This author was partially supported by NSF grants DMS-1115568, IIP-1237814 and DMS-1319050.

†Department of Industrial and System Engineering, University of Florida (glan@ise.ufl.edu). This author was partially supported by NSF grant CMMI-1000347, ONR grant N00014-13-1-0036, NSF DMS-1319050, and NSF CAREER Award CMMI-1254446.

‡Department of Industrial and System Engineering, University of Florida (ouyang@ufl.edu). Part of the research was done while the author was a PhD student at the Department of Mathematics, University of Florida. This author was partially supported by AFRL Mathematical Modeling Optimization Institute. The authors acknowledge the University of Florida Research Computing (http://researchcomputing.ufl.edu) for providing computational resources.

1.1. Deterministic SPP. Since the objective function f defined in (1.1) is nonsmooth in general, traditional nonsmooth optimization methods, e.g., subgradient methods, would exhibit an $\mathcal{O}(1/\sqrt{N})$ rate of convergence when applied to (1.1) [36], where N denotes the number of iterations. However, following the breakthrough paper by Nesterov [41], much research effort has been devoted to the development of more efficient methods for solving problem (1.1).

(1) *Smoothing techniques.* In [41], Nesterov proposed to approximate the nonsmooth objective function f in (1.1) by a smooth one with Lipschitz-continuous gradient. Then, the smooth approximation function is minimized by an accelerated gradient method in [39, 40]. Nesterov demonstrated in [41] that, if X and Y are compact, then the rate of convergence of this smoothing scheme applied to (1.1) can be bounded by:

$$\mathcal{O}\left(\frac{L_G}{N^2} + \frac{L_K}{N}\right), \tag{1.4}$$

which significantly improves the previous bound $\mathcal{O}(1/\sqrt{N})$. It can be seen that the rate of convergence in (1.4) is actually optimal, based on the following observations:

a) There exists a function G with Lipschitz continuous gradients, such that for any first-order method, the rate of convergence for solving $\min_{x\in X} G(x)$ is at most $\mathcal{O}\left(L_G/N^2\right)$ [40].

b) There exists $b \in Y$, where Y is a convex compact set of $\mathbb{R}^m$ for some $m > 0$, and a linear bounded operator K, such that for any first-order method, the rate of convergence for solving $\min_{x\in X}\max_{y\in Y}\langle Kx, y\rangle - J(y) := \min_{x\in X}\max_{y\in Y}\langle Kx - b, y\rangle$ is at most $\mathcal{O}\left(L_K/N\right)$ [37, 34].

Nesterov's smoothing technique has been extensively studied (see, e.g., [38, 2, 26, 10, 42, 51, 4, 25]). Observe that in order to properly apply these smoothing techniques, we need to assume either X or Y to be bounded.

(2) *Primal-dual methods.* While Nesterov's smoothing scheme or its variants rely on a smooth approximation to the original problem (1.1), primal-dual methods work directly with the original saddle-point problem. This type of method was first presented by Arrow et al. [1] and named as the primal-dual hybrid gradient (PDHG) method in [52]. The results in [52, 9, 12] showed that the PDHG algorithm, if employed with well-chosen stepsize policies, exhibits very fast convergence in practice, especially for some imaging applications. Recently Chambolle and Pork [9] presented a unified form of primal-dual algorithms, and demonstrated that, with a properly specified stepsize policy and averaging scheme, these algorithms can also achieve the $\mathcal{O}(1/N)$ rate of convergence. They also discussed possible ways to extend primal-dual algorithms to deal with the case when both X and Y are unbounded. In the original work of Chambolle and Pork, they assume G to be relatively simple so that the subproblems can be solved efficiently. With little additional effort, one can show that, by linearizing G at each step, their method can also be applied for a general smooth convex function G and the rate of convergence of this modified algorithm is given by

$$\mathcal{O}\left(\frac{L_G + L_K}{N}\right). \tag{1.5}$$

The rate of convergence in (1.4) has a significantly better dependence on L_G than that in (1.5). Therefore, Nesterov's smoothing scheme allows a very large Lipschitz constant L_G (as big as $\mathcal{O}(N)$) without affecting the rate of convergence (up to a constant factor of 2). This is desirable in many data analysis applications (e.g., image processing), where L_G is usually significantly bigger than L_K. Note that the primal-dual methods are also related to the Douglas-Rachford splitting method [11, 29] and a pre-conditioned version of the alternating direction method of multipliers [13, 16] (see, e.g., [9, 12, 18, 33] for detailed reviews on the relationship between the primal-dual methods and other algorithms, as well as recent theoretical developments).

(3) *Extragradient methods for variation inequality (VI) reformulation.* Motivated by Nesterov's work, Nemirovski presented a mirror-prox method, by modifying Korpelevich's extragradient algorithm [23], for solving a more general class of variational inequalities [34] (see also [20]). Similar to the primal-dual methods mentioned above, the extragradient methods update iterates on both the primal space $\mathcal{X}$ and dual space $\mathcal{Y}$, and

do not require any smoothing technique. The difference is that each iteration of the extragradient methods requires an extra gradient descent step. Nemirovski's method, when specialized to (1.1), also exhibits a rate of convergence given by (1.5), which, in view of our previous discussion, is not optimal in terms of its dependence on L_G. It can be shown that, in some special cases (e.g., G is quadratic), one can write explicitly the (strongly concave) dual function of $G(x)$ and obtain a result similar to (1.4), e.g., by applying an improved algorithm in [20]. However, this approach would increase the dimension of the problem and cannot be applied for a general smooth function G. It should be noted that, while Nemirovski's initial work only considers the case when both X and Y are bounded, Monteiro and Svaiter [31] recently showed that extragradient methods can deal with unbounded sets X and Y by using a slightly modified termination criterion.

1.2. Stochastic SPP. While deterministic SPP has been extensively explored, the study on stochastic first-order methods for stochastic SPP is still quite limited. In the stochastic setting, we assume that there exists a *stochastic oracle* ($\mathcal{SO}$) that can provide unbiased estimators to the gradient operators $\nabla G(x)$ and $(-Kx, K^T y)$. More specifically, at the i-th call to $\mathcal{SO}$, $(x_i, y_i) \in X \times Y$ being the input, the oracle will output the *stochastic gradient* $(\hat{\mathcal{G}}(x_i), \hat{\mathcal{K}}_x(x_i), \hat{\mathcal{K}}_y(y_i)) \equiv (\mathcal{G}(x_i, \xi_i), \mathcal{K}_x(x_i, \xi_i), \mathcal{K}_y(y_i, \xi_i))$ such that

$$\mathbb{E}[\hat{\mathcal{G}}(x_i)] = \nabla G(x_i), \qquad \mathbb{E}\left[\begin{pmatrix} -\hat{\mathcal{K}}_x(x_i) \\ \hat{\mathcal{K}}_y(y_i) \end{pmatrix}\right] = \begin{pmatrix} -Kx_i \\ K^T y_i \end{pmatrix}. \tag{1.6}$$

Here $\{\xi_i \in \mathbb{R}^d\}_{i=1}^{\infty}$ is a sequence of i.i.d. random variables. In addition, we assume that, for some $\sigma_{x,G}, \sigma_y, \sigma_{x,K} \geq 0$, the following assumption holds for all $x_i \in X$ and $y_i \in Y$:

A1. $\mathbb{E}[\|\hat{\mathcal{G}}(x_i) - \nabla G(x_i)\|_*^2] \leq \sigma_{x,G}^2$, $\mathbb{E}[\|\hat{\mathcal{K}}_x(x_i) - Kx_i\|_*^2] \leq \sigma_y^2$ and $\mathbb{E}[\|\hat{\mathcal{K}}_y(y_i) - K^T y_i\|_*^2] \leq \sigma_{x,K}^2$.

Sometimes we simply denote $\sigma_x := \sqrt{\sigma_{x,G}^2 + \sigma_{x,K}^2}$ for the sake of notational convenience. Stochastic SPP often appears in machine learning applications. For example, for problems given in the form of (1.3), $G(x)$ (resp. $F(Kx)$) can be used to denote a smooth (resp. nonsmooth) expected convex loss function. It should also be noted that deterministic SPP is a special case of the above setting with $\sigma_x = \sigma_y = 0$.

In view of the classic complexity theory for convex programming [36, 21], a lower bound on the rate of convergence for solving stochastic SPP is given by

$$\Omega\left(\frac{L_G}{N^2} + \frac{L_K}{N} + \frac{\sigma_x + \sigma_y}{\sqrt{N}}\right), \tag{1.7}$$

where the first two terms follow from the discussion after (1.4) and the last term follows from Section 5.3 and 6.3 of [36]. However, to the best of our knowledge, there does not exist an optimal algorithm in the literature which exhibits exactly the same rate of convergence as in (1.7), although there are a few general-purpose stochastic optimization algorithms which possess different nearly optimal rates of convergence when applied to above stochastic SPP.

(1) *Mirror-descent stochastic approximation (MD-SA).* The MD-SA method developed by Nemirovski et al. in [35] originates from the classical stochastic approximation (SA) of Robbins and Monro [46]. The classical SA mimics the simple gradient descent method by replacing exact gradients with stochastic gradients, but can only be applied to solve strongly convex problems (see also Polyak [44] and Polyak and Juditsky [45], and Nemirovski et al. [35] for an account for the earlier development of SA methods). By properly modifying the classical SA, Nemirovski et al. showed in [35] that the MD-SA method can optimally solve general nonsmooth stochastic programming problems. The rate of convergence of this algorithm, when applied to the stochastic SPP, is given by (see Section 3 of [35])

$$\mathcal{O}\left\{(L_G + L_K + \sigma_x + \sigma_y)\frac{1}{\sqrt{N}}\right\}. \tag{1.8}$$

However, the above bound is significantly worse than the lower bound in (1.7) in terms of its dependence on both L_G and L_K.

(2) *Stochastic mirror-prox (SMP).* In order to improve the convergence of the MD-SA method, Juditsky et al. [21] developed a stochastic counterpart of Nemirovski's mirror-prox method for solving general variational inequalities. The stochastic mirror-prox method, when specialized to the above stochastic SPP, yields a rate of convergence given by

$$\mathcal{O}\left\{\frac{L_G + L_K}{N} + \frac{\sigma_x + \sigma_y}{\sqrt{N}}\right\}. \tag{1.9}$$

Note however, that the above bound is still significantly worse than the lower bound in (1.7) in terms of its dependence on L_G.

(3) *Accelerated stochastic approximation (AC-SA).* More recently, Lan presented in [24] (see also [14, 15]) a unified optimal method for solving smooth, nonsmooth and stochastic optimization by developing a stochastic version of Nesterov's method [39, 40]. The developed AC-SA algorithm in [24], when applied to the aforementioned stochastic SPP, possesses the rate of convergence given by

$$\mathcal{O}\left\{\frac{L_G}{N^2} + (L_K + \sigma_x + \sigma_y)\frac{1}{\sqrt{N}}\right\}. \tag{1.10}$$

However, since the nonsmooth term in f of (1.1) has certain special structure, the above bound is still significantly worse than the lower bound in (1.7) in terms of its dependence on L_K. It should be noted that some improvement for AC-SA has been made by Lin et al. [28] by applying the smoothing technique to (1.1). However, such an improvement works only for the case when Y is bounded and $\sigma_y = \sigma_{x,K} = 0$. Otherwise, the rate of convergence of the AC-SA algorithm will depend on the "variance" of the stochastic gradients computed for the smooth approximation problem, which is usually unknown and difficult to characterize (see Section 3 for more discussions).

Therefore, none of the stochastic optimization algorithms mentioned above could achieve the lower bound on the rate of convergence in (1.7).

1.3. Contribution of this paper. Our contribution in this paper mainly consists of the following three aspects. Firstly, we present a new primal-dual type method, namely the accelerated primal-dual (APD) method, that can achieve the optimal rate of convergence in (1.4) for deterministic SPP. The basic idea of this algorithm is to incorporate a multi-step acceleration scheme into the primal-dual method in [9]. We demonstrate that, without requiring the application of the smoothing technique, this method can also achieve the same optimal rate of convergence as Nesterov's smoothing scheme when applied to (1.1). We also show that the cost per iteration for APD is comparable to that of Nesterov's smoothing scheme. Hence our method can efficiently solve problems with a big Lipschtiz constant L_G.

Secondly, in order to solve stochastic SPP, we develop a stochastic counterpart of the APD method, namely stochastic APD and demonstrate that it can actually achieve the lower bound on the rate of convergence in (1.7). Therefore, this algorithm exhibits an optimal rate of convergence for stochastic SPP not only in terms of its dependence on N, but also on a variety of problem parameters including, L_G, L_K, σ_x and σ_y. To the best of our knowledge, this is the first time that such an optimal algorithm has been developed for stochastic SPP in the literature. In addition, we investigate the stochastic APD method in more details, e.g., by developing the large-deviation results associated with the rate of convergence of the stochastic APD method.

Thirdly, for both deterministic and stochastic SPP, we demonstrate that the developed APD algorithms can deal with the situation when either X or Y is unbounded, as long as a saddle point of problem (1.1) exists. We incorporate into the APD method the termination criterion employed by Monteiro and Svaiter [32] for solving variational inequalities, and generalize it for solving stochastic SPP. In both deterministic and stochastic cases, the rate of convergence of the APD algorithms will depend on the distance from the initial point to the set of optimal solutions.

Finally, we demonstrate the advantages of the proposed deterministic and stochastic ADP method for solving certain classes of SPP through numerical experiments.

1.4. Organization of the paper. We present the APD methods and discuss their main convergence properties for solving deterministic and stochastic SPP problems, respectively, in Sections 2 and 3. In order to facilitate the readers, we put the proofs of our main results in Section 4. Experimental results on deterministic and stochastic APD methods including comparisons with several existing algorithms are presented in section 5. Some brief concluding remarks are made in Section 6.

2. Accelerated primal-dual method for deterministic SPP. Our goal in this section is to present an accelerated primal-dual method for deterministic SPP and discuss its main convergence properties.

The study on first-order primal-dual method for nonsmooth convex optimization has been mainly motivated by solving total variation based image processing problems (e.g. [52, 12, 43, 9, 6, 17]). Algorithm 1 shows a primal-dual method summarized in [9] for solving a special case of problem (1.1), where $Y = R^m$ for some $m > 0$, and $J(y) = F^*(y)$ is the convex conjugate of a convex and l.s.c. function F.

Algorithm 1 Primal-dual method for solving deterministic SPP

1: Choose $x_1 \in X$, $y_1 \in Y$. Set $\bar{x}_1 = x_1$.
2: For $t = 1, \ldots, N$, calculate

$$y_{t+1} = \operatorname*{argmin}_{y \in Y} \langle -K\bar{x}_t, y\rangle + J(y) + \frac{1}{2\tau_t}\|y - y_t\|^2, \tag{2.1}$$

$$x_{t+1} = \operatorname*{argmin}_{x \in X} G(x) + \langle Kx, y_{t+1}\rangle + \frac{1}{2\eta_t}\|x - x_t\|^2, \tag{2.2}$$

$$\bar{x}_{t+1} = \theta_t(x_{t+1} - x_t) + x_{t+1}. \tag{2.3}$$

3: Output $x^N = \frac{1}{N}\sum_{t=1}^N x_t$, $y^N = \frac{1}{N}\sum_{t=1}^N y_t$.

Algorithm 2 Accelerated primal-dual method for deterministic SPP

1: Choose $x_1 \in X, y_1 \in Y$. Set $x_1^{ag} = x_1, y_1^{ag} = y_1$, $\bar{x}_1 = x_1$.
2: For $t = 1, 2, \ldots, N-1$, calculate

$$x_t^{md} = (1 - \beta_t^{-1})x_t^{ag} + \beta_t^{-1}x_t, \tag{2.4}$$

$$y_{t+1} = \operatorname*{argmin}_{y \in Y} \langle -K\bar{x}_t, y\rangle + J(y) + \frac{1}{\tau_t}V_Y(y, y_t), \tag{2.5}$$

$$x_{t+1} = \operatorname*{argmin}_{x \in X} \langle \nabla G(x_t^{md}), x\rangle + \langle x, K^T y_{t+1}\rangle + \frac{1}{\eta_t}V_X(x, x_t), \tag{2.6}$$

$$x_{t+1}^{ag} = (1 - \beta_t^{-1})x_t^{ag} + \beta_t^{-1}x_{t+1}, \tag{2.7}$$

$$y_{t+1}^{ag} = (1 - \beta_t^{-1})y_t^{ag} + \beta_t^{-1}y_{t+1}, \tag{2.8}$$

$$\bar{x}_{t+1} = \theta_{t+1}(x_{t+1} - x_t) + x_{t+1}. \tag{2.9}$$

3: Output x_N^{ag}, y_N^{ag}.

The convergence of the sequence $\{(x_t, y_t)\}$ in Algorithm 1 has been studied in [43, 12, 9, 6, 17] for various choices of θ_t, and under different conditions on the stepsizes τ_t and η_t. In the study by Chambolle and Pock [9], they consider the case when constant stepsizes are used, i.e., $\tau_t = \tau$, $\eta_t = \eta$ and $\theta_t = \theta$ for some $\tau, \eta, \theta > 0$ for all $t \geq 1$. If $\tau\eta L_K^2 < 1$, where L_K is defined in (1.2), then the output (x^N, y^N) possesses a rate of convergence

of $\mathcal{O}(1/N)$ for $\theta = 1$, and of $O(1/\sqrt{N})$ for $\theta = 0$, in terms of partial duality gap (duality gap in a bounded domain, see (2.13) below).

One possible limitation of [9] is that both G and J need to be simple enough so that the two subproblems (2.1) and (2.2) in Algorithm 1 are easy to solve. To make Algorithm 1 applicable to more practical problems we consider more general cases, where J is simple, but G may not be so. In particular, we assume that G is a general smooth convex function satisfying (1.2). In this case, we can replace G in (2.2) by its linear approximation $G(x_t) + \langle \nabla G(x_t), x - x_t \rangle$. Then (2.2) becomes

$$x_{t+1} = \underset{x \in X}{\operatorname{argmin}} \langle \nabla G(x_t), x \rangle + \langle Kx, y_{t+1} \rangle + \frac{1}{2\eta_t} \|x - x_t\|^2. \tag{2.10}$$

In the following context, we will refer to this modified algorithm as the "linearized version" of Algorithm 1. By some extra effort we can show that, if for $t = 1, \ldots, N$, $0 < \theta_t = \tau_{t-1}/\tau_t = \eta_{t-1}/\eta_t \leq 1$, and $L_G \eta_t + L_K^2 \eta_t \tau_t \leq 1$, then (x^N, y^N) has an $\mathcal{O}((L_G + L_K)/N)$ rate of convergence in the sense of the partial duality gap.

As discussed in Section 1, the aforementioned rate of convergence for the linearized version of Algorithm 1 is the same as that proved in [9], and not optimal in terms of its dependence on L_G (see (1.5)). However, this algorithm solves the problem (1.1) directly without smoothing the nonsmooth objective function. Considering the primal-dual method as an alternative to Nesterov's smoothing method, and inspired by his idea of using accelerated gradient descent algorithm to solve the smoothed problem [39, 40, 41], we propose the following accelerated primal-dual algorithm that integrates the accelerated gradient descent algorithm into the linearized version of Algorithm 1.

Our accelerated primal-dual (APD) method is presented in Algorithm 2. Observe that in this algorithm, the superscript "ag" stands for "aggregated", and "md" stands for "middle". For any $x, u \in X$ and $y, v \in Y$, the functions $V_X(\cdot, \cdot)$ and $V_Y(\cdot, \cdot)$ are Bregman divergences defined as

$$V_X(x, u) := d_X(x) - d_X(u) - \langle \nabla d_X(u), x - u \rangle, \text{ and } V_Y(y, v) := d_Y(y) - d_Y(v) - \langle \nabla d_Y(v), y - v \rangle, \tag{2.11}$$

where $d_X(\cdot)$ and $d_Y(\cdot)$ are strongly convex functions with strong convexity parameters α_X and α_Y. For example, under the Euclidean setting, we can simply set $V_X(x, x_t) := \|x - x_t\|^2/2$ and $V_Y(y, y_t) := \|y - y_t\|^2/2$, and $\alpha_X = \alpha_Y = 1$. We assume that $J(y)$ is a simple convex function, so that the optimization problem in (2.5) can be solved efficiently.

Note that if $\beta_t = 1$ for all $t \geq 1$, then $x_t^{md} = x_t$, $x_{t+1}^{ag} = x_{t+1}$, and Algorithm 2 is the same as the linearized version of Algorithm 1. However, by specifying a different selection of β_t (e.g., $\beta_t = O(t)$), we can significantly improve the rate of convergence of Algorithm 2 in terms of its dependence on L_G. It should be noted that the iteration cost for the APD algorithm is about the same as that for the linearized version of Algorithm 1.

In order to analyze the convergence of Algorithm 2, it is necessary to introduce a notion to characterize the solutions of (1.1). Specifically, denoting $Z = X \times Y$, for any $\tilde{z} = (\tilde{x}, \tilde{y}) \in Z$ and $z = (x, y) \in Z$, we define

$$Q(\tilde{z}, z) := [G(\tilde{x}) + \langle K\tilde{x}, y \rangle - J(y)] - [G(x) + \langle Kx, \tilde{y} \rangle - J(\tilde{y})]. \tag{2.12}$$

It can be easily seen that $\tilde{z}$ is a solution of problem (1.1), if and only if $Q(\tilde{z}, z) \leq 0$ for all $z \in Z$. Therefore, if Z is bounded, it is suggestive to use the gap function

$$g(\tilde{z}) := \max_{z \in Z} Q(\tilde{z}, z) \tag{2.13}$$

to assess the quality of a feasible solution $\tilde{z} \in Z$. In fact, we can show that $f(\tilde{x}) - f^* \leq g(\tilde{z})$ for all $\tilde{z} \in Z$, where f^* denotes the optimal value of problem (1.1). However, if Z is unbounded, then $g(\tilde{z})$ is not well-defined even for a nearly optimal solution $\tilde{z} \in Z$. Hence, in the sequel, we will consider the bounded and unbounded case separately, by employing a slightly different error measure for the latter situation.

The following theorem describes the convergence properties of Algorithm 2 when Z is bounded.

THEOREM 2.1. *Suppose that for some $\Omega_X, \Omega_Y > 0$,*

$$\sup_{x_1, x_2 \in X} V_X(x_1, x_2) \leq \Omega_X^2 \quad \textit{and} \quad \sup_{y_1, y_2 \in Y} V_Y(y_1, y_2) \leq \Omega_Y^2. \tag{2.14}$$

Also assume that the parameters $\beta_t, \theta_t, \eta_t, \tau_t$ in Algorithm 2 are chosen such that for all $t \geq 1$,

$$\beta_1 = 1,\ \beta_{t+1} - 1 = \beta_t \theta_{t+1}, \tag{2.15}$$

$$0 < \theta_t \leq \min\{\frac{\eta_{t-1}}{\eta_t}, \frac{\tau_{t-1}}{\tau_t}\}, \tag{2.16}$$

$$\frac{\alpha_X}{\eta_t} - \frac{L_G}{\beta_t} - \frac{L_K^2 \tau_t}{\alpha_Y} \geq 0. \tag{2.17}$$

Then for all $t \geq 1$,

$$g(z_{t+1}^{ag}) \leq \frac{1}{\beta_t \eta_t} \Omega_X^2 + \frac{1}{\beta_t \tau_t} \Omega_Y^2. \tag{2.18}$$

There are various options for choosing the parameters β_t, η_t, τ_t and θ_t such that (2.15)–(2.17) hold. Below we provide such an example.

COROLLARY 2.2. *Suppose that (2.14) holds. In Algorithm 2, if the parameters are set to*

$$\beta_t = \frac{t+1}{2},\ \theta_t = \frac{t-1}{t},\ \eta_t = \frac{\alpha_X t}{2L_G + t L_K D_Y / D_X} \text{ and } \tau_t = \frac{\alpha_Y D_Y}{L_K D_X}, \tag{2.19}$$

where $D_X := \Omega_X \sqrt{2/\alpha_X}$ and $D_Y := \Omega_Y \sqrt{2/\alpha_Y}$, then for all $t \geq 2$,

$$g(z_t^{ag}) \leq \frac{2L_G D_X^2}{t(t-1)} + \frac{2L_K D_X D_Y}{t}. \tag{2.20}$$

Proof. It suffices to verify that the parameters in (2.19) *satisfies* (2.15)–(2.17) *in Theorem 2.1. It is easy to check that* (2.15) *and* (2.16) *hold. Furthermore,*

$$\tfrac{\alpha_X}{\eta_t} - \tfrac{L_G}{\beta_t} - \tfrac{L_K^2 \tau_t}{\alpha_Y} = \tfrac{2L_G + t L_K D_Y / D_X}{t} - \tfrac{2L_G}{t+1} - \tfrac{L_K D_Y}{D_X} \geq 0,$$

so (2.17) *holds. Therefore, by* (2.18)*, for all $t \geq 1$ we have*

$$\begin{aligned} g(z_t^{ag}) &\leq \tfrac{1}{\beta_{t-1}\eta_{t-1}} \Omega_X^2 + \tfrac{1}{\beta_{t-1}\tau_{t-1}} \Omega_Y^2 = \tfrac{4L_G + 2(t-1) L_K D_Y / D_X}{\alpha_X t(t-1)} \cdot \tfrac{\alpha_X}{2} D_X^2 + \tfrac{2L_K D_X / D_Y}{\alpha_Y t} \cdot \tfrac{\alpha_Y}{2} D_Y^2 \\ &= \tfrac{2L_G D_X^2}{t(t-1)} + \tfrac{2L_K D_X D_Y}{t}. \end{aligned}$$

□

Clearly, in view of (1.4), the rate of convergence of Algorithm 2 applied to problem (1.1) is optimal when the parameters are chosen according to (2.19). Also observe that we need to estimate D_Y/D_X to use these parameters. However, it should be pointed out that replacing the ratio D_Y/D_X in (2.19) by any positive constant only results in an increase in the RHS of (2.20) by a constant factor.

Now, we study the convergence properties of the APD algorithm for the case when $Z = X \times Y$ is unbounded, by using a perturbation-based termination criterion recently employed by Monteiro and Svaiter and applied to SPP [31, 33, 32]. This termination criterion is based on the enlargement of a maximal monotone operator, which is first introduced in [7]. One advantage of using this criterion is that its definition does not depend on the boundedness of the domain of the operator. More specifically, as shown in [32, 31], there always exists a perturbation vector v such that

$$\tilde{g}(\tilde{z}, v) := \max_{z \in Z} Q(\tilde{z}, z) - \langle v, \tilde{z} - z \rangle \tag{2.21}$$

is well-defined, although the value of $g(\tilde{z})$ in (2.13) may be unbounded if Z is unbounded. In the following result, we show that the APD algorithm can compute a nearly optimal solution $\tilde{z}$ with a small residue $\tilde{g}(\tilde{z}, v)$,

for a small perturbation vector v (i.e., $\|v\|$ is small). In addition, our derived iteration complexity bounds are proportional to the distance from the initial point to the solution set.

THEOREM 2.3. *Let $\{z_t^{ag}\} = \{(x_t^{ag}, y_t^{ag})\}$ be the iterates generated by Algorithm 2 with $V_X(x, x_t) = \|x - x_t\|^2/2$ and $V_Y(y, y_t) = \|y - y_t\|^2/2$. Assume that the parameters $\beta_t, \theta_t, \eta_t$ and τ_t satisfy* (2.15),

$$\theta_t = \frac{\eta_{t-1}}{\eta_t} = \frac{\tau_{t-1}}{\tau_t}, \tag{2.22}$$

$$\frac{1}{\eta_t} - \frac{L_G}{\beta_t} - \frac{L_K^2 \tau_t}{p} \geq 0, \tag{2.23}$$

for all $t \geq 1$ and for some $0 < p < 1$, then there exists a perturbation vector v_{t+1} such that

$$\tilde{g}(z_{t+1}^{ag}, v_{t+1}) \leq \frac{(2-p)D^2}{\beta_t \eta_t (1-p)} =: \varepsilon_{t+1} \tag{2.24}$$

for any $t \geq 1$. Moreover, we have

$$\|v_{t+1}\| \leq \frac{1}{\beta_t \eta_t}\|\hat{x} - x_1\| + \frac{1}{\beta_t \tau_t}\|\hat{y} - y_1\| + \left[\frac{1}{\beta_t \eta_t}\left(1 + \sqrt{\frac{\eta_1}{\tau_1(1-p)}}\right) + \frac{2L_K}{\beta_t}\right] D, \tag{2.25}$$

where $(\hat{x}, \hat{y})$ is a pair of solutions for problem (1.1) and

$$D := \sqrt{\|\hat{x} - x_1\|^2 + \frac{\eta_1}{\tau_1}\|\hat{y} - y_1\|^2}. \tag{2.26}$$

Below we suggest a specific parameter setting which satisfies (2.15), (2.22) and (2.23).

COROLLARY 2.4. *In Algorithm 2, if N is given and the parameters are set to*

$$\beta_t = \frac{t+1}{2},\ \theta_t = \frac{t-1}{t},\ \eta_t = \frac{t+1}{2(L_G + NL_K)}, \text{ and } \tau_t = \frac{t+1}{2NL_K} \tag{2.27}$$

then there exists v_N that satisfies (2.24) *with*

$$\varepsilon_N \leq \frac{10L_G\hat{D}^2}{N^2} + \frac{10L_K\hat{D}^2}{N} \quad \text{and} \quad \|v_N\| \leq \frac{15L_G\hat{D}}{N^2} + \frac{19L_K\hat{D}}{N}, \tag{2.28}$$

where $\hat{D} = \sqrt{\|\hat{x} - x_1\|^2 + \|\hat{y} - y_1\|^2}$.

Proof. For the parameters β_t, γ_t, η_t, τ_t in (2.27), *it is clear that* (2.15), (2.22) *holds. Furthermore, let $p = 1/4$, for any $t = 1, \ldots, N-1$, we have*

$$\tfrac{1}{\eta_t} - \tfrac{L_G}{\beta_t} - \tfrac{L_K^2\tau_t}{p} = \tfrac{2L_G + 2L_K N}{t+1} - \tfrac{2L_G}{t+1} - \tfrac{2L_K^2(t+1)}{L_K N} \geq \tfrac{2L_K N}{t+1} - \tfrac{2L_K(t+1)}{N} \geq 0,$$

thus (2.23) *holds. By Theorem 2.3, inequalities* (2.24) *and* (2.25) *hold. Noting that $\eta_t \leq \tau_t$, in* (2.24) *and* (2.25) *we have $D \leq \hat{D}$, $\|\hat{x} - x_1\| + \|\hat{y} - y_1\| \leq \sqrt{2}\hat{D}$, hence*

$$\|v_{t+1}\| \leq \tfrac{\sqrt{2}\hat{D}}{\beta_t\eta_t} + \tfrac{(1+\sqrt{4/3})\hat{D}}{\beta_t\eta_t} + \tfrac{2L_K\hat{D}}{\beta_t}$$

and

$$\varepsilon_{t+1} \leq \tfrac{(2-p)\hat{D}^2}{\beta_t\eta_t(1-p)} = \tfrac{7\hat{D}^2}{3\beta_t\eta_t}.$$

Also note that by (2.27), $\frac{1}{\beta_{N-1}\eta_{N-1}} = \frac{4(L_G + L_K N)}{N^2} = \frac{4L_G}{N^2} + \frac{4L_K}{N}$. Using these three relations and the definition of β_t in (2.27), we obtain (2.28) after simplifying the constants. □

It is interesting to notice that, if the parameters in Algorithm 2 are set to (2.27), then both residues ε_N and $\|v_N\|$ in (2.28) reduce to zero with approximately the same rate of convergence (up to a factor of $\hat{D}$). Also observe that in Theorem 2.3 and Corollary 2.4, we fix $V_X(\cdot,\cdot)$ and $V_Y(\cdot,\cdot)$ to be regular distance functions rather than more general Bregman divergences. This is due to fact that we need to apply the Triangular inequality associated with $\sqrt{V_X(\cdot,\cdot)}$ and $\sqrt{V_Y(\cdot,\cdot)}$, while such an inequality does not necessarily hold for Bregman divergences in general.

3. Stochastic APD method for stochastic SPP. Our goal in this section is to present a stochastic APD method for stochastic SPP (i.e., problem (1.1) with a stochastic oracle) and demonstrate that it can actually achieve the lower bound in (1.7) on the rate of convergence for stochastic SPP.

The stochastic APD method is a stochastic counterpart of the APD algorithm in Section 2, obtained by simply replacing the gradient operators $-K\bar{x}_t$, $\nabla G(x_t^{md})$ and $K^T y_{t+1}$, used in (2.5) and (2.6), with the stochastic gradient operators computed by the $\mathcal{SO}$, i.e., $-\hat{\mathcal{K}}_x(\bar{x}_t)$, $\hat{\mathcal{G}}(x_t^{md})$ an $\hat{\mathcal{K}}_y(y_{t+1})$, respectively. This algorithm is formally described as in Algorithm 3.

Algorithm 3 Stochastic APD method for stochastic SPP

Modify (2.5) and (2.6) in Algorithm 2 to

$$y_{t+1} = \operatorname*{argmin}_{y\in Y} \langle -\hat{\mathcal{K}}_x(\bar{x}_t), y\rangle + J(y) + \frac{1}{\tau_t} V_Y(y, y_t) \tag{3.1}$$

$$x_{t+1} = \operatorname*{argmin}_{x\in X} \langle \hat{\mathcal{G}}(x_t^{md}), x\rangle + \langle x, \hat{\mathcal{K}}_y(y_{t+1})\rangle + \frac{1}{\eta_t} V_X(x, x_t) \tag{3.2}$$

A few more remarks about the development of the above stochastic APD method are in order. Firstly, observe that, although primal-dual methods have been extensively studied for solving deterministic saddle-point problems, it seems that these types of methods have not yet been generalized for stochastic SPP in the literature. Secondly, as noted in Section 1, one possible way to solve stochastic SPP is to apply the AC-SA algorithm in [24] to a certain smooth approximation of (1.1) by Nesterov [41]. However, the rate of convergence of this approach will depend on the variance of the stochastic gradients computed for the smooth approximation problem, which is usually unknown and difficult to characterize. On the other hand, the stochastic APD method described above works directly with the original problem without requiring the application of the smoothing technique, and its rate of convergence will depend on the variance of the stochastic gradient operators computed for the original problem, i.e., $\sigma_{x,G}^2$, σ_y^2 and $\sigma_{x,K}^2$ in A1. We will show that it can achieve exactly the lower bound in (1.7) on the rate of convergence for stochastic SPP.

Similarly to Section 2, we use the two gap functions $g(\cdot)$ and $\tilde{g}(\cdot,\cdot)$, respectively, defined in (2.13) and (2.21) as the termination criteria for the stochastic APD algorithm, depending on whether the feasible set $Z = X \times Y$ is bounded or not. Since the algorithm is stochastic in nature, for both cases we establish its expected rate of convergence in terms of $g(\cdot)$ or $\tilde{g}(\cdot,\cdot)$, i.e., the "average" rate of convergence over many runs of the algorithm. In addition, we show that if Z is bounded, then the convergence of the APD algorithm can be strengthened under the following "light-tail" assumption on $\mathcal{SO}$.

A2. $\mathbb{E}\left[\exp\{\|\nabla G(x) - \hat{\mathcal{G}}(x)\|_*^2/\sigma_{x,G}^2\}\right] \le \exp\{1\}$, $\mathbb{E}\left[\exp\{\|Kx - \hat{\mathcal{K}}_x(x)\|_*^2/\sigma_y^2\}\right] \le \exp\{1\}$ and $\mathbb{E}\left[\exp\{\|K^T y - \hat{\mathcal{K}}_y(y)\|_*^2/\sigma_{x,K}^2\}\right] \le \exp\{1\}$.

It is easy to see that A2 implies A1 by Jensen's inequality.

Theorem 3.1 below summarizes the convergence properties of Algorithm 3 when Z is bounded. Note that the following quantity will be used in the statement of this result and the convergence analysis of the APD algorithms (see Section 4):

$$\gamma_t = \begin{cases} 1, & t = 1, \\ \theta_t^{-1}\gamma_{t-1}, & t \ge 2. \end{cases} \tag{3.3}$$

THEOREM 3.1. *Suppose that (2.14) holds for some $\Omega_X, \Omega_Y > 0$. Also assume that for all $t \ge 1$, the parameters $\beta_t, \theta_t, \eta_t$ and τ_t in Algorithm 3 satisfy* (2.15), (2.16), *and*

$$\frac{q\alpha_X}{\eta_t} - \frac{L_G}{\beta_t} - \frac{L_K^2 \tau_t}{p\alpha_Y} \ge 0 \tag{3.4}$$

for some $p, q \in (0,1)$. Then,

(a). Under assumption **A1**, *for all $t \geq 1$,*

$$\mathbb{E}[g(z_{t+1}^{ag})] \leq \mathcal{Q}_0(t), \tag{3.5}$$

where

$$\mathcal{Q}_0(t) := \tfrac{1}{\beta_t\gamma_t}\left\{\tfrac{2\gamma_t}{\eta_t}\Omega_X^2 + \tfrac{2\gamma_t}{\tau_t}\Omega_Y^2\right\} + \tfrac{1}{2\beta_t\gamma_t}\textstyle\sum_{i=1}^t\left\{\tfrac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\sigma_x^2 + \tfrac{(2-p)\tau_i\gamma_i}{(1-p)\alpha_Y}\sigma_y^2\right\}. \tag{3.6}$$

(b). Under assumption **A2**, *for all $\lambda > 0$ and $t \geq 1$,*

$$Prob\{g(z_{t+1}^{ag}) > \mathcal{Q}_0(t) + \lambda\mathcal{Q}_1(t)\} \leq 3\exp\{-\lambda^2/3\} + 3\exp\{-\lambda\}, \tag{3.7}$$

where

$$\mathcal{Q}_1(t) := \tfrac{1}{\beta_t\gamma_t}\left(\tfrac{\sqrt{2}\sigma_x\Omega_X}{\sqrt{\alpha_X}} + \tfrac{\sigma_y\Omega_Y}{\sqrt{\alpha_Y}}\right)\sqrt{2\textstyle\sum_{i=1}^t\gamma_i^2} + \tfrac{1}{2\beta_t\gamma_t}\textstyle\sum_{i=1}^t\left\{\tfrac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\sigma_x^2 + \tfrac{(2-p)\tau_i\gamma_i}{(1-p)\alpha_Y}\sigma_y^2\right\}. \tag{3.8}$$

We provide below a specific choice of the parameters β_t, θ_t, η_t and τ_t for the stochastic APD method for the case when Z is bounded.

COROLLARY 3.2. *Suppose that (2.14) holds and let D_X and D_Y be defined in Corollary 2.2. In Algorithm 3, if the parameters are set to*

$$\beta_t = \frac{t+1}{2},\ \theta_t = \frac{t-1}{t},\ \eta_t = \frac{2\alpha_X D_X t}{6L_G D_X + 3L_K D_Y t + 3\sigma_x t^{3/2}} \text{ and } \tau_t = \frac{2\alpha_Y D_Y}{3L_K D_X + 3\sigma_y\sqrt{t}}. \tag{3.9}$$

Then under Assumption **A1**, (3.5) *holds, and*

$$\mathcal{Q}_0(t) \leq \frac{6L_G D_X^2}{t(t+1)} + \frac{6L_K D_X D_Y}{t} + \frac{6(\sigma_x D_X + \sigma_y D_Y)}{\sqrt{t}}. \tag{3.10}$$

If in addition, Assumption **A2** *holds, then for all $\lambda > 0$,* (3.7) *holds, and*

$$\mathcal{Q}_1(t) \leq \frac{5\sigma_x D_X + 4\sigma_y D_Y}{\sqrt{t}}. \tag{3.11}$$

Proof. First we check that the parameters in (3.9) satisfy the conditions in Theorem 3.1. The inequalities (2.15) *and* (2.16) *can be checked easily. Furthermore, setting $p = q = 2/3$ we have for all t,*

$$\tfrac{q\alpha_X}{\eta_t} - \tfrac{L_G}{\beta_t} - \tfrac{L_K^2\tau_t}{p\alpha_Y} \geq \tfrac{2L_G D_X + L_K D_Y t}{D_X t} - \tfrac{2L_G}{t+1} - \tfrac{L_K^2 D_Y t}{L_K D_X t} \geq 0,$$

thus (3.4) *hold, and hence Theorem 3.1 holds. Now it suffice to show that* (3.10) *and* (3.11) *hold.*

Observe that by (3.3) *and* (3.9), *we have $\gamma_t = t$. Also, observe that $\sum_{i=1}^t\sqrt{i} \leq \int_1^{t+1}\sqrt{u}du \leq \frac{2}{3}(t+1)^{3/2} \leq \frac{2\sqrt{2}}{3}(t+1)\sqrt{t}$, thus*

$$\tfrac{1}{\gamma_t}\textstyle\sum_{i=1}^t\eta_i\gamma_i \leq \tfrac{2\alpha_X D_X}{3\sigma_x t}\sum_{i=1}^t\sqrt{i} \leq \tfrac{4\sqrt{2}\alpha_X D_X(t+1)\sqrt{t}}{9\sigma_x t} \text{ and } \tfrac{1}{\gamma_t}\textstyle\sum_{i=1}^t\tau_i\gamma_i \leq \tfrac{2\alpha_Y D_Y}{3\sigma_y t}\sum_{i=1}^t\sqrt{i} \leq \tfrac{4\sqrt{2}\alpha_Y D_Y(t+1)\sqrt{t}}{9\sigma_y t}.$$

Apply the above bounds to (3.6) *and* (3.8), *we get*

$$\begin{aligned}
\mathcal{Q}_0(t) \leq\ & \tfrac{2}{t+1}\Big(\tfrac{6L_G D_X + 3L_K D_Y t + 3\sigma_x t^{3/2}}{\alpha_X D_X t}\cdot\tfrac{\alpha_X}{2}D_X^2 + \tfrac{3L_K D_X + 3\sigma_y\sqrt{t}}{\alpha_Y D_Y}\cdot\tfrac{\alpha_Y}{2}D_Y^2 \\
& + \tfrac{2\sigma_x^2}{\alpha_X}\cdot\tfrac{4\sqrt{2}\alpha_X D_X(t+1)\sqrt{t}}{9\sigma_x t} + \tfrac{2\sigma_y^2}{\alpha_Y}\cdot\tfrac{4\sqrt{2}\alpha_Y D_Y(t+1)\sqrt{t}}{9\sigma_y t}\Big), \\
\mathcal{Q}_1(t) \leq\ & \tfrac{2}{t(t+1)}\left(\sigma_x D_X + \tfrac{\sigma_y D_Y}{\sqrt{2}}\right)\sqrt{\tfrac{2t(t+1)^2}{3}} + \tfrac{4\sigma_x^2}{\alpha_X(t+1)}\cdot\tfrac{4\sqrt{2}\alpha_X D_X(t+1)\sqrt{t}}{9\sigma_x t} + \tfrac{4\sigma_y^2}{\alpha_Y(t+1)}\cdot\tfrac{4\sqrt{2}\alpha_Y D_Y(t+1)\sqrt{t}}{9\sigma_y t}.
\end{aligned}$$

Simplifying the above inequalities, we see that (3.10) *and* (3.11) *hold.* □

Comparing the rate of convergence established in (3.10) with the lower bound in (1.7), we can clearly see that the stochastic APD algorithm is an optimal method for solving the stochastic saddle-point problems. More specifically, in view of (3.10), this algorithm allows us to have very large Lipschitz constants L_G (as big as $\mathcal{O}(N^{\frac{3}{2}})$) and L_K (as big as $\mathcal{O}(\sqrt{N})$) without significantly affecting its rate of convergence.

We now present the convergence results for the stochastic APD method applied to stochastic saddle-point problems with possibly unbounded feasible set Z. It appears that the solution methods of these types of problems have not been well-studied in the literature.

THEOREM 3.3. *Let* $\{z_t^{ag}\} = \{(x_t^{ag}, y_t^{ag})\}$ *be the iterates generated by Algorithm 2 with* $V_X(x, x_t) = \|x - x_t\|^2/2$ *and* $V_Y(y, y_t) = \|y - y_t\|^2/2$. *Assume that the parameters* $\beta_t, \theta_t, \eta_t$ *and* τ_t *in Algorithm 3 satisfy* (2.15), (2.22) *and (3.4) for all* $t \geq 1$ *and some* $p, q \in (0, 1)$, *then there exists a perturbation vector* v_{t+1} *such that*

$$\mathbb{E}[\tilde{g}(z_{t+1}^{ag}, v_{t+1})] \leq \frac{1}{\beta_t \eta_t}\left(\frac{6-4p}{1-p}D^2 + \frac{5-3p}{1-p}C^2\right) =: \varepsilon_{t+1} \tag{3.12}$$

for any $t \geq 1$. *Moreover, we have*

$$\mathbb{E}[\|v_{t+1}\|] \leq \frac{2\|\hat{x} - x_1\|}{\beta_t \eta_t} + \frac{2\|\hat{y} - y_1\|}{\beta_t \tau_t} + \sqrt{2D^2 + 2C^2}\left[\frac{2}{\beta_t \eta_t} + \frac{1}{\beta_t \tau_t}\sqrt{\frac{\tau_1}{\eta_1}}\left(\sqrt{\frac{1}{1-p}} + 1\right) + \frac{2L_K}{\beta_t}\right], \tag{3.13}$$

where $(\hat{x}, \hat{y})$ *is a pair of solutions for problem* (1.1), D *is defined in (2.26) and*

$$C := \sqrt{\sum_{i=1}^{t} \frac{\eta_i^2 \sigma_x^2}{1-q} + \sum_{i=1}^{t} \frac{\eta_i \tau_i \sigma_y^2}{1-p}}. \tag{3.14}$$

Below we specialize the results in Theorem 3.3 by choosing a set of parameters satisfying (2.15), (2.22) and (3.4).

COROLLARY 3.4. *In Algorithm 3, if* N *is given and the parameters are set to*

$$\beta_t = \frac{t+1}{2},\ \theta_t = \frac{t-1}{t},\ \eta_t = \frac{3t}{4\eta},\ \text{and } \tau_t = \frac{t}{\eta}, \tag{3.15}$$

where

$$\eta = 2L_G + 2L_K(N-1) + N\sqrt{N-1}\sigma/\tilde{D} \text{ for some } \tilde{D} > 0,\ \sigma = \sqrt{\frac{9}{4}\sigma_x^2 + \sigma_y^2}, \tag{3.16}$$

then there exists v_N *that satisfies* (3.12) *with*

$$\varepsilon_N \leq \frac{36L_G D^2}{N(N-1)} + \frac{36L_K D^2}{N} + \frac{\sigma D\left(18D/\tilde{D} + 6\tilde{D}/D\right)}{\sqrt{N-1}}, \tag{3.17}$$

$$\mathbb{E}[\|v_N\|] \leq \frac{50L_G D}{N(N-1)} + \frac{L_K D(55 + 4\tilde{D}/D)}{N} + \frac{\sigma(9 + 25D/\tilde{D})}{\sqrt{N-1}}, \tag{3.18}$$

where D *is defined in* (2.26).

Proof. For the parameters in (3.15), *it is clear that* (2.15) *and* (2.22) *hold. Furthermore, let* $p = 1/4$, $q = 3/4$, *then for all* $t = 1, \ldots, N-1$, *we have*

$$\tfrac{q}{\eta_t} - \tfrac{L_G}{\beta_t} - \tfrac{L_K^2 \tau_t}{p} = \tfrac{\eta}{t} - \tfrac{2L_G}{t+1} - \tfrac{4L_K^2 t}{\eta} \geq \tfrac{2L_G + 2L_K(N-1)}{t} - \tfrac{2L_G}{t} - \tfrac{2L_K^2 t}{L_K(N-1)} \geq 0,$$

thus (3.4) *holds. By Theorem 3.3, we get* (3.12) *and* (3.13). *Note that* $\eta_t/\tau_t = 3/4$, *and*

$$\tfrac{1}{\beta_{N-1}\eta_{N-1}}\|\hat{x}-x_1\| \le \tfrac{1}{\beta_{N-1}\eta_{N-1}}D,\ \ \tfrac{1}{\beta_{N-1}\tau_{N-1}}\|\hat{y}-y_1\| \le \tfrac{1}{\beta_{N-1}\eta_{N-1}}\cdot\tfrac{\eta_{N-1}}{\tau_{N-1}}\cdot\sqrt{\tfrac{4}{3}}D = \tfrac{\sqrt{3/4}D}{\beta_{N-1}\eta_{N-1}},$$

so in (3.12) *and* (3.13) *we have*

$$\varepsilon_N \le \frac{1}{\beta_{N-1}\eta_{N-1}}(\frac{20}{3}D^2+\frac{17}{3}C^2), \tag{3.19}$$

$$\mathbb{E}[\|v_N\|] \le \frac{(2+\sqrt{3})D}{\beta_{N-1}\eta_{N-1}} + \frac{\sqrt{2D^2+2C^2}\left(3+\sqrt{3/4}\right)}{\beta_{N-1}\eta_{N-1}} + \frac{2L_K\sqrt{2D^2+2C^2}}{\beta_{N-1}}. \tag{3.20}$$

By (3.14) *and the fact that* $\sum_{i=1}^{N-1} i^2 \le N^2(N-1)/3$, *we have*

$$C = \sqrt{\textstyle\sum_{i=1}^{N-1}\frac{9\sigma_x^2 i^2}{4\eta^2}+\sum_{i=1}^{N-1}\frac{\sigma_y^2 i^2}{\eta^2}} \le \sqrt{\tfrac{1}{3\eta^2}N^2(N-1)\left(\tfrac{9\sigma_x^2}{4}+\sigma_y^2\right)} = \tfrac{\sigma N\sqrt{N-1}}{\sqrt{3}\eta}$$

Applying the above bound to (3.19) *and* (3.20), *and using* (3.16) *and the fact that* $\sqrt{2D^2+C^2} \le \sqrt{2}D + C$, *we obtain*

$$\begin{aligned}
\varepsilon_N \le\ & \tfrac{8\eta}{3N(N-1)}\left(\tfrac{20}{3}D^2+\tfrac{17\sigma^2N^2(N-1)}{9\eta^2}\right) = \tfrac{8}{3N(N-1)}\left(\tfrac{20}{3}\eta D^2+\tfrac{17\sigma^2N^2(N-1)}{9\eta}\right)\\
\le\ & \tfrac{320L_GD^2}{9N(N-1)}+\tfrac{320L_K(N-1)D^2}{9N(N-1)}+\tfrac{160N\sqrt{N-1}\sigma D^2/\tilde{D}}{9N(N-1)}+\tfrac{136\sigma^2N^2(N-1)}{27N^2(N-1)^{3/2}\sigma/\tilde{D}}\\
\le\ & \tfrac{36L_GD^2}{N(N-1)}+\tfrac{36L_KD^2}{N}+\tfrac{\sigma D\left(18D/\tilde{D}+6\tilde{D}/D\right)}{\sqrt{N-1}},\\
\mathbb{E}[\|v_N\|] \le\ & \tfrac{1}{\beta_{N-1}\eta_{N-1}}\left(2D+\sqrt{3}D+3\sqrt{2}D+\sqrt{6}D/2+3\sqrt{2}C+\sqrt{6}C/2\right)+\tfrac{2\sqrt{2}L_KD}{\beta_{N-1}}+\tfrac{2\sqrt{2}L_KC}{\beta_{N-1}}\\
\le\ & \tfrac{16L_G+16L_K(N-1)+8N\sqrt{N-1}\sigma/\tilde{D}}{3N(N-1)}\left(2+\sqrt{3}+3\sqrt{2}+\sqrt{6}/2\right)D\\
& +\tfrac{8\sigma}{3\sqrt{N-1}}\left(\sqrt{6}+\sqrt{2}/2\right)+\tfrac{4\sqrt{2}L_KD}{N}+\tfrac{4\sqrt{2}L_K\sigma N\sqrt{N-1}}{N\sqrt{3}N\sqrt{N-1}\sigma/\tilde{D}}\\
\le\ & \tfrac{50L_GD}{N(N-1)}+\tfrac{L_KD(55+4\tilde{D}/D)}{N}+\tfrac{\sigma(9+25D/\tilde{D})}{\sqrt{N-1}}.
\end{aligned}$$

□

Observe that the parameter settings in (3.15)-(3.16) are more complicated than the ones in (2.27) for the deterministic unbounded case. In particular, for the stochastic unbounded case, we need to choose a parameter $\tilde{D}$ which is not required for the deterministic case. Clearly, the optimal selection for $\tilde{D}$ minimizing the RHS of (3.17) is given by $\sqrt{6}D$. Note however, that the value of D will be very difficult to estimate for the unbounded case and hence one often has to resort to a suboptimal selection for $\tilde{D}$. For example, if $\tilde{D}=1$, then the RHS of (3.17) and (3.18) will become $\mathcal{O}(L_GD^2/N^2+L_KD^2/N+\sigma D^2/\sqrt{N})$ and $\mathcal{O}(L_GD/N^2+L_KD/N+\sigma D/\sqrt{N})$, respectively.

4. Convergence analysis. Our goal in this section is to prove the main results presented in Section 2 and 3, namely, Theorems 2.1, 2.3, 3.1 and 3.3.

4.1. Convergence analysis for the deterministic APD algorithm. In this section, we prove Theorems 2.1 and 2.3 which, respectively, describe the convergence properties for the deterministic APD algorithm for the bounded and unbounded SPPs.

Before proving Theorem 2.1, we first prove two technical results: Proposition 4.1 shows some important properties for the function $Q(\cdot,\cdot)$ in (2.12) and Lemma 4.2 establishes a bound on $Q(x_t^{ag},z)$.

PROPOSITION 4.1. *Assume that* $\beta_t \ge 1$ *for all* t. *If* $z_{t+1}^{ag} = (x_{t+1}^{ag}, y_{t+1}^{ag})$ *is generated by Algorithm 2, then for all* $z=(x,y)\in Z$,

$$\begin{aligned}
&\beta_tQ(z_{t+1}^{ag},z)-(\beta_t-1)Q(z_t^{ag},z)\\
&\le \langle\nabla G(x_t^{md}),x_{t+1}-x\rangle+\frac{L_G}{2\beta_t}\|x_{t+1}-x_t\|^2+[J(y_{t+1})-J(y)]+\langle Kx_{t+1},y\rangle-\langle Kx,y_{t+1}\rangle.
\end{aligned} \tag{4.1}$$

Proof. By equations (2.4) *and* (2.7), $x_{t+1}^{ag} - x_t^{md} = \beta_t^{-1}(x_{t+1} - x_t)$. *Using this observation and the convexity of* $G(\cdot)$, *we have*

$$\begin{aligned}
& \beta_t G(x_{t+1}^{ag}) \le \ \beta_t G(x_t^{md}) + \beta_t \langle \nabla G(x_t^{md}), x_{t+1}^{ag} - x_t^{md} \rangle + \tfrac{\beta_t L_G}{2} \|x_{t+1}^{ag} - x_t^{md}\|^2 \\
= \ & \beta_t G(x_t^{md}) + \beta_t \langle \nabla G(x_t^{md}), x_{t+1}^{ag} - x_t^{md} \rangle + \tfrac{L_G}{2\beta_t} \|x_{t+1} - x_t\|^2 \\
= \ & \beta_t G(x_t^{md}) + (\beta_t - 1)\langle \nabla G(x_t^{md}), x_t^{ag} - x_t^{md} \rangle + \langle \nabla G(x_t^{md}), x_{t+1} - x_t^{md} \rangle + \tfrac{L_G}{2\beta_t} \|x_{t+1} - x_t\|^2 \\
= \ & (\beta_t - 1)\left[G(x_t^{md}) + \langle \nabla G(x_t^{md}), x_t^{ag} - x_t^{md} \rangle \right] + \left[G(x_t^{md}) + \langle \nabla G(x_t^{md}), x_{t+1} - x_t^{md} \rangle \right] + \tfrac{L_G}{2\beta_t} \|x_{t+1} - x_t\|^2 \\
= \ & (\beta_t - 1)\left[G(x_t^{md}) + \langle \nabla G(x_t^{md}), x_t^{ag} - x_t^{md} \rangle \right] + \left[G(x_t^{md}) + \langle \nabla G(x_t^{md}), x - x_t^{md} \rangle \right] + \langle \nabla G(x_t^{md}), x_{t+1} - x \rangle \\
& + \tfrac{L_G}{2\beta_t} \|x_{t+1} - x_t\|^2 \\
\le \ & (\beta_t - 1) G(x_t^{ag}) + G(x) + \langle \nabla G(x_t^{md}), x_{t+1} - x \rangle + \tfrac{L_G}{2\beta_t} \|x_{t+1} - x_t\|^2.
\end{aligned}$$

Moreover, by (2.8) *and the convexity of* $J(\cdot)$, *we have*

$$\beta_t J(y_{t+1}^{ag}) - \beta_t J(y) \le (\beta_t - 1) J(y_t^{ag}) + J(y_{t+1}) - \beta_t J(y) = \ (\beta_t - 1)\left[J(y_t^{ag}) - J(y) \right] + J(y_{t+1}) - J(y).$$

By (2.12), (2.7), (2.8) *and the above two inequalities above, we obtain*

$$\begin{aligned}
& \beta_t Q(z_{t+1}^{ag}, z) - (\beta_t - 1) Q(z_t^{ag}, z) \\
= \ & \beta_t \left\{ \left[G(x_{t+1}^{ag}) + \langle K x_{t+1}^{ag}, y \rangle - J(y) \right] - \left[G(x) + \langle K x, y_{t+1}^{ag} \rangle - J(y_{t+1}^{ag}) \right] \right\} \\
& - (\beta_t - 1) \left\{ [G(x_t^{ag}) + \langle K x_t^{ag}, y \rangle - J(y)] - [G(x) + \langle K x, y_t^{ag} \rangle - J(y_t^{ag})] \right\} \\
= \ & \beta_t G(x_{t+1}^{ag}) - (\beta_t - 1) G(x_t^{ag}) - G(x) + \beta_t \left[J(y_{t+1}^{ag}) - J(y) \right] \\
& - (\beta_t - 1)\left[J(y_t^{ag}) - J(y) \right] + \langle K(\beta_t x_{t+1}^{ag} - (\beta_t - 1) x_t^{ag}), y \rangle - \langle K x, \beta_t y_{t+1}^{ag} - (\beta_t - 1) y_t^{ag} \rangle \\
\le \ & \langle \nabla G(x_t^{md}), x_{t+1} - x \rangle + \tfrac{L_G}{2\beta_t} \|x_{t+1} - x_t\|^2 + J(y_{t+1}) - J(y) + \langle K x_{t+1}, y \rangle - \langle K x, y_{t+1} \rangle.
\end{aligned}$$

□

Lemma 4.2 establishes a bound for $Q(z_{t+1}^{ag}, z)$ for all $z \in Z$, which will be used in the proof of both Theorems 2.1 and 2.3.

LEMMA 4.2. *Let* $z_{t+1}^{ag} = (x_{t+1}^{ag}, y_{t+1}^{ag})$ *be the iterates generated by Algorithm 2. Assume that the parameters* $\beta_t, \theta_t, \eta_t$, *and* τ_t *satisfy* (2.15), (2.16) *and* (2.17). *Then, for any* $z \in Z$, *we have*

$$\beta_t \gamma_t Q(z_{t+1}^{ag}, z) \le \mathcal{B}_t(z, z_{[t]}) + \gamma_t \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle - \gamma_t \left(\frac{\alpha_X}{2\eta_t} - \frac{L_G}{2\beta_t} \right) \|x_{t+1} - x_t\|^2, \tag{4.2}$$

where γ_t *is defined in* (3.3), $z_{[t]} := \{(x_i, y_i)\}_{i=1}^{t+1}$ *and*

$$\mathcal{B}_t(z, z_{[t]}) := \sum_{i=1}^{t} \left\{ \frac{\gamma_i}{\eta_i} \left[V_X(x, x_i) - V_X(x, x_{i+1}) \right] + \frac{\gamma_i}{\tau_i} \left[V_Y(y, y_i) - V_Y(y, y_{i+1}) \right] \right\}. \tag{4.3}$$

Proof. First of all, we explore the optimality conditions in iterations (2.5) *and* (2.6). *Apply Lemma 2 in [15] to* (2.5), *we have*

$$\begin{aligned}
& \langle -K \bar{x}_t, y_{t+1} - y \rangle + J(y_{t+1}) - J(y) \le \frac{1}{\tau_t} V_Y(y, y_t) - \frac{1}{\tau_t} V_Y(y_{t+1}, y_t) - \frac{1}{\tau_t} V_Y(y, y_{t+1}) \\
\le \ & \frac{1}{\tau_t} V_Y(y, y_t) - \frac{\alpha_Y}{2\tau_t} \|y_{t+1} - y_t\|^2 - \frac{1}{\tau_t} V_Y(y, y_{t+1}),
\end{aligned} \tag{4.4}$$

where the last inequality follows from the fact that, by the strong convexity of $d_Y(\cdot)$ *and (2.11),*

$$V_Y(y_1, y_2) \ge \frac{\alpha_Y}{2} \|y_1 - y_2\|^2, \ \textit{for all } y_1, y_2 \in \mathcal{Y}. \tag{4.5}$$

Similarly, from (2.6) *we can derive that*

$$\langle \nabla G(x_t^{md}), x_{t+1} - x \rangle + \langle x_{t+1} - x, K^T y_{t+1} \rangle \le \frac{1}{\eta_t} V_X(x, x_t) - \frac{\alpha_X}{2\eta_t} \|x_{t+1} - x_t\|^2 - \frac{1}{\eta_t} V_X(x, x_{t+1}). \tag{4.6}$$

Our next step is to establish a crucial recursion of Algorithm 2. It follows from (4.1), (4.4) *and* (4.6) *that*

$$\begin{aligned}
&\beta_t Q(z_{t+1}^{ag}, z) - (\beta_t - 1) Q(z_t^{ag}, z) \\
\le\ & \langle \nabla G(x_t^{md}), x_{t+1} - x \rangle + \frac{L_G}{2\beta_t} \|x_{t+1} - x_t\|^2 + [J(y_{t+1}) - J(y)] + \langle K x_{t+1}, y \rangle - \langle K x, y_{t+1} \rangle \\
\le\ & \frac{1}{\eta_t} V_X(x, x_t) - \frac{1}{\eta_t} V_X(x, x_{t+1}) - \left(\frac{\alpha_X}{2\eta_t} - \frac{L_G}{2\beta_t} \right) \|x_{t+1} - x_t\|^2 \\
&+ \frac{1}{\tau_t} V_Y(y, y_t) - \frac{1}{\tau_t} V_Y(y, y_{t+1}) - \frac{\alpha_Y}{2\tau_t} \|y_{t+1} - y_t\|^2 \\
&- \langle x_{t+1} - x, K^T y_{t+1} \rangle + \langle K \bar{x}_t, y_{t+1} - y \rangle + \langle K x_{t+1}, y \rangle - \langle K x, y_{t+1} \rangle.
\end{aligned} \tag{4.7}$$

Also observe that by (2.9), *we have*

$$\begin{aligned}
&- \langle x_{t+1} - x, K^T y_{t+1} \rangle + \langle K \bar{x}_t, y_{t+1} - y \rangle + \langle K x_{t+1}, y \rangle - \langle K x, y_{t+1} \rangle \\
=\ & \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle - \theta_t \langle K(x_t - x_{t-1}), y - y_{t+1} \rangle \\
=\ & \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle - \theta_t \langle K(x_t - x_{t-1}), y - y_t \rangle - \theta_t \langle K(x_t - x_{t-1}), y_t - y_{t+1} \rangle.
\end{aligned}$$

Multiplying both sides of (4.7) by γ_t, *using the above identity and the fact that* $\gamma_t \theta_t = \gamma_{t-1}$ *due to (3.3), we obtain*

$$\begin{aligned}
&\beta_t \gamma_t Q(z_{t+1}^{ag}, z) - (\beta_t - 1) \gamma_t Q(z_t^{ag}, z) \\
\le\ & \frac{\gamma_t}{\eta_t} V_X(x, x_t) - \frac{\gamma_t}{\eta_t} V_X(x, x_{t+1}) + \frac{\gamma_t}{\tau_t} V_Y(y, y_t) - \frac{\gamma_t}{\tau_t} V_Y(y, y_{t+1}) \\
&+ \gamma_t \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle - \gamma_{t-1} \langle K(x_t - x_{t-1}), y - y_t \rangle \\
&- \gamma_t \left(\frac{\alpha_X}{2\eta_t} - \frac{L_G}{2\beta_t} \right) \|x_{t+1} - x_t\|^2 - \frac{\alpha_Y \gamma_t}{2\tau_t} \|y_{t+1} - y_t\|^2 - \gamma_{t-1} \langle K(x_t - x_{t-1}), y_t - y_{t+1} \rangle.
\end{aligned} \tag{4.8}$$

Now, applying Cauchy-Schwartz inequality to the last term in (4.8), *using the notation* L_K *in* (1.2) *and noticing that* $\gamma_{t-1}/\gamma_t = \theta_t \le \min\{\eta_{t-1}/\eta_t, \tau_{t-1}/\tau_t\}$ *from* (2.16), *we have*

$$\begin{aligned}
& -\gamma_{t-1} \langle K(x_t - x_{t-1}), y_t - y_{t+1} \rangle \le \gamma_{t-1} \|K(x_t - x_{t-1})\|_* \|y_t - y_{t+1}\| \\
\le\ & L_K \gamma_{t-1} \|x_t - x_{t-1}\| \, \|y_t - y_{t+1}\| \le \tfrac{L_K^2 \gamma_{t-1}^2 \tau_t}{2\alpha_Y \gamma_t} \|x_t - x_{t-1}\|^2 + \tfrac{\alpha_Y \gamma_t}{2\tau_t} \|y_t - y_{t+1}\|^2 \\
\le\ & \tfrac{L_K^2 \gamma_{t-1} \tau_{t-1}}{2\alpha_Y} \|x_t - x_{t-1}\|^2 + \tfrac{\alpha_Y \gamma_t}{2\tau_t} \|y_t - y_{t+1}\|^2.
\end{aligned}$$

Noting that $\theta_{t+1} = \gamma_t/\gamma_{t+1}$, *so by* (2.15) *we have* $(\beta_{t+1} - 1)\gamma_{t+1} = \beta_t \gamma_t$. *Combining the above two relations with inequality* (4.8), *we get the following recursion for Algorithm 2.*

$$\begin{aligned}
& (\beta_{t+1} - 1)\gamma_{t+1} Q(z_{t+1}^{ag}, z) - (\beta_t - 1)\gamma_t Q(z_t^{ag}, z) = \ \beta_t \gamma_t Q(z_{t+1}^{ag}, z) - (\beta_t - 1)\gamma_t Q(z_t^{ag}, z) \\
\le\ & \tfrac{\gamma_t}{\eta_t} V_X(x, x_t) - \tfrac{\gamma_t}{\eta_t} V_X(x, x_{t+1}) + \tfrac{\gamma_t}{\tau_t} V_Y(y, y_t) - \tfrac{\gamma_t}{\tau_t} V_Y(y, y_{t+1}) \\
& + \gamma_t \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle - \gamma_{t-1} \langle K(x_t - x_{t-1}), y - y_t \rangle \\
& - \gamma_t \left(\tfrac{\alpha_X}{2\eta_t} - \tfrac{L_G}{2\beta_t} \right) \|x_{t+1} - x_t\|^2 + \tfrac{L_K^2 \gamma_{t-1} \tau_{t-1}}{2\alpha_Y} \|x_t - x_{t-1}\|^2, \forall t \ge 1.
\end{aligned}$$

Applying the above inequality inductively and assuming that $x_0 = x_1$, *we conclude that*

$$\begin{aligned}
& (\beta_{t+1} - 1)\gamma_{t+1} Q(z_{t+1}^{ag}, z) - (\beta_1 - 1)\gamma_1 Q(z_1^{ag}, z) \le \mathcal{B}_t(z, z_{[t]}) + \gamma_t \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle \\
& - \gamma_t \left(\tfrac{\alpha_X}{2\eta_t} - \tfrac{L_G}{2\beta_t} \right) \|x_{t+1} - x_t\|^2 - \textstyle\sum_{i=1}^{t-1} \gamma_i \left(\tfrac{\alpha_X}{2\eta_i} - \tfrac{L_G}{2\beta_i} - \tfrac{L_K^2 \tau_i}{2\alpha_Y} \right) \|x_{i+1} - x_i\|^2,
\end{aligned}$$

which, in view of (2.17) *and the facts that* $\beta_1 = 1$ *and* $(\beta_{t+1} - 1)\gamma_{t+1} = \beta_t \gamma_t$ *by* (2.15), *implies* (4.2). □

We are now ready to prove Theorem 2.1, which follows as an immediate consequence of Lemma 4.2.

Proof of Theorem 2.1. Let $\mathcal{B}_t(z, z_{[t]})$ be defined in (4.3). First note that by the definition of γ_t in (3.3) and relation (2.16), we have $\theta_t = \gamma_{t-1}/\gamma_t \le \eta_{t-1}/\eta_t$ and hence $\gamma_{t-1}/\eta_{t-1} \le \gamma_t/\eta_t$. Using this observation and (2.14), we conclude that

$$\begin{aligned}
\mathcal{B}_t(z, z_{[t]}) = & \ \tfrac{\gamma_1}{\eta_1} V_X(x, x_1) - \textstyle\sum_{i=1}^{t-1} \left(\tfrac{\gamma_i}{\eta_i} - \tfrac{\gamma_{i+1}}{\eta_{i+1}} \right) V_X(x, x_{i+1}) - \tfrac{\gamma_t}{\eta_t} V_X(x, x_{t+1}) \\
& + \tfrac{\gamma_1}{\tau_1} V_Y(y, y_1) - \textstyle\sum_{i=1}^{t-1} \left(\tfrac{\gamma_i}{\tau_i} - \tfrac{\gamma_{i+1}}{\tau_{i+1}} \right) V_Y(y, y_{i+1}) - \tfrac{\gamma_t}{\tau_t} V_Y(y, y_{t+1}) \\
\le & \ \tfrac{\gamma_1}{\eta_1} \Omega_X^2 - \textstyle\sum_{i=1}^{t-1} \left(\tfrac{\gamma_i}{\eta_i} - \tfrac{\gamma_{i+1}}{\eta_{i+1}} \right) \Omega_X^2 - \tfrac{\gamma_t}{\eta_t} V_X(x, x_{t+1}) \\
& + \tfrac{\gamma_1}{\tau_1} \Omega_Y^2 - \textstyle\sum_{i=1}^{t-1} \left(\tfrac{\gamma_i}{\tau_i} - \tfrac{\gamma_{i+1}}{\tau_{i+1}} \right) \Omega_Y^2 - \tfrac{\gamma_t}{\tau_t} V_Y(y, y_{t+1}) \\
= & \ \tfrac{\gamma_t}{\eta_t} \Omega_X^2 - \tfrac{\gamma_t}{\eta_t} V_X(x, x_{t+1}) + \tfrac{\gamma_t}{\tau_t} \Omega_Y^2 - \tfrac{\gamma_t}{\tau_t} V_Y(y, y_{t+1}).
\end{aligned} \tag{4.9}$$

Now applying Cauchy-Schwartz inequality to the inner product term in (4.2), we get

$$\gamma_t \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle \le L_K \gamma_t \|x_{t+1} - x_t\| \|y - y_{t+1}\| \le \frac{L_K^2 \gamma_t \tau_t}{2\alpha_Y} \|x_{t+1} - x_t\|^2 + \frac{\alpha_Y \gamma_t}{2\tau_t} \|y - y_{t+1}\|^2. \tag{4.10}$$

Using the above two relations, (2.17), (4.2) and (4.5), we have

$$\begin{aligned}
\beta_t \gamma_t Q(z_{t+1}^{ag}, z) \le & \ \tfrac{\gamma_t}{\eta_t} \Omega_X^2 - \tfrac{\gamma_t}{\eta_t} V_X(x, x_{t+1}) + \tfrac{\gamma_t}{\tau_t} \Omega_Y^2 - \tfrac{\gamma_t}{\tau_t} \left(V_Y(y, y_{t+1}) - \tfrac{\alpha_Y}{2} \|y - y_{t+1}\|^2 \right) \\
-\gamma_t \left(\tfrac{\alpha_X}{2\eta_t} - \tfrac{L_G}{2\beta_t} - \tfrac{L_K^2 \tau_t}{2\alpha_Y} \right) \|x_{t+1} - x_t\|^2 \le & \ \tfrac{\gamma_t}{\eta_t} \Omega_X^2 + \tfrac{\gamma_t}{\tau_t} \Omega_Y^2, \quad \forall z \in Z,
\end{aligned}$$

which together with (2.13), then clearly imply (2.18). □

Our goal in the remaining part of this subsection is to prove Theorem 2.3, which summarizes the convergence properties of Algorithm 2 when X or Y is unbounded. We will first prove a technical result which specializes the results in Lemma 4.2 for the case when (2.15), (2.22) and (2.23) hold.

LEMMA 4.3. *Let $\hat{z} = (\hat{x}, \hat{y}) \in Z$ be a saddle point of* (1.1). *If $V_X(x, x_t) = \|x - x_t\|^2/2$ and $V_Y(y, y_t) = \|y - y_t\|^2/2$ in Algorithm 2, and the parameters $\beta_t, \theta_t, \eta_t$ and τ_t satisfy* (2.15), (2.22) *and* (2.23), *then*

(a).
$$\|\hat{x} - x_{t+1}\|^2 + \frac{\eta_t(1-p)}{\tau_t} \|\hat{y} - y_{t+1}\|^2 \le \|\hat{x} - x_1\|^2 + \frac{\eta_t}{\tau_t} \|\hat{y} - y_1\|^2, \ \textit{for all } t \ge 1. \tag{4.11}$$

(b).
$$\tilde{g}(z_{t+1}^{ag}, v_{t+1}) \le \frac{1}{2\beta_t \eta_t} \|x_{t+1}^{ag} - x_1\|^2 + \frac{1}{2\beta_t \tau_t} \|y_{t+1}^{ag} - y_1\|^2 =: \delta_{t+1}, \ \textit{for all } t \ge 1, \tag{4.12}$$

where $\tilde{g}(\cdot, \cdot)$ is defined in (2.21) and

$$v_{t+1} = \left(\frac{1}{\beta_t \eta_t}(x_1 - x_{t+1}), \frac{1}{\beta_t \tau_t}(y_1 - y_{t+1}) - \frac{1}{\beta_t} K(x_{t+1} - x_t) \right). \tag{4.13}$$

Proof. It is easy to check that the conditions in Lemma 4.2 are satisfied. By (2.22), (4.2) *in Lemma 4.2 becomes*

$$\begin{aligned}
\beta_t Q(z_{t+1}^{ag}, z) \le & \ \frac{1}{2\eta_t} \|x - x_1\|^2 - \frac{1}{2\eta_t} \|x - x_{t+1}\|^2 + \frac{1}{2\tau_t} \|y - y_1\|^2 - \frac{1}{2\tau_t} \|y - y_{t+1}\|^2 \\
& + \langle K(x_{t+1} - x_t), y - y_{t+1} \rangle - \left(\frac{1}{2\eta_t} - \frac{L_G}{2\beta_t} \right) \|x_{t+1} - x_t\|^2.
\end{aligned} \tag{4.14}$$

To prove (4.11), *observe that*

$$\langle K(x_{t+1} - x_t), y - y_{t+1} \rangle \le \frac{L_K^2 \tau_t}{2p} \|x_{t+1} - x_t\|^2 + \frac{p}{2\tau_t} \|y - y_{t+1}\|^2 \tag{4.15}$$

where p is the constant in (2.23). *By* (2.23) *and the above two inequalities, we get*

$$\beta_t Q(z_{t+1}^{ag}, z) \le \tfrac{1}{2\eta_t}\|x - x_1\|^2 - \tfrac{1}{2\eta_t}\|x - x_{t+1}\|^2 + \tfrac{1}{2\tau_t}\|y - y_1\|^2 - \tfrac{1-p}{2\tau_t}\|y - y_{t+1}\|^2.$$

Letting $z = \hat{z}$ in the above, and using the fact that $Q(z_{t+1}^{ag}, \hat{z}) \ge 0$, we obtain (4.11).

Now we prove (4.12). *Noting that*

$$\begin{aligned}
&\|x - x_1\|^2 - \|x - x_{t+1}\|^2 = 2\langle x_{t+1} - x_1, x\rangle + \|x_1\|^2 - \|x_{t+1}\|^2 \\
=&2\langle x_{t+1} - x_1, x - x_{t+1}^{ag}\rangle + 2\langle x_{t+1} - x_1, x_{t+1}^{ag}\rangle + \|x_1\|^2 - \|x_{t+1}\|^2 \\
=&2\langle x_{t+1} - x_1, x - x_{t+1}^{ag}\rangle + \|x_{t+1}^{ag} - x_1\|^2 - \|x_{t+1}^{ag} - x_{t+1}\|^2,
\end{aligned} \tag{4.16}$$

we conclude from (2.23) *and* (4.14) *that for any $z \in Z$,*

$$\begin{aligned}
& \beta_t Q(z_{t+1}^{ag}, z) + \langle K(x_{t+1} - x_t), y_{t+1}^{ag} - y\rangle - \tfrac{1}{\eta_t}\langle x_1 - x_{t+1}, x_{t+1}^{ag} - x\rangle - \tfrac{1}{\tau_t}\langle y_1 - y_{t+1}, y_{t+1}^{ag} - y\rangle \\
\le\ & \tfrac{1}{2\eta_t}\left(\|x_{t+1}^{ag} - x_1\|^2 - \|x_{t+1}^{ag} - x_{t+1}\|^2\right) + \tfrac{1}{2\tau_t}\left(\|y_{t+1}^{ag} - y_1\|^2 - \|y_{t+1}^{ag} - y_{t+1}\|^2\right) \\
& + \langle K(x_{t+1} - x_t), y_{t+1}^{ag} - y_{t+1}\rangle - \left(\tfrac{1}{2\eta_t} - \tfrac{L_G}{2\beta_t}\right)\|x_{t+1} - x_t\|^2 \\
\le\ & \tfrac{1}{2\eta_t}\left(\|x_{t+1}^{ag} - x_1\|^2 - \|x_{t+1}^{ag} - x_{t+1}\|^2\right) + \tfrac{1}{2\tau_t}\left(\|y_{t+1}^{ag} - y_1\|^2 - \|y_{t+1}^{ag} - y_{t+1}\|^2\right) \\
& + \tfrac{p}{2\tau_t}\|y_{t+1}^{ag} - y_{t+1}\|^2 - \left(\tfrac{1}{2\eta_t} - \tfrac{L_G}{2\beta_t} - \tfrac{L_K^2\tau_t}{2p}\right)\|x_{t+1} - x_t\|^2 \\
\le\ & \tfrac{1}{2\eta_t}\|x_{t+1}^{ag} - x_1\|^2 + \tfrac{1}{2\tau_t}\|y_{t+1}^{ag} - y_1\|^2.
\end{aligned}$$

The result in (4.12) *and* (4.13) *immediately follows from the above inequality and (2.21).*
□

We are now ready to prove Theorem 2.3.

Proof. Proof of Theorem 2.3. We have established the expression of v_{t+1} and δ_{t+1} in Lemma 4.3. It suffices to estimate the bound on $\|v_{t+1}\|$ and δ_{t+1}. It follows from the definition of D, (2.22) and (4.11) that for all $t \ge 1$, $\|\hat{x} - x_{t+1}\| \le D$ and $\|\hat{y} - y_{t+1}\| \le D\sqrt{\frac{\tau_1}{\eta_1(1-p)}}$. Now by (4.13), we have

$$\begin{aligned}
\|v_{t+1}\| &\le \tfrac{1}{\beta_t\eta_t}\|x_1 - x_{t+1}\| + \tfrac{1}{\beta_t\tau_t}\|y_1 - y_{t+1}\| + \tfrac{L_K}{\beta_t}\|x_{t+1} - x_t\| \\
&\le \tfrac{1}{\beta_t\eta_t}\left(\|\hat{x} - x_1\| + \|\hat{x} - x_{t+1}\|\right) + \tfrac{1}{\beta_t\tau_t}\left(\|\hat{y} - y_1\| + \|\hat{y} - y_{t+1}\|\right) + \tfrac{L_K}{\beta_t}\left(\|\hat{x} - x_{t+1}\| + \|\hat{x} - x_t\|\right) \\
&\le \tfrac{1}{\beta_t\eta_t}\left(\|\hat{x} - x_1\| + D\right) + \tfrac{1}{\beta_t\tau_t}\left(\|\hat{y} - y_1\| + D\sqrt{\tfrac{\tau_1}{\eta_1(1-p)}}\right) + \tfrac{2L_K}{\beta_t}D \\
&= \tfrac{1}{\beta_t\eta_t}\|\hat{x} - x_1\| + \tfrac{1}{\beta_t\tau_t}\|\hat{y} - y_1\| + D\left[\tfrac{1}{\beta_t\eta_t}\left(1 + \sqrt{\tfrac{\eta_1}{\tau_1(1-p)}}\right) + \tfrac{2L_K}{\beta_t}\right].
\end{aligned}$$

To estimate the bound of δ_{t+1}, consider the sequence $\{\gamma_t\}$ defined in (3.3). Using the fact that $(\beta_{t+1} - 1)\gamma_{t+1} = \beta_t\gamma_t$ due to (2.15) and (3.3), and applying (2.7) and (2.8) inductively, we have

$$x_{t+1}^{ag} = \frac{1}{\beta_t\gamma_t}\sum_{i=1}^{t}\gamma_i x_{i+1},\ y_{t+1}^{ag} = \frac{1}{\beta_t\gamma_t}\sum_{i=1}^{t}\gamma_i y_{i+1} \text{ and } \frac{1}{\beta_t\gamma_t}\sum_{i=1}^{t}\gamma_i = 1. \tag{4.17}$$

Thus x_{t+1}^{ag} and y_{t+1}^{ag} are convex combinations of sequences $\{x_{i+1}\}_{i=1}^t$ and $\{y_{i+1}\}_{i=1}^t$. Using these relations and (4.11), we have

$$\begin{aligned}
\delta_{t+1} &= \tfrac{1}{2\beta_t\eta_t}\|x_{t+1}^{ag} - x_1\|^2 + \tfrac{1}{2\beta_t\tau_t}\|y_{t+1}^{ag} - y_1\|^2 \le \tfrac{1}{\beta_t\eta_t}\left(\|\hat{x} - x_{t+1}^{ag}\|^2 + \|\hat{x} - x_1\|^2\right) + \tfrac{1}{\beta_t\tau_t}\left(\|\hat{y} - y_{t+1}^{ag}\|^2 + \|\hat{y} - y_1\|^2\right) \\
&= \tfrac{1}{\beta_t\eta_t}\left(D^2 + \|\hat{x} - x_{t+1}^{ag}\|^2 + \tfrac{\eta_t(1-p)}{\tau_t}\|\hat{y} - y_{t+1}^{ag}\|^2 + \tfrac{\eta_t p}{\tau_t}\|\hat{y} - y_{t+1}^{ag}\|^2\right) \\
&\le \tfrac{1}{\beta_t\eta_t}\left[D^2 + \tfrac{1}{\beta_t\gamma_t}\textstyle\sum_{i=1}^t\gamma_i\left(\|\hat{x} - x_{i+1}\|^2 + \tfrac{\eta_t(1-p)}{\tau_t}\|\hat{y} - y_{i+1}\|^2 + \tfrac{\eta_t p}{\tau_t}\|\hat{y} - y_{i+1}\|^2\right)\right] \\
&\le \tfrac{1}{\beta_t\eta_t}\left[D^2 + \tfrac{1}{\beta_t\gamma_t}\textstyle\sum_{i=1}^t\gamma_i\left(D^2 + \tfrac{\eta_t p}{\tau_t}\cdot\tfrac{\tau_1}{\eta_1(1-p)}D^2\right)\right] = \tfrac{(2-p)D^2}{\beta_t\eta_t(1-p)}.
\end{aligned}$$

□

4.2. Convergence analysis for the stochastic APD algorithm. In this subsection, we prove Theorems 3.1 and 3.3 which describe the convergence properties of the stochastic APD algorithm presented in Section 3.

Let $\hat{\mathcal{G}}(x_t^{md})$, $\hat{\mathcal{K}}_x(\bar{x}_t)$ and $\hat{\mathcal{K}}_y(y_{t+1})$ be the output from the $\mathcal{SO}$ at the t-th iteration of Algorithm 3. Throughout this subsection, we denote

$$\Delta_{x,G}^t := \hat{\mathcal{G}}(x_t^{md}) - \nabla G(x_t^{md}),\ \Delta_{x,K}^t := \hat{\mathcal{K}}_y(y_{t+1}) - K^T y_{t+1},\ \Delta_y^t := -\hat{\mathcal{K}}_x(\bar{x}_t) + K\bar{x}_t,$$
$$\Delta_x^t := \Delta_{x,G}^t + \Delta_{x,K}^t \ \text{ and } \ \Delta^t := (\Delta_x^t, \Delta_y^t).$$

Moreover, for a given $z = (x, y) \in Z$, let us denote $\|z\|^2 = \|x\|^2 + \|y\|^2$ and its associate dual norm for $\Delta = (\Delta_x, \Delta_y)$ by $\|\Delta\|_*^2 = \|\Delta_x\|_*^2 + \|\Delta_y\|_*^2$. We also define the Bregman divergence $V(z, \tilde{z}) := V_X(x, \tilde{x}) + V_Y(y, \tilde{y})$ for $z = (x, y)$ and $\tilde{z} = (\tilde{x}, \tilde{y})$.

Before proving Theorem 3.1, we first estimate a bound on $Q(z_{t+1}^{ag}, z)$ for all $z \in Z$. This result is analogous to Lemma 4.2 for the deterministic APD method.

LEMMA 4.4. *Let $z_t^{ag} = (x_t^{ag}, y_t^{ag})$ be the iterates generated by Algorithm 3. Assume that the parameters $\beta_t, \theta_t, \eta_t$ and τ_t satisfy (2.15), (2.16) and (3.4). Then, for any $z \in Z$, we have*

$$\beta_t\gamma_t Q(z_{t+1}^{ag}, z) \le \mathcal{B}_t(z, z_{[t]}) + \gamma_t\langle K(x_{t+1} - x_t), y - y_{t+1}\rangle - \gamma_t\left(\frac{q\alpha_X}{2\eta_t} - \frac{L_G}{2\beta_t}\right)\|x_{t+1} - x_t\|^2 + \sum_{i=1}^{t}\Lambda_i(z), \tag{4.18}$$

where γ_t and $\mathcal{B}_t(z, z_{[t]})$, respectively, are defined in (3.3) and (4.3), $z_{[t]} = \{(x_i, y_i)\}_{i=1}^{t+1}$ and

$$\Lambda_i(z) := -\frac{(1-q)\alpha_X\gamma_i}{2\eta_i}\|x_{i+1} - x_i\|^2 - \frac{(1-p)\alpha_Y\gamma_i}{2\tau_i}\|y_{i+1} - y_i\|^2 - \gamma_i\langle\Delta^i, z_{i+1} - z\rangle. \tag{4.19}$$

Proof. Similar to (4.4) *and* (4.6)*, we conclude from the optimality conditions of* (3.1) *and* (3.2) *that*

$$\begin{aligned}\langle -\hat{\mathcal{K}}_x(\bar{x}_t), y_{t+1} - y\rangle + J(y_{t+1}) - J(y) \le\ & \tfrac{1}{\tau_t}V_Y(y, y_t) - \tfrac{\alpha_Y}{2\tau_t}\|y_{t+1} - y_t\|^2 - \tfrac{1}{\tau_t}V_Y(y, y_{t+1}),\\ \langle\hat{\mathcal{G}}(x_t^{md}), x_{t+1} - x\rangle + \langle x_{t+1} - x, \hat{\mathcal{K}}_y(y_{t+1})\rangle \le\ & \tfrac{1}{\eta_t}V_X(x, x_t) - \tfrac{\alpha_X}{2\eta_t}\|x_{t+1} - x_t\|^2 - \tfrac{1}{\eta_t}V_X(x, x_{t+1}).\end{aligned}$$

Now we establish an important recursion for Algorithm 3. Observing that Proposition 4.1 also holds for Algorithm 3, and applying the above two inequalities to (4.1) *in Proposition 4.1, similar to* (4.8)*, we have*

$$\begin{aligned}&\beta_t\gamma_t Q(z_{t+1}^{ag}, z) - (\beta_t - 1)\gamma_t Q(z_t^{ag}, z)\\ \le\ & \frac{\gamma_t}{\eta_t}V_X(x, x_t) - \frac{\gamma_t}{\eta_t}V_X(x, x_{t+1}) + \frac{\gamma_t}{\tau_t}V_Y(y, y_t) - \frac{\gamma_t}{\tau_t}V_Y(y, y_{t+1})\\ &+ \gamma_t\langle K(x_{t+1} - x_t), y - y_{t+1}\rangle - \gamma_{t-1}\langle K(x_t - x_{t-1}), y - y_t\rangle\\ &- \gamma_t\left(\frac{\alpha_X}{2\eta_t} - \frac{L_G}{2\beta_t}\right)\|x_{t+1} - x_t\|^2 - \frac{\alpha_Y\gamma_t}{2\tau_t}\|y_{t+1} - y_t\|^2 - \gamma_{t-1}\langle K(x_t - x_{t-1}), y_t - y_{t+1}\rangle\\ &- \gamma_t\langle\Delta_{x,G}^t + \Delta_{x,K}^t, x_{t+1} - x\rangle - \gamma_t\langle\Delta_y^t, y_{t+1} - y\rangle, \quad \forall z \in Z.\end{aligned} \tag{4.20}$$

By Cauchy-Schwartz inequality and (2.16)*, for all $p \in (0, 1)$,*

$$\begin{aligned}&-\gamma_{t-1}\langle K(x_t - x_{t-1}), y_t - y_{t+1}\rangle \le \gamma_{t-1}\|K(x_t - x_{t-1})\|_*\|y_t - y_{t+1}\|\\ \le\ & L_K\gamma_{t-1}\|x_t - x_{t-1}\|\|y_t - y_{t+1}\| \le \frac{L_K^2\gamma_{t-1}^2\tau_t}{2p\alpha_Y\gamma_t}\|x_t - x_{t-1}\|^2 + \frac{p\alpha_Y\gamma_t}{2\tau_t}\|y_t - y_{t+1}\|^2\\ \le\ & \frac{L_K^2\gamma_{t-1}\tau_{t-1}}{2p\alpha_Y}\|x_t - x_{t-1}\|^2 + \frac{p\alpha_Y\gamma_t}{2\tau_t}\|y_t - y_{t+1}\|^2.\end{aligned} \tag{4.21}$$

By (2.15), (4.19), (4.20) *and* (4.21), *we can develop the following recursion for Algorithm 3:*

$$\begin{aligned}
&(\beta_{t+1}-1)\gamma_{t+1}Q(z_{t+1}^{ag},z)-(\beta_t-1)\gamma_tQ(z_t^{ag},z)= \ \beta_t\gamma_tQ(z_{t+1}^{ag},z)-(\beta_t-1)\gamma_tQ(z_t^{ag},z)\\
\leq\ & \tfrac{\gamma_t}{\eta_t}V_X(x,x_t)-\tfrac{\gamma_t}{\eta_t}V_X(x,x_{t+1})+\tfrac{\gamma_t}{\tau_t}V_Y(y,y_t)-\tfrac{\gamma_t}{\tau_t}V_Y(y,y_{t+1})\\
&+\gamma_t\langle K(x_{t+1}-x_t),y-y_{t+1}\rangle-\gamma_{t-1}\langle K(x_t-x_{t-1}),y-y_t\rangle\\
&-\gamma_t\left(\tfrac{q\alpha_X}{2\eta_t}-\tfrac{L_G}{2\beta_t}\right)\|x_{t+1}-x_t\|^2+\tfrac{L_K^2\gamma_{t-1}\tau_{t-1}}{2p\alpha_Y}\|x_t-x_{t-1}\|^2+\Lambda_t(x),\quad\forall z\in Z.
\end{aligned}$$

Applying the above inequality inductively and assuming that $x_0=x_1$, *we obtain*

$$\begin{aligned}
&(\beta_{t+1}-1)\gamma_{t+1}Q(z_{t+1}^{ag},z)-(\beta_1-1)\gamma_1Q(z_1^{ag},z)\\
\leq\ & \mathcal{B}_t(z,z_{[t]})+\gamma_t\langle K(x_{t+1}-x_t),y-y_{t+1}\rangle-\gamma_t\left(\tfrac{q\alpha_X}{2\eta_t}-\tfrac{L_G}{2\beta_t}\right)\|x_{t+1}-x_t\|^2\\
&-\textstyle\sum_{i=1}^{t-1}\gamma_i\left(\tfrac{q\alpha_X}{2\eta_i}-\tfrac{L_G}{2\beta_i}-\tfrac{L_K^2\tau_i}{2p\alpha_Y}\right)\|x_{i+1}-x_i\|^2+\textstyle\sum_{i=1}^t\Lambda_i(x),\quad\forall z\in Z.
\end{aligned}$$

Relation (4.18) *then follows immediately from the above inequality,* (2.15) *and* (3.4). □

We also need the following technical result whose proof is based on Lemma 2.1 of [35].

LEMMA 4.5. *Let* η_i,τ_i *and* γ_i, $i=1,2,\ldots$, *be given positive constants. For any* $z_1\in Z$, *if we define* $z_1^v=z_1$ *and*

$$z_{i+1}^v=\operatorname*{argmin}_{z=(x,y)\in Z}\left\{-\eta_i\langle\Delta_x^i,x\rangle-\tau_i\langle\Delta_y^i,y\rangle+V(z,z_i^v)\right\},\tag{4.22}$$

then

$$\sum_{i=1}^t\gamma_i\langle-\Delta^i,z_i^v-z\rangle\leq\mathcal{B}_t(z,z_{[t]}^v)+\sum_{i=1}^t\frac{\eta_i\gamma_i}{2\alpha_X}\|\Delta_x^i\|_*^2+\sum_{i=1}^t\frac{\tau_i\gamma_i}{2\alpha_Y}\|\Delta_y^i\|_*^2,\tag{4.23}$$

where $z_{[t]}^v:=\{z_i^v\}_{i=1}^t$ *and* $\mathcal{B}_t(z,z_{[t]}^v)$ *is defined in (4.3).*

Proof. Noting that (4.22) *implies* $z_{i+1}^v=(x_{i+1}^v,y_{i+1}^v)$ *where* $x_{i+1}^v=\operatorname*{argmin}_{x=\in X}\left\{-\eta_i\langle\Delta_x^i,x\rangle+V_X(x,x_i^v)\right\}$ *and* $y_{i+1}^v=\operatorname*{argmin}_{y\in Y}\left\{-\tau_i\langle\Delta_y^i,y\rangle+V(y,y_i^v)\right\}$, *from Lemma 2.1 of [35] we have*

$V_X(x,x_{i+1}^v)\leq V_X(x,x_i^v)-\eta_i\langle\Delta_x^i,x-x_i^v\rangle+\frac{\eta_i^2\|\Delta_x^i\|_*^2}{2\alpha_X}$, *and* $V_Y(y,y_{i+1}^v)\leq V_Y(y,y_i^v)-\tau_i\langle\Delta_y^i,y-y_i^v\rangle+\frac{\tau_i^2\|\Delta_y^i\|_*^2}{2\alpha_Y}$ *for all* $i\geq1$. *Thus*

$$\begin{aligned}
\frac{\gamma_i}{\eta_i}V_X(x,x_{i+1}^v)\leq&\frac{\gamma_i}{\eta_i}V_X(x,x_i^v)-\gamma_i\langle\Delta_x^i,x-x_i^v\rangle+\frac{\gamma_i\eta_i\|\Delta_x^i\|_*^2}{2\alpha_X},\ \text{and}\\
\frac{\gamma_i}{\tau_i}V_Y(y,y_{i+1}^v)\leq&\frac{\gamma_i}{\tau_i}V_Y(y,y_i^v)-\gamma_i\langle\Delta_y^i,y-y_i^v\rangle+\frac{\gamma_i\tau_i\|\Delta_y^i\|_*^2}{2\alpha_Y}.
\end{aligned}$$

Adding the above two inequalities together, and summing up them from $i=1$ *to* t *we get*

$$0\leq\mathcal{B}_t(z,z_{[t]}^v)-\textstyle\sum_{i=1}^t\gamma_i\langle\Delta^i,z-z_i^v\rangle+\sum_{i=1}^t\frac{\gamma_i\eta_i\|\Delta_x^i\|_*^2}{2\alpha_X}+\sum_{i=1}^t\frac{\gamma_i\tau_i\|\Delta_y^i\|_*^2}{2\alpha_Y},$$

so (4.23) *holds.* □

We are now ready to prove Theorem 3.1.

Proof of Theorem 3.1 Firstly, applying the bounds in (4.9) and (4.10) to (4.18), we get

$$\begin{aligned}
\beta_t\gamma_tQ(z_{t+1}^{ag},z)\leq&\frac{\gamma_t}{\eta_t}\Omega_X^2-\frac{\gamma_t}{\eta_t}V_X(x,x_{t+1})+\frac{\gamma_t}{\tau_t}\Omega_Y^2-\frac{\gamma_t}{\tau_t}V_Y(y,y_{t+1})+\frac{\alpha_Y\gamma_t}{2\tau_t}\|y-y_{t+1}\|^2\\
&-\gamma_t\left(\frac{q\alpha_X}{2\eta_t}-\frac{L_G}{2\beta_t}-\frac{L_K^2\tau_t}{2\alpha_Y}\right)\|x_{t+1}-x_t\|^2+\sum_{i=1}^t\Lambda_i(z)\\
\leq&\frac{\gamma_t}{\eta_t}\Omega_X^2+\frac{\gamma_t}{\tau_t}\Omega_Y^2+\sum_{i=1}^t\Lambda_i(z),\quad\forall\,z\in Z.
\end{aligned}\tag{4.24}$$

By (4.19), we have

$$\begin{aligned}\Lambda_i(z) &= -\frac{(1-q)\alpha_X\gamma_i}{2\eta_i}\|x_{i+1}-x_i\|^2 - \frac{(1-p)\alpha_Y\gamma_i}{2\tau_i}\|y_{i+1}-y_i\|^2 + \gamma_i\langle\Delta^i, z-z_{i+1}\rangle \\ &= -\frac{(1-q)\alpha_X\gamma_i}{2\eta_i}\|x_{i+1}-x_i\|^2 - \frac{(1-p)\alpha_Y\gamma_i}{2\tau_i}\|y_{i+1}-y_i\|^2 + \gamma_i\langle\Delta^i, z_i-z_{i+1}\rangle + \gamma_i\langle\Delta^i, z-z_i\rangle \\ &\le \frac{\eta_i\gamma_i}{2(1-q)\alpha_X}\|\Delta_x^i\|_*^2 + \frac{\tau_i\gamma_i}{2(1-p)\alpha_Y}\|\Delta_y^i\|_*^2 + \gamma_i\langle\Delta^i, z-z_i\rangle,\end{aligned} \tag{4.25}$$

where the last relation follows from Young's inequality. For all $i \ge 1$, letting $z_1^v = z_1$, and z_{i+1}^v as in (4.22), we conclude from (4.25) and Lemma 4.5 that, $\forall z \in Z$,

$$\begin{aligned}\sum_{i=1}^t \Lambda_i(z) &\le \sum_{i=1}^t \left\{ \frac{\eta_i\gamma_i}{2(1-q)\alpha_X}\|\Delta_x^i\|_*^2 + \frac{\tau_i\gamma_i}{2(1-p)\alpha_Y}\|\Delta_y^i\|_*^2 + \gamma_i\langle\Delta^i, z_i^v - z_i\rangle + \gamma_i\langle -\Delta^i, z_i^v - z\rangle \right\} \\ &\le \mathcal{B}_t(z, z_{[t]}^v) + \underbrace{\frac{1}{2}\sum_{i=1}^t \left\{ \frac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\|\Delta_x^i\|_*^2 + \frac{(2-p)\tau_i\gamma_i}{(1-p)\alpha_Y}\|\Delta_y^i\|_*^2 + \gamma_i\langle\Delta^i, z_i^v - z_i\rangle \right\}}_{U_t},\end{aligned} \tag{4.26}$$

where similar to (4.9) we have $\mathcal{B}_t(z, z_{[t]}^v) \le \Omega_X^2\gamma_t/\eta_t + \Omega_Y^2\gamma_t/\tau_t$. Using the above inequality, (2.13), (2.14) and (4.24), we obtain

$$\beta_t\gamma_t g(z_{t+1}^{ag}) \le \frac{2\gamma_t}{\eta_t}\Omega_X^2 + \frac{2\gamma_t}{\tau_t}\Omega_Y^2 + U_t. \tag{4.27}$$

Now it suffices to bound the above quantity U_t, both in expectation (part a)) and in probability (part b)).

We first show part a). Note that by our assumptions on $\mathcal{SO}$, at iteration i of Algorithm 3, the random noises Δ^i are independent of z_i and hence $\mathbb{E}[\langle\Delta^i, z - z_i\rangle] = 0$. In addition, Assumption **A1** implies that $\mathbb{E}[\|\Delta_x^i\|_*^2] \le \sigma_{x,G}^2 + \sigma_{x,K}^2 = \sigma_x^2$ (noting that $\Delta_{x,G}^i$ and $\Delta_{x,K}^i$ are independent at iteration i), and $\mathbb{E}[\|\Delta_y^i\|_*^2] \le \sigma_y^2$. Therefore,

$$\mathbb{E}[U_t] \le \frac{1}{2}\sum_{i=1}^t \left\{ \frac{(2-q)\eta_i\gamma_i\sigma_x^2}{(1-q)\alpha_X} + \frac{(2-p)\tau_i\gamma_i\sigma_y^2}{(1-p)\alpha_Y} \right\}. \tag{4.28}$$

Taking expectation on both sides of (4.27) and using the above inequality, we obtain (3.5).

We now show that part b) holds. Note that by our assumptions on $\mathcal{SO}$ and the definition of z_i^v, the sequences $\{\langle\Delta_{x,G}^i, x_i^v - x_i\rangle\}_{i\ge1}$ is a martingale-difference sequence. By the well-known large-deviation theorem for martingale-difference sequence (e.g., Lemma 2 of [27]), and the fact that

$$\begin{aligned}&\mathbb{E}[\exp\left\{\alpha_X\gamma_i^2\langle\Delta_{x,G}^i, x_i^v - x_i\rangle^2/\left(2\gamma_i^2\Omega_X^2\sigma_{x,G}^2\right)\right\}] \le \mathbb{E}[\exp\left\{\alpha_X\|\Delta_{x,G}^i\|_*^2\|x_i^v - x_i\|^2/\left(2\Omega_X^2\sigma_{x,G}^2\right)\right\}] \\ \le&\mathbb{E}[\exp\left\{\|\Delta_{x,G}^i\|_*^2 V_X(x_i^v, x_i)/\left(\Omega_X^2\sigma_{x,G}^2\right)\right\}] \le \mathbb{E}[\exp\left\{\|\Delta_{x,G}^i\|_*^2/\sigma_{x,G}^2\right\}] \le \exp\{1\},\end{aligned}$$

we conclude that

$$\text{Prob}\left\{\sum_{i=1}^t \gamma_i\langle\Delta_{x,G}^i, x_i^v - x_i\rangle > \lambda\cdot\sigma_{x,G}\Omega_X\sqrt{\tfrac{2}{\alpha_X}\sum_{i=1}^t\gamma_i^2}\right\} \le \exp\{-\lambda^2/3\}, \forall\lambda > 0.$$

By using a similar argument, we can show that, $\forall\lambda > 0$,

$$\begin{aligned}&\text{Prob}\left\{\sum_{i=1}^t \gamma_i\langle\Delta_y^i, y_i^v - y_i\rangle > \lambda\cdot\sigma_y\Omega_Y\sqrt{\tfrac{2}{\alpha_Y}\sum_{i=1}^t\gamma_i^2}\right\} \le \exp\{-\lambda^2/3\}, \\ &\text{Prob}\left\{\sum_{i=1}^t \gamma_i\langle\Delta_{x,K}^i, x - x_i\rangle > \lambda\cdot\sigma_{x,K}\Omega_X\sqrt{\tfrac{2}{\alpha_X}\sum_{i=1}^t\gamma_i^2}\right\} \le \exp\{-\lambda^2/3\}.\end{aligned}$$

Using the previous three inequalities and the fact that $\sigma_{x,G}+\sigma_{x,K}\le\sqrt{2}\sigma_x$, we have, $\forall\lambda>0$,

$$\begin{aligned}&\text{Prob}\left\{\sum_{i=1}^t\gamma_i\langle\Delta^i,z_i^v-z_i\rangle>\lambda\left[\tfrac{\sqrt{2}\sigma_x\Omega_X}{\sqrt{\alpha_X}}+\tfrac{\sigma_y\Omega_Y}{\sqrt{\alpha_Y}}\right]\sqrt{2\textstyle\sum_{i=1}^t\gamma_i^2}\right\}\le\\&\text{Prob}\left\{\sum_{i=1}^t\gamma_i\langle\Delta^i,z_i^v-z_i\rangle>\lambda\left[\tfrac{(\sigma_{x,G}+\sigma_{x,K})\Omega_X}{\sqrt{\alpha_X}}+\tfrac{\sigma_y\Omega_Y}{\sqrt{\alpha_Y}}\right]\sqrt{2\textstyle\sum_{i=1}^t\gamma_i^2}\right\}\le 3\exp\{-\lambda^2/3\}.\end{aligned}\tag{4.29}$$

Now let $S_i:=(2-q)\eta_i\gamma_i/[(1-q)\alpha_X]$ and $S:=\sum_{i=1}^t S_i$. By the convexity of exponential function, we have

$$\mathbb{E}\left[\exp\left\{\tfrac{1}{S}\textstyle\sum_{i=1}^t S_i\|\Delta_{x,G}^i\|_*^2/\sigma_{x,G}^2\right\}\right]\le\mathbb{E}\left[\tfrac{1}{S}\textstyle\sum_{i=1}^t S_i\exp\left\{\|\Delta_{x,G}^i\|_*^2/\sigma_{x,G}^2\right\}\right]\le\exp\{1\}.$$

where the last inequality follows from Assumption **A2**. Therefore, by Markov's inequality, for all $\lambda>0$,

$$\begin{aligned}&\text{Prob}\left\{\textstyle\sum_{i=1}^t\frac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\|\Delta_{x,G}^i\|_*^2>(1+\lambda)\sigma_{x,G}^2\sum_{i=1}^t\frac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\right\}\\&=\text{Prob}\left\{\exp\left\{\tfrac{1}{S}\textstyle\sum_{i=1}^t S_i\|\Delta_{x,G}^i\|_*^2/\sigma_{x,G}^2\right\}\ge\exp\{1+\lambda\}\right\}\le\exp\{-\lambda\}.\end{aligned}$$

Using an similar argument, we can show that

$$\begin{aligned}&\text{Prob}\left\{\textstyle\sum_{i=1}^t\frac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\|\Delta_{x,K}^i\|_*^2>(1+\lambda)\sigma_{x,K}^2\sum_{i=1}^t\frac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\right\}\le\exp\{-\lambda\},\\&\text{Prob}\left\{\textstyle\sum_{i=1}^t\frac{(2-p)\tau_i\gamma_i}{(1-p)\alpha_Y}\|\Delta_y^i\|_*^2>(1+\lambda)\sigma_y^2\sum_{i=1}^t\frac{(2-p)\tau_i\gamma_i}{(1-p)\alpha_Y}\right\}\le\exp\{-\lambda\}.\end{aligned}$$

Combining the previous three inequalities, we obtain

$$\begin{aligned}&\text{Prob}\left\{\textstyle\sum_{i=1}^t\frac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}\|\Delta_x^i\|_*^2+\sum_{i=1}^t\frac{(2-p)\tau_i\gamma_i}{(1-p)\alpha_Y}\|\Delta_y^i\|_*^2>\right.\\&\left.(1+\lambda)\left[\sigma_x^2\textstyle\sum_{i=1}^t\frac{(2-q)\eta_i\gamma_i}{(1-q)\alpha_X}+\sigma_y^2\sum_{i=1}^t\frac{(2-p)\tau_i\gamma_i}{(1-p)\alpha_Y}\right]\right\}\le 3\exp\{-\lambda\},\end{aligned}\tag{4.30}$$

Our result now follows directly from (4.26), (4.27), (4.29) and (4.30). □

In the remaining part of this subsection, our goal is to prove Theorem 3.3, which describes the convergence rate of Algorithm 3 when X and Y are both unbounded. Similar as proving Theorem 2.3, first we specialize the result of Lemma 4.4 under (2.15), (2.22) and (3.4). The following lemma is analogous to Lemma 4.3.

LEMMA 4.6. *Let $\hat{z}=(\hat{x},\hat{y})\in Z$ be a saddle point of (1.1). If $V_X(x,x_t)=\|x-x_t\|^2/2$ and $V_Y(y,y_t)=\|y-y_t\|^2/2$ in Algorithm 3, and the parameters β_t,θ_t,η_t and τ_t satisfy (2.15), (2.22) and (3.4), then*

(a).

$$\begin{aligned}&\|\hat{x}-x_{t+1}\|^2+\|\hat{x}-x_{t+1}^v\|^2+\frac{\eta_t(1-p)}{\tau_t}\|\hat{y}-y_{t+1}\|^2+\frac{\eta_t}{\tau_t}\|\hat{y}-y_{t+1}^v\|^2\\&\le 2\|\hat{x}-x_1\|^2+\frac{2\eta_t}{\tau_t}\|\hat{y}-y_1\|^2+\frac{2\eta_t}{\gamma_t}U_t,\ \textit{for all } t\ge 1,\end{aligned}\tag{4.31}$$

where $(x_{t+1}^v,\,y_{t+1}^v)$ and U_t are defined in (4.22) and (4.26), respectively.

(b).

$$\tilde{g}(z_{t+1}^{ag},v_{t+1})\le\frac{1}{\beta_t\eta_t}\|x_{t+1}^{ag}-x_1\|^2+\frac{1}{\beta_t\tau_t}\|y_{t+1}^{ag}-y_1\|^2+\frac{1}{\beta_t\gamma_t}U_t=:\delta_{t+1},\ \textit{for all } t\ge 1,\tag{4.32}$$

where $\tilde{g}(\cdot,\cdot)$ is defined in (2.21) and

$$v_{t+1}=\left(\frac{1}{\beta_t\eta_t}(2x_1-x_{t+1}-x_{t+1}^v),\frac{1}{\beta_t\tau_t}(2y_1-y_{t+1}-y_{t+1}^v)+\frac{1}{\beta_t}K(x_{t+1}-x_t)\right).\tag{4.33}$$

Proof. Apply (3.4), (4.15) and (4.26) to (4.18) in Lemma 4.4, we get

$$\beta_t\gamma_tQ(z_{t+1}^{ag},z)\le\bar{\mathcal{B}}(z,z_{[t]})+\tfrac{p\gamma_t}{2\tau_t}\|y-y_{t+1}\|^2+\bar{\mathcal{B}}(z,z_{[t]}^v)+U_t,$$

where $\bar{\mathcal{B}}(\cdot,\cdot)$ *is defined as*

$$\bar{\mathcal{B}}(z,\tilde{z}_{[t]}) := \tfrac{\gamma_t}{2\eta_t}\|x-\tilde{x}_1\|^2 - \tfrac{\gamma_t}{2\eta_t}\|x-\tilde{x}_{t+1}\|^2 + \tfrac{\gamma_t}{2\tau_t}\|y-\tilde{y}_1\|^2 - \tfrac{\gamma_t}{2\tau_t}\|y-\tilde{y}_{t+1}\|^2,\ \forall z\in Z \text{ and } \tilde{z}_{[t]}\subset Z$$

thanks to (2.22). *Now letting* $z=\hat{z}$, *and noting that* $Q(z^{ag}_{t+1},\hat{z})\geq 0$, *we get* (4.31).

On the other hand, if we only apply (3.4) *and* (4.26) *to* (4.18) *in Lemma 4.4, then we get*

$$\beta_t\gamma_t Q(z^{ag}_{t+1},z) \leq \bar{\mathcal{B}}(z,z_{[t]}) + \gamma_t\langle K(x_{t+1}-x_t), y-y_{t+1}\rangle + \bar{\mathcal{B}}(z,z^v_{[t]}) + U_t.$$

Apply (2.22) *and* (4.16) *to* $\bar{\mathcal{B}}(z,z_{[t]})$ *and* $\bar{\mathcal{B}}(z,z^v_{[t]})$ *in the above inequality, we get* (4.32). □

With the help of Lemma 4.6, we are ready to prove Theorem 3.3.

Proof of Theorem 3.3 Let δ_{t+1} and v_{t+1} be defined in (4.32) and (4.33), respectively. Also let C and D, respectively, be defined in (3.14) and (2.26). It suffices to estimate $\mathbb{E}[\|v_{t+1}\|]$ and $\mathbb{E}[\delta_{t+1}]$. First it follows from (2.22), (3.14) and (4.28) that

$$\mathbb{E}[U_t] \leq \frac{\gamma_t}{\eta_t}C^2. \tag{4.34}$$

Using the above inequality, (2.22), (2.26) and (4.31), we have $\mathbb{E}[\|\hat{x}-x_{t+1}\|^2] \leq 2D^2+2C^2$ and $\mathbb{E}[\|\hat{y}-y_{t+1}\|^2] \leq (2D^2+2C^2)\frac{\tau_1}{\eta_1(1-p)}$, which, by Jensen's inequality, then imply that $\mathbb{E}[\|\hat{x}-x_{t+1}\|] \leq \sqrt{2D^2+2C^2}$ and $\mathbb{E}[\|\hat{y}-y_{t+1}\|] \leq \sqrt{2D^2+2C^2}\sqrt{\frac{\tau_1}{\eta_1(1-p)}}$. Similarly, we can show that $\mathbb{E}[\|\hat{x}-x^v_{t+1}\|] \leq \sqrt{2D^2+2C^2}$ and $\mathbb{E}[\|\hat{y}-y^v_{t+1}\|] \leq \sqrt{2D^2+2C^2}\sqrt{\frac{\tau_1}{\eta_1}}$. Therefore, by (4.33) and the above four inequalities, we have

$$\begin{aligned}
&\mathbb{E}[\|v_{t+1}\|]\\
&\leq \mathbb{E}\left[\tfrac{1}{\beta_t\eta_t}\left(\|x_1-x_{t+1}\|+\|x_1-x^v_{t+1}\|\right) + \tfrac{1}{\beta_t\tau_t}\left(\|y_1-y_{t+1}\|+\|y_1-y^v_{t+1}\|\right) + \tfrac{L_K}{\beta_t}\|x_{t+1}-x_t\|\right]\\
&\leq \mathbb{E}\left[\tfrac{1}{\beta_t\eta_t}\left(2\|\hat{x}-x_1\|+\|\hat{x}-x_{t+1}\|+\|\hat{x}-x^v_{t+1}\|\right)\right.\\
&\quad \left.+\tfrac{1}{\beta_t\tau_t}\left(2\|\hat{y}-y_1\|+\|\hat{y}-y_{t+1}\|+\|\hat{y}-y^v_{t+1}\|\right) + \tfrac{L_K}{\beta_t}\left(\|\hat{x}-x_{t+1}\|+\|\hat{x}-x_t\|\right)\right]\\
&\leq \tfrac{2\|\hat{x}-x_1\|}{\beta_t\eta_t} + \tfrac{2\|\hat{y}-y_1\|}{\beta_t\tau_t} + \sqrt{2D^2+2C^2}\left[\tfrac{2}{\beta_t\eta_t} + \tfrac{1}{\beta_t\tau_t}\sqrt{\tfrac{\tau_1}{\eta_1}}\left(\sqrt{\tfrac{1}{1-p}}+1\right) + \tfrac{2L_K}{\beta_t}\right],
\end{aligned}$$

thus (3.13) holds.

Now let us estimate a bound on δ_{t+1}. By (4.17), (4.28), (4.31) and (4.34), we have

$$\begin{aligned}
\mathbb{E}[\delta_{t+1}] =&\ \mathbb{E}\left[\tfrac{1}{\beta_t\eta_t}\|x^{ag}_{t+1}-x_1\|^2 + \tfrac{1}{\beta_t\tau_t}\|y^{ag}_{t+1}-y_1\|^2\right] + \tfrac{1}{\beta_t\gamma_t}\mathbb{E}[U_t]\\
\leq&\ \mathbb{E}\left[\tfrac{2}{\beta_t\eta_t}\left(\|\hat{x}-x^{ag}_{t+1}\|^2+\|\hat{x}-x_1\|^2\right) + \tfrac{2}{\beta_t\tau_t}\left(\|\hat{y}-y^{ag}_{t+1}\|^2+\|\hat{y}-y_1\|^2\right)\right] + \tfrac{1}{\beta_t\eta_t}C^2\\
=&\ \mathbb{E}\left[\tfrac{1}{\beta_t\eta_t}\left(2D^2+2\|\hat{x}-x^{ag}_{t+1}\|^2 + \tfrac{2\eta_t(1-p)}{\tau_t}\|\hat{y}-y^{ag}_{t+1}\|^2 + \tfrac{2\eta_t p}{\tau_t}\|\hat{y}-y^{ag}_{t+1}\|^2\right)\right] + \tfrac{1}{\beta_t\eta_t}C^2\\
\leq&\ \tfrac{1}{\beta_t\eta_t}\left[2D^2 + \tfrac{2}{\beta_t\gamma_t}\textstyle\sum_{i=1}^t\gamma_i\left(\mathbb{E}\left[\|\hat{x}-x_{i+1}\|^2\right] + \tfrac{\eta_t(1-p)}{\tau_t}\mathbb{E}\left[\|\hat{y}-y_{i+1}\|^2\right] + \tfrac{\eta_t p}{\tau_t}\mathbb{E}\left[\|\hat{y}-y_{i+1}\|^2\right]\right) + C^2\right]\\
\leq&\ \tfrac{1}{\beta_t\eta_t}\left[2D^2 + \tfrac{2}{\beta_t\gamma_t}\textstyle\sum_{i=1}^t\gamma_i\left(2D^2+C^2 + \tfrac{\eta_t p}{\tau_t}\cdot\tfrac{\tau_1}{\eta_1(1-p)}(2D^2+C^2)\right) + C^2\right] = \tfrac{1}{\beta_t\eta_t}\left(\tfrac{6-4p}{1-p}D^2 + \tfrac{5-3p}{1-p}C^2\right).
\end{aligned}$$

Therefore (3.12) holds.
□

5. Numerical examples. In this section we will present our experimental results on solving three saddle point problems using the deterministic or stochastic APD algorithm. The comparisons with the linearized version of the primal dual algorithm in [9], Nesterov's smoothing technique in [43], Nemirovski's mirror-prox method in [36], the mirror-descent stochastic approximation method in [35] and the stochastic mirror-prox method in [21] are provided for a better examination of the performance of the APD algorithm.

5.1. Image reconstruction. Our primary goal in this subsection is to compare the performance of Algorithms 1 and 2. Consider the following total variation (TV) regularized linear inversion problem, which has been widely used as a framework for image reconstruction:

$$\min_{x\in X} f(x) := \frac{1}{2}\|Ax-b\|^2 + \lambda\|Dx\|_{2,1}, \tag{5.1}$$

where x is the reconstructed image, $\|Dx\|_{2,1}$ is the discrete form of the TV semi-norm, A is a given structure matrix (depending on the physics of the data acquisition), b represents the observed data, and $X := \{x \in \mathbb{R}^n : l_* \le x^{(i)} \le u_*, \forall i = 1,\dots,n\}$. For simplicity, we consider x as a n-vector form of a two-dimensional image. Problem (5.1) can be reformulated as the following SPP problem of in the form of (1.1):

$$\min_{x\in X}\max_{y\in Y}\{\frac{1}{2}\|Ax-b\|^2 + \lambda\langle Dx, y\rangle\},$$

where $Y := \{y \in \mathbb{R}^{2n} : \|y\|_{2,\infty} := \max_{i=1,\dots,n}\|y^i\|_2 \le 1\}$, and $\|y^i\|_2$ is the Euclidean norm of y^i in R^2.

In our experiment, we consider two types of instances depending on how the structure matrix $A \in R^{k\times n}$ is generated. More specifically, the entries of A are normally distributed according to $N(0, 1/\sqrt{k})$ for the first tye of instance, while for the second one, the entries of A are generated independently from a Bernoulli distribution, i.e., each entry of A is given by $1/\sqrt{k}$ or $-1/\sqrt{k}$ with equal probability. Both types of structure matrices are widely used in compressive sensing (see, e.g., [3]). For a given A, the measurements are generated by $b = Ax_{true} + \varepsilon$, where x_{true} is a 64 by 64 Shepp-Logan phantom [48] with intensities in $[0, 1]$, and $\varepsilon \equiv N(0, 10^{-6}I_k)$ with $k = 2048$. We set $X := \{x \in \mathbb{R}^n : 0 \le x^{(i)} \le 1, \forall i = 1,\dots,n\}$ and $\lambda = 10^{-3}$ in (5.1).

We applied the linearized version of Algorithm 1, denoted by LPD, in which (2.2) replaced by (2.10), and the APD algorithm to solve problem (5.1). In LPD the stepsize parameters are set to $\eta_t = 1/(L_G + L_K D_Y/D_X)$, $\tau_t = D_Y/(L_K D_X)$ and $\theta_t = (t-1)/t$. The stepsizes in APD are chosen as in Corollary 2.2, and the Bregman divergences are defined as $V_X(x_t, x) := \|x_t - x\|_2^2/2$ and $V_Y(y_t, y) := \|y_t - y\|_2^2/2$, hence $D_Y/D_X = 1$. In addition, we also applied the APD algorithm with unbounded feasible sets, denoted APD-U, to solve (5.1) by assuming that X is unbounded. The stepsizes in APD-U are chosen as in Corollary 2.4, and we set $N = 150$. To have a fair comparison, we use the same Lipschitz constants L_G and L_K for all algorithms without performing a backtracking. It can be easily seen that $L_G^* = \lambda_{max}(A^T A)$ and $L_K^* = \lambda\sqrt{8}$ (see [8]) are the smallest Lipschitz constants that satisfy (1.2). Moreover, since in many applications the Lipschitz constants are either unknown or expensive to compute, the robustness to the overestimated Lipschitz constants of the algorithm is important in practice. Hence, we also compare the sensitivity to the overestimated Lipschitz constants of the algorithms APD, APD-U and LPD in image reconstruction.

To do so, we supply all algorithms with the best Lipschitz constants $L_G = L_G^*$ and $L_K = L_K^*$ first. For an approximate solution $\tilde{x} \in \mathbb{R}^n$, we report both the primal objective function value $f(\tilde{x})$ and the reconstruction error relative to the ground truth, i.e., $r(\tilde{x}) := \|\tilde{x} - x_{true}\|_2/\|x_{true}\|_2$, versus CPU time, as shown in Figure 5.1. Moreover, to test the sensitivity of all algorithms with respect to L_G and L_K, we also supply all algorithms with over-estimated Lipschitz constants $L_G = \zeta_G L_G^*$ and $L_K = \zeta_K L_K^*$, where $\zeta_G, \zeta_K \in \{2^{i/2}\}_{i=0}^8$. We report in Figure 5.2 the relationship between the multipliers ζ_G, ζ_K and the primal objective function value of all algorithms after N iterations. We make a few observations about the obtained results. Firstly, for solving the image reconstruction problem (5.1), both APD and APD-U outperform LPD in terms of the decreasing of objective value and relative error. Secondly, although APD-U has the same rate of convergence as APD, its practical performance is not as good as APD. A plausible explanation is that we need to specify more conservative stepsize parameters in APD-U (see (2.27) and (2.19)) in order to ensure its convergence for unbounded sets X and Y, which may contribute to its inferior practical performance. Finally, the performance of both APD and APD-U is more robust than LPD when L_G is over-estimated. This is consistent with our theoretical observations that both APD and APD-U have better rates of convergence than LPD in terms of the dependence on L_G.

5.2. Nonlinear game. Our next experiment considers a nonlinear two-person game

$$\min_{x\in\Delta^n}\max_{y\in\Delta^m}\frac{1}{2}\langle Qx, x\rangle + \langle Kx, y\rangle, \tag{5.2}$$

where $Q = A^T A$ is a positive semidefinite matrix with $A \in \mathbb{R}^{k\times n}$, and Δ^n and Δ^m are standard simplices:

$$\Delta^n := \left\{x \in \mathbb{R}^n_+ : \sum_{i=1}^n x^{(i)} = 1\right\} \text{ and } \Delta^m := \left\{y \in \mathbb{R}^n_+ : \sum_{i=1}^m y^{(i)} = 1\right\}.$$

We generate each entry of A independently from the standard normal distribution, and each entry of K independently and uniformly from the interval $[-1, 1]$.

Problem (5.2) can be interpreted as a two-person game, in which the first player has n strategies and chooses the i-th strategy with probability $x^{(i)}$, $i = 1, \ldots, n$. On the other hand, the second player has m strategies and chooses strategy $i = 1, \ldots, m$ with probability $y^{(i)}$. The goal of the first player is to minimize the loss while the second player aims to maximize the gain, and the payoff of the game is a quadratic function that depends on the strategies of both players. A saddle point of (5.2) is a Nash equilibrium of this nonlinear game.

It has been shown (e.g., [41, 34, 35]) that the Euclidean distances $V_X(x_t, x) = \|x_t - x\|_2^2/2$ and$V_Y(y_t, y) = \|y_t - y\|_2^2/2$ are not the most suitable for solving optimization problems on simplices. In this experiment, we

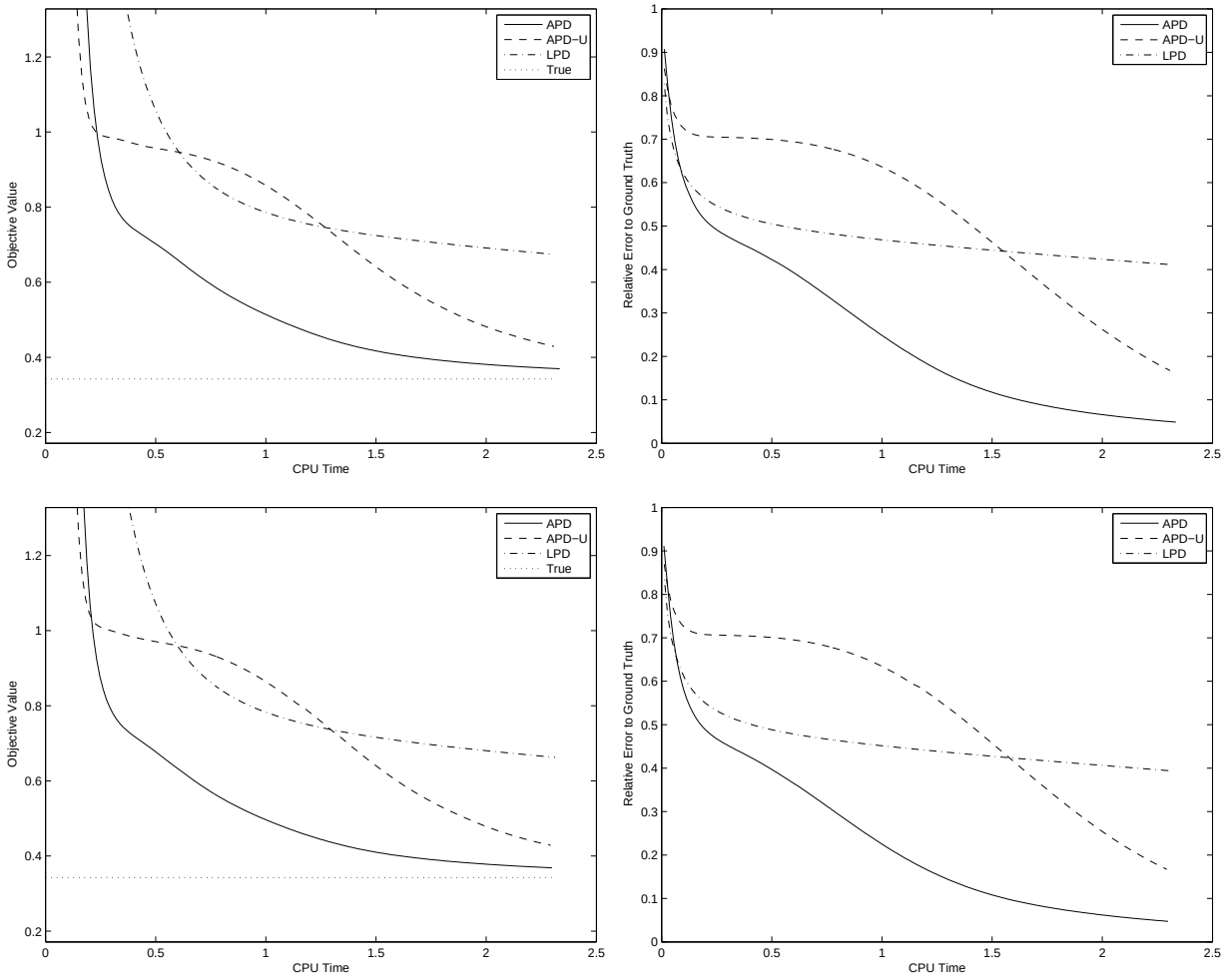


Fig. 5.1: Comparisons of APD, APD-U and LPD in image reconstruction. The top and bottom rows, respectively, show the performance of these algorithms on the "Gaussian" and "Bernoulli" instances. Left: the objective function values $f(x_t^{ag})$ from APD and APD-U, and $f(x_t)$ from LPD vs. CPU time. The straight line at the bottom is $f(x_{true})$. Right: the relative errors $r(x_t^{ag})$ from APD and APD-U and $r(x_t)$ in LPD vs. CPU time.

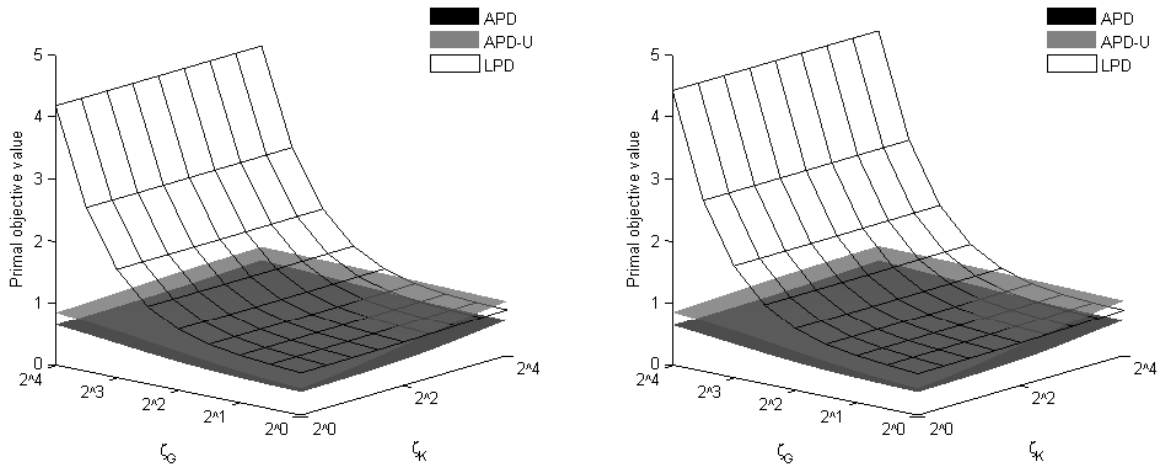


Fig. 5.2: Sensitivity to the overestimated Lipschitz constants: comparisons of APD, APD-U and LPD in image reconstruction. Left: the primal objective function values $f(x_N^{ag})$ from APD and APD-U, and $f(x_N)$ from LPD vs. ζ_G and ζ_K on the "Gaussian" instance. Right: the primal objective function values $f(x_N^{ag})$ from APD and APD-U, and $f(x_N)$ from LPD vs. ζ_G and ζ_K on the "Bernoulli" instance.

choose $\|\cdot\| := \|\cdot\|_1$ and $\|\cdot\|_* := \|\cdot\|_\infty$ in both spaces $\mathcal{X}$ and $\mathcal{Y}$, and use the following entropy setting for Bregman divergences $V_X(\cdot,\cdot)$ and $V_Y(\cdot,\cdot)$:

$$\begin{aligned} &V_X(x_t,x) := \sum_{i=1}^{n}(x^{(i)}+\nu/n)\ln\frac{x^{(i)}+\nu/n}{x_t^{(i)}+\nu/n},\ V_Y(y_t,y) := \sum_{i=1}^{m}(y^{(i)}+\nu/m)\ln\frac{y^{(i)}+\nu/m}{y_t^{(i)}+\nu/m},\\ &L_G^* = \max_{i,j}|Q^{(i,j)}|,\ L_K^* = \max_{i,j}|K^{(i,j)}|,\ \alpha_X = 1+\nu,\ \alpha_Y = 1+\nu,\\ &\Omega_X^2 = (1+\frac{\nu}{n})\ln(\frac{n}{\nu}+1),\ \Omega_Y^2 = (1+\frac{\nu}{m})\ln(\frac{m}{\nu}+1),\ D_X = \Omega_X\sqrt{2/\alpha_X},\ D_Y = \Omega_Y\sqrt{2/\alpha_Y}, \end{aligned} \tag{5.3}$$

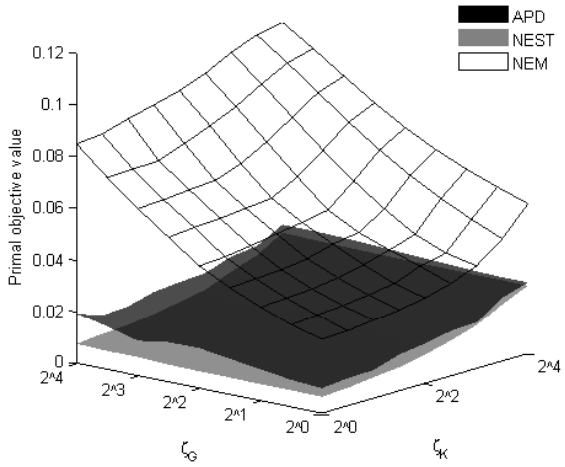
where L_G^* and L_K^* are the smallest Lipschitz constants, and ν is arbitrarily small (e.g., $\nu = 10^{-16}$), see [5] for the calculation of α_X, α_Y, Ω_X and Ω_Y. With this setting, the subproblems in (2.5) and (2.6) can be efficiently solved within machine accuracy [5].

In this experiment, we compare the proposed APD algorithm with Nesterov's smoothing technique in [41] and Nemirovski's mirror-prox method in [34]. The notation APD denotes the APD algorithm with the stepsizes in Corollary 2.2 and (5.3). NEST denotes Nesterov's algorithm in Section 5.3 of [41] with entropy distance (See Theorem 3 and Section 4.1 in [41] for details about the setting for Nesterov's algorithm). NEM denotes Nemirovski's mirror-prox method in (3.2)-(3.4) of [34] in which $L = \max_{i,j}|Q^{(i,j)}|D_X^2/2 + \max_{i,j}|K^{(i,j)}|D_XD_Y$ (see "Mixed setups" in Section 5 in [34] for the variational inequality formation of SPP (5.2). In particular, we set $L_{11} = L_G$, $L_{12} = L_{21} = L_K$, $L_{22} = 0$, $\Theta_1 = \Omega_X^2$, $\Theta_2 = \Omega_Y^2$, $\alpha_1 = \alpha_X$ and $\alpha_2 = \alpha_Y$ in (5.11) in [34]).

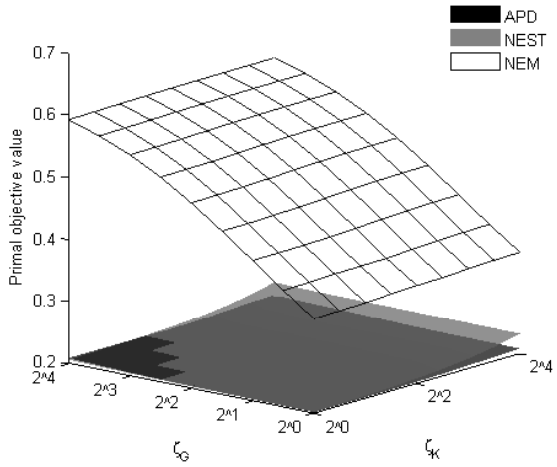
Our basic observations are as follows. First, both APD and NEST exhibit similar numerical performance when applied to the test problems in Table 5.1. It should be noted, however, that APD can be potentially applied to a wider class of problems, e.g., those with unbounded dual feasible set Y. Second, both APD and NEST decrease the primal objective function value faster than NEM, and are more robust to the overestimation of L_G. This is consistent with our theoretical observations that both APD and NEST enjoy the optimal rate of convergence (1.4), while NEM obeys a sub-optimal rate of convergence in (1.5). In addition, both APD and NEST have lower iteration cost than NEM, since NEM requires an extragradient step in its inner iteration.

Table 5.1: Nonlinear game.

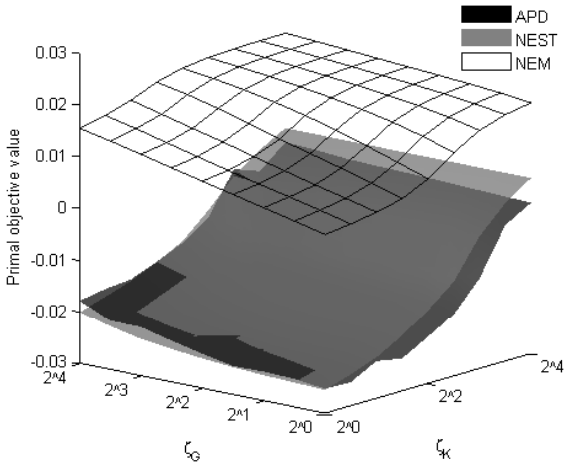
Algorithm	N	Instance 1: k=100, n=1000, m=1000 $L_G = 148.60$, $L_K = 1$		Instance 2: k=1000, n=1000, m=1000 $L_G = 1139.98$, $L_K = 1$		Instance 3: k=100 n=10000, m=1000 $L_G = 161.93$, $L_K = 1$		Instance 4: k=1000, n=10000, m=1000 $L_G = 1198.02$, $L_K = 1$	
		Obj. Val.	CPU	Obj. Val.	CPU	Obj. Val.	CPU	Obj. Val.	CPU
APD	100	0.038	0.7	0.302	0.4	0.015	17.8	0.031	14.3
	1000	0.014	6.3	0.203	4.0	-0.023	184.7	-0.005	141.2
	2000	0.010	12.5	0.202	8.1	-0.025	354.1	-0.018	285.0
NEST	100	0.047	0.6	0.304	0.4	0.016	18.0	0.031	14.4
	1000	0.008	6.3	0.205	4.0	-0.021	279.6	-0.011	141.8
	2000	0.006	12.5	0.202	8.0	-0.026	441.9	-0.019	284.6
NEM	100	0.114	1.2	0.604	0.8	0.023	35.5	0.073	28.4
	1000	0.038	12.1	0.427	7.6	0.009	401.5	0.043	279.6
	2000	0.028	24.1	0.352	15.3	0.005	1127.3	0.029	562.0



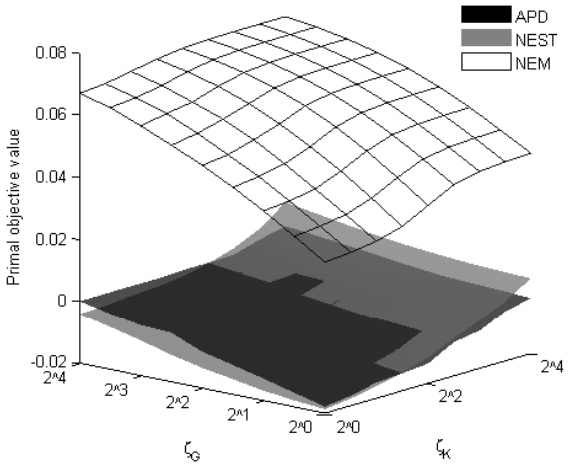
(a) Instance 1



(b) Instance 2



(c) Instance 3



(d) Instance 4

Fig. 5.3: Sensitivity to the overestimated Lipschitz constants: comparisons of APD, NEST and NEM in nonlinear game. The figures are the primal objective function values vs. ζ_G and ζ_K after $N = 2000$ iterations.

5.3. Randomized algorithms for nonlinear game. Our goal in the last experiment is to test the efficiency of stochastic APD in solving the nonlinear game in (5.2). In particular, we consider the case when both A and K are full matrix, and $m = n \gg k$. Consequently, the computation of Kx and $K^T y$ is much more expensive than that of Qx. In order to reduce the arithmetic cost for computing Kx and $K^T y$, Nemirovski et al. introduced a novel randomized algorithm in [35], where the calculations of Kx and $K^T y$ are replaced by calls to a stochastic oracle and then the mirror-descent SA algorithm is applied to solve the resulting stochastic SPP. When $Q = 0$ in (5.2), it is reported in [35] that the time for mirror-descent SA to solve (5.2) is almost within one call to the deterministic oracle to compute $[K^T y, -Kx]$ (see Sections 3.3 and 4.6 in [35]).

Using similar ideas to [35], we assume that for each input $(x_i, y_i) \in X \times Y$, the $\mathcal{SO}$ outputs the *stochastic gradient* $(\nabla G(x_i), \hat{\mathcal{K}}_x(x_i), \hat{\mathcal{K}}_y(y_i)) \equiv (Qx_i, \mathcal{K}_x(x_i, \xi_i), \mathcal{K}_y(y_i, \xi_i))$ such that

$$\text{Prob}(\hat{\mathcal{K}}_x(x_i) = K_j) = x_i^{(j)},\ \forall j = 1, \dots n, \text{ and } \text{Prob}(-\hat{\mathcal{K}}_y(y_i) = -K^l) = y_i^{(l)},\ \forall l = 1, \dots m, \tag{5.4}$$

where K_j and $(K^l)^T$ are the j-th column and l-th row of K, respectively. In other words, each call to the $\mathcal{SO}$ outputs the random samples of the columns and rows of K whose distributions depend on the input (x_i, y_i). It can be checked that $\mathbb{E}[-\hat{\mathcal{K}}_x(x_i)] = -Kx_i$, $\mathbb{E}[\hat{\mathcal{K}}_y(y_i)] = K^T y_i$, and under the settings in (5.3),

$$\mathbb{E}\left[\|\hat{\mathcal{K}}_x(x_i) - Kx_i\|_*^2\right] = \sum_{j=1}^n x_i^{(j)} \max_{1\le k\le n} \left(K^{(k,j)} - \sum_{l=1}^n K^{(k,l)} x_i^{(l)}\right)^2 = \sum_{j=1}^n x_i^{(j)} \max_{1\le k\le n} \langle K^k, e_j - x_i\rangle^2$$

$$\le \sum_{j=1}^n x_i^{(j)} \max_{1\le k\le n} \|K^k\|_\infty^2 \|e_j - x_i\|_1^2 \le 4 \max_{k,j} |K^{(k,j)}|^2,$$

and similarly $\mathbb{E}[\|\hat{\mathcal{K}}_y(y_i) - K^T y_i\|_*^2] \le 4 \max_{l,j} |K^{(l,j)}|^2$. Therefore, we set $\sigma_{x,G} = 0$, and $\sigma_y = \sigma_{x,K} = 2 \max_{l,j} |K^{(l,j)}|$.

In our experiment we set $n = m = 10^4$, $k = 100$, and use the same matrix K as in [35], i.e.,

$$K^{(i,j)} = \left(\frac{i + j - 1}{2n - 1}\right)^c,\ 1 \le i, j \le n, \tag{5.5}$$

or

$$K^{(i,j)} = \left(\frac{|i - j| + 1}{2n - 1}\right)^c,\ 1 \le i, j \le n, \tag{5.6}$$

for some constants $c > 0$. We use S-APD [1] to denote the proposed stochastic APD method with parameters described in (5.3) and Corollary 3.2, MD-SA to denote the mirror-descent SA method in [35], and SMP to denote the stochastic mirror-prox method in [21]. The iterations of MD-SA are described in (3.7) as well as Sections 3.2 and 3.3 in [35]. We set the stepsize constants (3.12) in [35] to $\theta = 1$ and $M_*^2 = 2 \ln n[(\max_{i,j} |Q^{(i,j)}| + \max_{i,j} |K^{(i,j)}|)^2 + \max_{i,j} (K^{(i,j)})^2]$. For SMP, we use the technique in NEM to formulate SPP (5.2) as a variational inequality, and apply the scheme described in (3.6) and (3.7) in [21], using the suggested stepsize constants (4.3) in [21] with $\Theta = 1$, $L = \max_{i,j} |Q^{(i,j)}| D_X^2/2 + \max_{i,j} |K^{(i,j)}| D_X D_Y$, and $M^2 = 4(\mu_1^{-2} + \mu_2^{-2}) \max_{i,j} |K^{(i,j)}|$ (see (5.11) in [34] for the definition of μ_1 and μ_2).

We report in Table 5.2 the average objective values obtained by S-APD, MD-SA and SMP over 100 runs, along with the estimated standard deviations. While a statistically more sound way to compare these algorithms is to estimate the confidence intervals associated with these objective values and/or conduct some statistical tests (e.g., the paired t test) on the collected observations, we can safely draw some conclusions directly from Table 5.2 since most of the estimated standard deviations are relatively small in comparison

[1] It should be noted that $\bar{x}_t$ in (2.9) is not necessary in Δ^n, thus $\hat{\mathcal{K}}_x(\bar{x}_t)$ may not be sampled by (5.4). However, we can set $\hat{\mathcal{K}}_x(\bar{x}_1) := \hat{\mathcal{K}}_x(x_1)$ and $\hat{\mathcal{K}}_x(\bar{x}_t) := (1 + \theta_t)\hat{\mathcal{K}}_x(x_t) - \theta_t \hat{\mathcal{K}}_x(x_{t-1})$ when $t > 1$. This setting does not affect the proof of Theorem 3.1, hence the rate of convergence of stochastic APD remains the same.

Table 5.2: Randomized algorithms for nonlinear game. The Lipschitz constant $L_G = 161.2$ in all the experiments.

Alg.	N	Instance 5: K in (5.5) with $c = 2$. $L_K = 1$			Instance 6: K in (5.5) with $c = 0.5$. $L_K = 1$			Instance 7: K in (5.6) with $c = 2$. $L_K = 0.25$			Instance 8: K in (5.6) with $c = 0.5$. $L_K = 0.5$		
		Value of f		CPU	Value of f		CPU	Value of f		CPU	Value of f		CPU
		Mean	Std.	(Avg.)	Mean	Std.	(Avg.)	Mean	Std.	(Avg.)	Mean	Std.	(Avg.)
S-APD	100	0.457	8.8e-4	0.6	0.834	3.0e-4	0.6	0.088	4.1e-3	0.6	0.495	1.6e-2	0.6
	1000	0.272	4.5e-5	3.9	0.727	5.0e-5	4.1	0.069	1.6e-3	3.9	0.472	2.8e-3	4.1
	2000	0.262	1.6e-5	9.8	0.718	1.5e-5	9.2	0.066	9.5e-4	9.8	0.449	6.7e-3	8.0
SMP	100	0.579	1.2e-4	0.9	0.865	2.0e-5	0.9	0.087	1.6e-4	0.9	0.476	1.0e-3	0.9
	1000	0.510	1.1e-4	7.4	0.850	2.0e-5	7.6	0.086	1.7e-3	7.5	0.474	5.0e-3	7.7
	2000	0.483	8.8e-5	18.6	0.844	2.1e-5	14.9	0.084	1.2e-3	15.6	0.472	2.8e-3	15.1
MD-SA	100	0.583	1.3e-4	0.6	0.866	2.5e-5	0.6	0.087	2.5e-5	0.5	0.476	1.9e-4	0.6
	1000	0.574	1.3e-4	3.7	0.861	2.1e-5	3.8	0.086	2.1e-5	3.7	0.474	1.5e-4	3.9
	2000	0.569	1.2e-4	7.3	0.859	2.2e-4	7.5	0.085	2.3e-5	7.3	0.474	1.6e-4	7.6

with the average objective values. Firstly, it can be seen that S-APD exhibits better performance than both MD-SA and SMP for the first two instances in Table 5.2. More specifically, the objective values obtained by S-APD in 100 iterations for these instances are better than those obtained by the other two algorithms in $2,000$ iterations. Secondly, for the last two instances in Table 5.2, when the number of iterations is small ($N = 100$), the objective values obtained by S-APD does not seem to be significantly different from and may even turn out to be worse than those obtained by MD-SA and SMP, respectively, for Instance 7 and 8. However, it seems that S-APD can decrease the objective values faster than the latter two algorithms, and hence that its advantages become more apparent as the number of iterations increases. In particular, the objective values obtained by S-APD in $1,000$ iterations appear to be better than those obtained by the other two algorithms in $2,000$ iterations for these two instances.

6. Concluding remarks. We present in this paper the APD method by incorporating a multi-step acceleration scheme into the primal-dual method in [9]. We show that this algorithm can achieve the optimal rate of convergence for solving both deterministic and stochastic SPP. In particular, the stochastic APD algorithm seems to be the first optimal algorithm for solving this important class of stochastic saddle-point problems in the literature. For both deterministic and stochastic SPP, the developed APD algorithms can deal with either bounded or unbounded feasible sets as long as a saddle point of SPP exists. In the unbounded case, the rate of convergence of the APD algorithms will depend on the distance from the initial point to the set of optimal solutions.

It should be noted that, although some preliminary numerical results have been reported, this paper focuses more on the theoretical studies of the iteration complexity associated with the proposed APD methods. We expect that the practical performance of these algorithms can be further improved, for example, by adaptively choosing a few algorithmic parameters, e.g., η_k, based on some backtracking techniques to search for the most appropriate Lipschitz constants L_G and L_K. It is also worth mentioning that in the APD algorithm the range of the combination parameter θ is restricted to $(0, 1]$. On the other hand, He and Yuan [17] recently showed that the range of θ in the primal-dual algorithm can be enlarged from $[0, 1]$ to $[-1, 1]$ from the perspective of contraction methods. As a result, the primal and dual stepsize condition with respect to L_K in [9] is relaxed when $\theta \in (0, 1]$. Moreover, both the primal and dual stepsizes can be arbitrarily large if $\theta = -1$. These enlarged ranges of the involved parameters can allow us to choose more aggressive stepsizes and hence possibly to improve the practical performance of the primal-dual algorithm. It will be interesting to study if the ranges of the parameters in the APD algorithm can be enlarged by incorporating the ideas in [17] or other novel methods.

REFERENCES

[1] K. Arrow, L. Hurwicz, and H. Uzawa, *Studies in Linear and Non-linear Programming*, Stanford Mathematical Studies in the Social Sciences, Stanford University Press, 1958.

[2] A. Auslender and M. Teboulle, *Interior gradient and proximal methods for convex and conic optimization*, SIAM Journal on Optimization, 16 (2006), pp. 697–725.

[3] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, *A simple proof of the restricted isometry property for random matrices*, Constructive Approximation, 28 (2008), pp. 253–263.

[4] S. Becker, J. Bobin, and E. Candès, *NESTA: a fast and accurate first-order method for sparse recovery*, SIAM Journal on Imaging Sciences, 4 (2011), pp. 1–39.

[5] A. Ben-Tal and A. Nemirovski, *Non-Euclidean restricted memory level method for large-scale convex optimizationd*, Mathematical Programming, 102 (2005), pp. 407–456.

[6] S. Bonettini and V. Ruggiero, *On the convergence of primal–dual hybrid gradient algorithms for total variation image restoration*, Journal of Mathematical Imaging and Vision, (2012), pp. 1–18.

[7] R. S. Burachik, A. N. Iusem, and B. F. Svaiter, *Enlargement of monotone operators with applications to variational inequalities*, Set-Valued Analysis, 5 (1997), pp. 159–180.

[8] A. Chambolle, *An algorithm for total variation minimization and applications*, Journal of Mathematical imaging and vision, 20 (2004), pp. 89–97.

[9] A. Chambolle and T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, Journal of Mathematical Imaging and Vision, 40 (2011), pp. 120–145.

[10] A. d'Aspremont, *Smooth optimization with approximate gradient*, SIAM Journal on Optimization, 19 (2008), pp. 1171–1183.

[11] J. Douglas and H. Rachford, *On the numerical solution of heat conduction problems in two and three space variables*, Transactions of the American mathematical Society, 82 (1956), pp. 421–439.

[12] E. Esser, X. Zhang, and T. Chan, *A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science*, SIAM Journal on Imaging Sciences, 3 (2010), pp. 1015–1046.

[13] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Computers & Mathematics with Applications, 2 (1976), pp. 17–40.

[14] S. Ghadimi and G. Lan, *Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization I: A generic algorithmic framework*, SIAM Journal on Optimization, 22 (2012), pp. 1469–1492.

[15] S. Ghadimi and G. Lan, *Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, II: Shrinking procedures and optimal algorithms*, SIAM Journal on Optimization, 23 (2013), pp. 2061–2089.

[16] R. Glowinski and A. Marroco, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires*, ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique, 9 (1975), pp. 41–76.

[17] B. He and X. Yuan, *Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective*, SIAM Journal on Imaging Sciences, 5 (2012), pp. 119–149.

[18] B. He and X. Yuan, *On the o(1/n) convergence rate of the douglas-rachford alternating direction method*, SIAM Journal on Numerical Analysis, 50 (2012), pp. 700–709.

[19] L. Jacob, G. Obozinski, and J.-P. Vert, *Group lasso with overlap and graph lasso*, in Proceedings of the 26th International Conference on Machine Learning, 2009.

[20] A. Juditsky and A. Nemirovski, *First-Order Methods for Nonsmooth Convex Large-Scale Optimization, II: Utilizing problems structure* , in Optimization for Machine Learning, Eds: S. Sra, S. Nowozin and S.J. Wright, MIT press, 2011.

[21] A. Juditsky, A. Nemirovski, and C. Tauvel, *Solving variational inequalities with stochastic mirror-prox algorithm*, Stochastic Systems, 1 (2011), pp. 17–58.

[22] T. G. Kolda and B. W. Bader, *Tensor decompositions and applications*, SIAM Review, 51(3) (2009), pp. 455–500.

[23] G. Korpelevich, *Extrapolation gradient methods and relation to modified lagrangeans*, Ekonomika i Matematicheskie Metody, 19 (1983), pp. 694–703. in Russian; English translation in Matekon.

[24] G. Lan, *An optimal method for stochastic composite optimization*, Mathematical Programming, 133 (1) (2012), pp. 365–397.

[25] G. Lan, *Bundle-level type methods uniformly optimal for smooth and nonsmooth convex optimization*, Mathematical Programming, (2013), pp. 1–45.

[26] G. Lan, Z. Lu, and R. D. C. Monteiro, *Primal-dual first-order methods with $\mathcal{O}(1/\epsilon)$ iteration-complexity for cone programming*, Mathematical Programming, 126 (2011), pp. 1–29.

[27] G. Lan, A. Nemirovski, and A. Shapiro, *Validation analysis of mirror descent stochastic approximation method*, Mathematical programming, 134 (2012), pp. 425–458.

[28] Q. Lin, X. Chen, and J. Pena, *A smoothing stochastic gradient method for composite optimization*, manuscript, Carnegie Mellon University, 2011.

[29] P. Lions and B. Mercier, *Splitting algorithms for the sum of two nonlinear operators*, SIAM Journal on Numerical Analysis, 16 (1979), pp. 964–979.

[30] J. Mairal, R. Jenatton, G. Obozinski, and F. Bach, *Convex and network flow optimization for structured sparsity*, Journal of Machine Learning Research, 12 (2011), pp. 2681–2720.

[31] R. D. Monteiro and B. F. Svaiter, *On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean*, SIAM Journal on Optimization, 20 (2010), pp. 2755–2787.

[32] ———, *Complexity of variants of Tseng's modified F-B splitting and Korpelevich's methods for hemivariational inequalities with applications to saddle-point and convex optimization problems*, SIAM Journal on Optimization, 21 (2011), pp. 1688–1720.

[33] R. D. Monteiro and B. F. Svaiter, *Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers*, SIAM Journal on Optimization, 23 (2013), pp. 475–507.
[34] A. Nemirovski, *Prox-method with rate of convergence $O(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems*, SIAM Journal on Optimization, 15 (2004), pp. 229–251.
[35] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, *Robust stochastic approximation approach to stochastic programming*, SIAM Journal on Optimization, 19 (2009), pp. 1574–1609.
[36] A. Nemirovski and D. Yudin, *Problem complexity and method efficiency in optimization*, Wiley-Interscience Series in Discrete Mathematics, John Wiley, XV, Philadelphia, 1983.
[37] A. Nemirovsky, *Information-based complexity of linear operator equations*, Journal of Complexity, 8 (1992), pp. 153–175.
[38] Y. Nesterov, *Excessive gap technique in nonsmooth convex minimization*, SIAM Journal on Optimization, 16 (2005), pp. 235–249.
[39] Y. E. Nesterov, *A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$*, Doklady AN SSSR, 269 (1983), pp. 543–547. translated as Soviet Math. Docl.
[40] ———, *Introductory Lectures on Convex Optimization: A Basic Course*, Kluwer Academic Publishers, Massachusetts, 2004.
[41] ———, *Smooth minimization of nonsmooth functions*, Mathematical Programming, 103 (2005), pp. 127–152.
[42] J. Peña, *Nash equilibria computation via smoothing techniques*, Optima, 78 (2008), pp. 12–13.
[43] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, *An algorithm for minimizing the Mumford-Shah functional*, in Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 1133–1140.
[44] B. Polyak, *New stochastic approximation type procedures*, Automat. i Telemekh., 7 (1990), pp. 98–107.
[45] B. Polyak and A. Juditsky, *Acceleration of stochastic approximation by averaging*, SIAM J. Control and Optimization, 30 (1992), pp. 838–855.
[46] H. Robbins and S. Monro, *A stochastic approximation method*, Annals of Mathematical Statistics, 22 (1951), pp. 400–407.
[47] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D: Nonlinear Phenomena, 60 (1992), pp. 259–268.
[48] L. A. Shepp and B. F. Logan, *The fourier reconstruction of a head section*, Nuclear Science, IEEE Transactions on, 21 (1974), pp. 21–43.
[49] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, *Sparsity and smoothness via the fused lasso*, Journal of Royal Statistical Society: B, 67(1) (2005), pp. 91–108.
[50] R. Tomioka, T. Suzuki, K. Hayashi, and H. Kashima, *Statistical performance of convex tensor decomposition*, Advances in Neural Information Processing Systems, 25 (2011).
[51] P. Tseng, *On accelerated proximal gradient methods for convex-concave optimization*, submitted to SIAM Journal on Optimization, (2008).
[52] M. Zhu and T. Chan, *An efficient primal-dual hybrid gradient algorithm for total variation image restoration*, UCLA CAM Report, (2008), pp. 08–34.