SIAM J. OPTIM.
Vol. 26, No. 1, pp. 29–56

AN ACCELERATED HPE-TYPE ALGORITHM FOR A CLASS OF COMPOSITE CONVEX-CONCAVE SADDLE-POINT PROBLEMS*

YUNLONG HE† AND RENATO D. C. MONTEIRO‡

Abstract. This paper proposes a new algorithm for solving a class of composite convex-concave saddle-point problems. The new algorithm is a special instance of the hybrid proximal extragradient framework in which a Nesterov accelerated variant is used to approximately solve the prox subproblems. One of the advantages of the new method is that it works for any constant choice of proximal stepsize. Moreover, a suitable choice of the latter stepsize yields a method with the best known (accelerated inner) iteration complexity for the aforementioned class of saddle-point problems. In contrast to the smoothing technique of [Y. Nesterov, *Math. Program.*, 103 (2005), pp. 127–152], our accelerated method does not assume that a feasible set is bounded due to its proximal point nature. Experiment results on three problem sets show that the new method outperforms Nesterov's smoothing technique of [Y. Nesterov, *Math. Program.*, 103 (2005), pp. 127–152].

Key words. saddle-point problem, composite convex optimization, monotone inclusion problem, inexact proximal point method, hybrid proximal extragradient, accelerated method, complexity, smoothing

AMS subject classifications. 90C60, 90C25, 90C30, 47H05, 47J20, 65K10, 65K05

DOI. 10.1137/14096757X

1. Introduction.

A broad class of optimization, saddle-point (SP), equilibrium, and variational inequality problems can be posed as the *monotone inclusion problem*; namely, find z such that

$$0 \in T(z), \tag{1.1}$$

where T is a maximal monotone point-to-set operator. The proximal point method, proposed by Rockafellar [16], is a classical iterative scheme for solving the monotone inclusion problem which generates a sequence $\{z_k\}$ according to

$$\|z_k - (\lambda_k T + I)^{-1}(z_{k-1})\| \le e_k, \qquad \sum_{k=1}^{\infty} e_k < \infty.$$

This method has been used as a generic framework for the design and analysis of several implementable algorithms.

New inexact versions of the proximal point method, which uses instead relative error criteria, were proposed by Solodov and Svaiter [18, 19, 20, 21]. In this paper, we will use one of these variants, namely, the hybrid proximal extragradient (HPE) framework studied in [18], to develop and analyze a new algorithm, and we now briefly discuss this framework. The *exact* proximal point iteration from z with stepsize $\lambda > 0$ is given by $z_+ = (\lambda T + I)^{-1}(z)$, which is equivalent to

$$r \in T(z_+), \qquad \lambda r + z_+ - z = 0. \tag{1.2}$$

*Received by the editors May 5, 2014; accepted for publication (in revised form) October 20, 2015; published electronically January 6, 2016.
http://www.siam.org/journals/siopt/26-1/96757.html

†School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332 (heyunlong@gatech.edu).

‡School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205 (monteiro@isye.gatech.edu). The work of this author was partially supported by NSF grant CMMI-1300221.

In each step of the HPE, the above *proximal system* is solved inexactly with $(z, \lambda) = (z_{k-1}, \lambda_k)$ to obtain $z_k = z_+$ as follows. For a given constant $\sigma \in [0, 1)$, a triple $(\tilde{z}, \tilde{r}, \varepsilon) = (\tilde{z}_k, \tilde{r}_k, \varepsilon_k)$ satisfying the HPE error criteria

$$\tilde{r} \in T^{\varepsilon}(\tilde{z}), \qquad \|\lambda\tilde{r} + \tilde{z} - z\|^2 + 2\lambda\varepsilon \leq \sigma^2\|\tilde{z} - z\|^2 \tag{1.3}$$

is found, where T^{ε} denotes the ε-enlargement [3] of T. (It has the property that $T^{\varepsilon}(z) \supset T(z)$ for each z.) Note that this construction relaxes both the inclusion and the equation in (1.2). Finally, instead of choosing $\tilde{z}$ as the next iterate z_+, the HPE framework computes the next iterate z_+ by means of the *extragradient* step $z_+ = z - \lambda\tilde{r}$.

Iteration complexity results for the HPE framework were established in [11], and these results depend on the distance of the initial iterate to the solution set instead of the diameter of the feasible set. Applications of the HPE framework to the iteration complexity analysis of several zero-order (resp., first-order) methods for solving monotone variational inequalities and monotone inclusions (resp., SP problems) are discussed in [11] and in the subsequent papers [10, 12]. More specifically, by viewing Korpelevich's method [9] as well as Tseng's modified forward-backward splitting (MF-BS) method [22] as special cases of the HPE framework, the authors have established in [10, 11] the pointwise and ergodic iteration complexities of these methods applied to monotone variational inequalities, monotone inclusions consisting of the sum of a Lipschitz continuous monotone map and a maximal monotone operator with an easily computable resolvent, and convex-concave SP problems.

A framework of block-decomposition (BD) prox-type algorithms is introduced in [12] for solving the monotone inclusion problem consisting of the sum of a continuous monotone map and a point-to-set maximal monotone operator with a separable two-block structure. The above BD framework is a special case of the HPE framework which approximately solves the proximal subproblem corresponding to the two-block inclusion by (possibly approximately) solving two smaller proximal single-block subproblems. When the stepsize is sufficiently small, the latter two subproblems can be approximately solved by performing a (single) step similar to the one performed by the gradient method in the case of composite convex optimization or by Korpelevich's and/or Tseng's MF-BS method in the more general context of variational inequality and maximal monotone inclusions. More recently, the authors have studied in [8] an accelerated BD prox-type algorithm for solving the SP (and, more generally, Nash equilibrium) problem where the above two proximal subproblems (which in this case are composite convex optimization) are solved by an accelerated variant of Nesterov's optimal method. The accelerated BD method is generally able to take a much larger stepsize than those for the aforementioned BD methods and, as a consequence, performs a significantly lower number of outer iterations. Moreover, computational results have shown that the accelerated BD method can substantially outperform the aforementioned methods on many relevant classes of SP and Nash equilibrium instances.

Given proper closed convex (possibly nonsmooth) functions g_1 and g_2 defined in finite dimensional inner product spaces, this paper considers the class of composite convex-concave SP problem

$$\min_{x \in X} \max_{y \in Y} \Psi(x, y) := f(x) + \langle Ax, y\rangle + g_1(x) - g_2(y), \tag{1.4}$$

or, equivalently, the problem

$$\min_{x \in X} f(x) + g_1(x) + g_2^*(Ax), \tag{1.5}$$

where $X := \operatorname{dom} g_1$, $Y := \operatorname{dom} g_2$, A is a linear operator, and f is a differentiable convex function whose gradient is L_f-Lipschitz continuous on X. It is assumed that g_1 and g_2 are simple functions in the sense that subproblems of the form

$$\min_{x\in X} \frac{1}{2}\|x-\tilde{x}\|^2 + \lambda g_1(x) \quad \text{and} \quad \min_{y\in Y} \frac{1}{2}\|y-\tilde{y}\|^2 + \lambda g_2(y) \tag{1.6}$$

are easy to solve for any $\tilde{x}$, $\tilde{y}$, and $\lambda > 0$.

Since (1.4) is well known to be equivalent to the monotone inclusion problem (1.1) with T given by

$$T(x,y) = \partial(\Psi(\cdot,y) - \Psi(x,\cdot))(x,y), \tag{1.7}$$

where Ψ is defined in problem (1.4), any instance of the HPE method, including those already discussed above, can be used to solve it. In particular, by taking a sufficiently small stepsize λ, Korpelevich's (resp., Tseng's) method is able to approximately solve the current proximal subproblem (i.e., a triple satisfying (1.3)) by solving at most four (resp., two) subproblems of the form (1.6).

This paper presents an accelerated instance of the HPE framework which arbitrarily chooses the stepsize λ and solves (1.3) with T given by (1.7) by using a Nesterov accelerated variant for smooth composite SP problems. Both the outer (i.e., HPE) iteration complexity and the inner (i.e., accelerated variant) iteration complexity are derived for the method in terms of a general stepsize λ. As in Tseng's and Korpelevich's methods, just a few (namely, three) subproblems of the form (1.6) are solved within an inner iteration. Hence, choosing λ so as to minimize the overall number of inner iterations is the best strategy toward minimizing the overall complexity of the accelerated HPE method. An explicit formula in terms of $\|A\|$, L_f, the distance d_0 of the initial iterate to the set of saddle-points of (1.4), and the specified tolerances is then derived for such a stepsize. Clearly, since d_0 is not known a priori, the above stepsize cannot be computed, but an alternative stepsize λ depending only on $\|A\|$ and L_f is provided which is optimal for the most common SP problems of the form (1.4). Moreover, when the feasible set $X \times Y$ is bounded, the expression for the above optimal stepsize with d_0 replaced by the diameter of $X \times Y$ yields another stepsize which implies (if an appropriate choice of inner product in the (x,y)-space is made) an overall complexity for the accelerated HPE method that is similar to that of Nesterov's smoothing technique (see [14]) for finding an ε-saddle-point of (1.4) (see (5.9) below). It is worth emphasizing that, in contrast to Nesterov's smoothing technique of [14], our accelerated method for solving (1.4) does not assume that $X \times Y$ is bounded due to its proximal point nature.

Our paper is organized as follows. Section 2 contains three subsections which provide the necessary background material for our presentation. More specifically, subsection 2.1 presents the notation and basic definitions used in the paper. Subsection 2.2 reviews a Nesterov accelerated variant for solving composite convex optimization problems. Subsection 2.3 discusses the HPE framework for the monotone inclusion problem. Section 3 reviews the definition of the SP problem, its connection to the composite convex-concave min-max problem, and the notion of an approximate saddle-point. Moreover, this section specializes the HPE framework to the context of the SP problem and states its convergence properties. Section 4 presents a scheme for finding a solution of (1.3) with T given by (1.4) and (1.7) (and without loss of generality (w.l.o.g.) $\lambda = 1$) based on the Nesterov's accelerated variant of subsection 2.2 applied to an associated composite convex-concave min-max problem. Section 5

presents a special instance of the HPE framework based on the accelerated scheme of section 4 for solving the composite convex-concave min-max problem (1.4) and derives its ergodic outer and overall inner iteration complexities for finding approximate saddle-points. It also discusses optimal ways of choosing the stepsize so as to minimize the overall ergodic inner iteration complexity of the accelerated HPE method for solving (1.4). Finally, numerical results are presented in section 6 showing that the new method outperforms Nesterov's smooth approximation scheme [14] on three classes of composite convex-concave min-max problems of the form (1.4).

1.1. Previous most related works. In the context of variational inequalities, Nemirovski [13] has established the ergodic iteration complexity of an extension of Korpelevich's method [9], namely, the Mirror-prox algorithm, under the assumption that the feasible set of the problem is bounded.

Nesterov's smoothing scheme [14] solves problem (1.4) under the assumption that X and Y are compact convex sets and g_1 is the indicator function of X. It consists of first approximating the objective function of (1.4) by a convex differentiable function with Lipschitz continuous gradient and then applying an accelerated gradient-type method (see, e.g., [14, 2, 23]) to the resulting approximation problem. It is shown that, if the approximation is properly chosen, the above scheme obtains an ε solution of (1.4) in at most

$$\mathcal{O}\left(\frac{\|A\|}{\varepsilon}D_X D_Y + \sqrt{\frac{L_f}{\varepsilon}}D_X\right)$$

iterations, where D_X and D_Y are the diameters of X and Y. The latter bound is also known to be optimal (see, for example, the discussion in paragraph (1) of subsection 1.1 of [5]).

Chambolle and Pock [4] have developed and established the convergence rate for a primal-dual method for solving problem (1.4) in the context of $f(x)$ being simple and g_1 being the indicator function of the feasible set X. Extensions of Chambolle and Pock's algorithm are also studied in [6, 7, 24]. A more recent paper [5] considers problem (1.4) with g_1 being the indicator function of the feasible set X and proposed an accelerated primal-dual algorithm that achieved an optimal convergence rate for both cases that the feasible set of the problem is bounded or unbounded.

2. Preliminaries. This section contains three subsections. The first presents the notation and basic definitions that will be used in the paper. The second subsection reviews a variant of Nesterov's accelerated method for the composite convex optimization problem. The third subsection describes the HPE framework for the monotone inclusion problem.

2.1. Notation and basic definitions. We denote the sets of real numbers by $\Re$. For a matrix $W \in \Re^{m\times n}$, we denote its Frobenius norm by $\|W\|$. Let $\mathcal{S}^n$ represent the linear space of $n \times n$ real symmetric matrices. For a matrix $W \in \mathcal{S}^n$, we denote its largest eigenvalue by $\theta_{\max}(W)$. Let $\lceil z \rceil$ denote the smallest integer not less than $z \in \Re$. The nth unit simplex $\Delta_n \subseteq \Re^n$ is defined as

$$\Delta_n = \left\{ z \in \Re^n : \sum_{i=1}^n z_i = 1, z_i \geq 0, i = 1, \ldots, n \right\}. \tag{2.1}$$

Throughout this paper, we let $\mathcal{Z}$ denote a finite dimensional inner product space with associated inner product denoted by $\langle \cdot, \cdot \rangle$ and the induced norm denoted by $\|\cdot\|$.

For a given set $\Omega \subset \mathcal{Z}$, the diameter D_Ω of Ω is defined as

$$D_\Omega := \sup\{\|z - z'\| : z, z' \in \Omega\}, \tag{2.2}$$

and the indicator function $\mathcal{I}_\Omega : \mathcal{Z} \to (-\infty, \infty]$ of Ω is defined as

$$\mathcal{I}_\Omega(z) = \begin{cases} 0, & z \in \Omega, \\ \infty, & z \notin \Omega. \end{cases}$$

Also, if Ω is a nonempty closed convex set, the *orthogonal projection* $P_\Omega : \mathcal{Z} \to \mathcal{Z}$ onto Ω is defined as

$$P_\Omega(z) = \operatorname{argmin}_{z' \in \Omega} \|z' - z\| \quad \forall z \in \mathcal{Z}.$$

A relation $T \subseteq \mathcal{Z} \times \mathcal{Z}$ can be identified with a point-to-set operator $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ in which

$$T(z) := \{r \in \mathcal{Z} : (z, r) \in T\} \quad \forall z \in \mathcal{Z}.$$

Note that the relation T is then the same as the graph of the point-to-set operator T defined as

$$Gr(T) := \{(z, r) \in \mathcal{Z} \times \mathcal{Z} : r \in T(z)\}.$$

An operator $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is *monotone* if

$$\langle r - \tilde{r}, z - \tilde{z} \rangle \geq 0 \qquad \forall (z, r), (\tilde{z}, \tilde{r}) \in Gr(T).$$

Moreover, T is *maximal monotone* if it is monotone and maximal in the family of monotone operators with respect to the partial order of inclusion; i.e., $S : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is monotone and $Gr(S) \supset Gr(T)$ implies that $S = T$. Given a scalar ε, the ε-enlargement of a point-to-set operator $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is the point-to-set operator $T^\varepsilon : \mathcal{Z} \rightrightarrows \mathcal{Z}$ defined as

$$T^\varepsilon(z) = \{r \in \mathcal{Z} \mid \langle z - \tilde{z}, r - \tilde{r} \rangle \geq -\varepsilon \quad \forall \tilde{z} \in \mathcal{Z},\ \forall \tilde{r} \in T(\tilde{z})\} \quad \forall z \in \mathcal{Z}. \tag{2.3}$$

The effective domain of a function $f : \mathcal{Z} \to [-\infty, \infty]$ is defined as $\operatorname{dom} f := \{z \in \mathcal{Z} : f(z) < \infty\}$. A function $f : \mathcal{Z} \to [-\infty, \infty]$ is said to be proper if $\operatorname{dom} f \neq \emptyset$ and $f(z) > -\infty$ for every z. Moreover, if f is differentiable at point $\tilde{z}$ such that $f(\tilde{z}) \in \Re$, its first-order (affine) approximation at $\tilde{z}$ is defined as

$$l_f(z; \tilde{z}) : = f(\tilde{z}) + \langle \nabla f(\tilde{z}), z - \tilde{z} \rangle \qquad \forall z \in \mathcal{Z}. \tag{2.4}$$

If f is in addition convex, the following inequality holds for all $z \in \mathcal{Z}$:

$$l_f(z; \tilde{z}) \leq f(z). \tag{2.5}$$

The conjugate f^* of f is the function $f^* : \mathcal{Z} \to [-\infty, \infty]$ defined as

$$f^*(r) = \sup_{z \in \mathcal{Z}} \langle r, z \rangle - f(z) \quad \forall r \in \mathcal{Z}.$$

Let a proper function $f : \mathcal{Z} \to (-\infty, +\infty]$ be given. Given a scalar $\varepsilon \geq 0$, the ε-subdifferential of f is the operator $\partial_\varepsilon f : \mathcal{Z} \rightrightarrows \mathcal{Z}$ defined as

$$\partial_\varepsilon f(z) = \{r \mid f(\tilde{z}) \geq f(z) + \langle \tilde{z} - z, r \rangle - \varepsilon\ \forall \tilde{z} \in \mathcal{Z}\} \quad \forall z \in \mathcal{Z}. \tag{2.6}$$

When $\varepsilon = 0$, the operator $\partial_\varepsilon f$ is simply denoted by ∂f and is referred to as the subdifferential of f. The operator ∂f is trivially monotone and is maximal monotone whenever f is closed convex [15].

The following result lists some useful properties about the ε-subdifferential of a proper convex function.

PROPOSITION 2.1. *Let a proper function $f : \mathcal{Z} \to (-\infty, +\infty]$ be given. Then*

(a) *if $r \in \partial f(z)$ and $f(\tilde{z}) \in \Re$, then $r \in \partial_\varepsilon f(\tilde{z})$, where $\varepsilon := f(\tilde{z}) - [f(z) + \langle \tilde{z} - z, r\rangle] \geq 0$; and*

(b) *if f is a closed convex function, then $r \in \partial f(z)$ is equivalent to $z \in \partial f^*(r)$.*

The domain of a point-to-point map F is denoted by $\operatorname{Dom} F$. For a constant $L \geq 0$, a map $F : \operatorname{Dom} F \subseteq \mathcal{Z} \to \mathcal{Z}$ is said to be L-Lipschitz continuous on $\Omega \subseteq \operatorname{Dom} F$ if

$$\|F(z) - F(\tilde{z})\| \leq L\|z - \tilde{z}\| \quad \forall z, \tilde{z} \in \Omega;$$

moreover, if in addition $\Omega = \operatorname{Dom} F$, we will simply say that F is L-Lipschitz continuous.

In this paper, a strongly convex function with modulus $\beta > 0$ is referred to as a β-strongly convex function. Moreover, the terminology "0-strongly convex function" is used to refer to a convex function. This terminology has the benefit of allowing us to treat both the convex and the strongly convex cases simultaneously.

The following result gives a characterization of a β-strongly convex function where $\beta > 0$ in terms of its conjugate.

PROPOSITION 2.2. *For a scalar $\beta > 0$ and a proper closed convex function $f : \mathcal{Z} \to [-\infty, \infty]$, the following two properties are equivalent:*

(a) *f is a β-strongly convex function;*

(b) *f^* is differentiable everywhere and ∇f^* is $1/\beta$-Lipschitz continuous.*

Proof. This proposition is equivalent to Proposition 12.60 of [17] in view of the well-known fact that $f = f^{**}$. ∎

2.2. Accelerated method for composite convex optimization. This subsection reviews a variant of Nesterov's accelerated first-order method [14, 23] for solving the composite convex optimization problem.

Let $\mathcal{X}$ denote a finite dimensional inner product space with associated inner product and norm denoted by $\langle \cdot, \cdot \rangle_{\mathcal{X}}$ and $\|\cdot\|_{\mathcal{X}}$, respectively. Consider the composite convex optimization problem

$$\min p(u) := \psi(u) + g(u), \tag{2.7}$$

where the functions $\psi, g : \mathcal{X} \to [-\infty, \infty]$ satisfy the following conditions:

(A.1) g is a proper closed μ-strongly convex function for some $\mu \geq 0$;

(A.2) ψ is differentiable (hence finite) and convex on a closed convex set $\Omega \supseteq X := \operatorname{dom} g$;

(A.3) the gradient of the function ψ is L-Lipschitz continuous on Ω.

We now explicitly state a variant of Nesterov's accelerated method for solving problem (2.7), which is due to Tseng (see Algorithm 3 in [23]).

[Algorithm 1] A variant of Nesterov's accelerated algorithm:

(0) Let $u_0 \in \mathcal{X}$ be given, and set $\Gamma_0 = 0$, $\tilde{u}_0 = w_0 = P_\Omega(u_0)$, $j = 1$;

(1) let $\Gamma_j > \Gamma_{j-1}$ be such that

$$\Gamma_j(\Gamma_{j-1}\mu + 1) = L(\Gamma_j - \Gamma_{j-1})^2, \tag{2.8}$$

and compute $(u_j, w_j, \tilde{u}_j) \in \Omega \times X \times X$ as

$$u_j := \frac{\Gamma_{j-1}}{\Gamma_j}\tilde{u}_{j-1} + \frac{\Gamma_j - \Gamma_{j-1}}{\Gamma_j} w_{j-1}, \tag{2.9}$$

$$w_j := \operatorname{argmin} \sum_{i=1}^{j} \frac{\Gamma_i - \Gamma_{i-1}}{\Gamma_j} l_\psi(u; u_i) + g(u) + \frac{1}{2\Gamma_j}\|u - u_0\|_{\mathcal{X}}^2, \tag{2.10}$$

$$\tilde{u}_j := \frac{\Gamma_{j-1}}{\Gamma_j}\tilde{u}_{j-1} + \frac{\Gamma_j - \Gamma_{j-1}}{\Gamma_j} w_j; \tag{2.11}$$

(2) set $j \leftarrow j + 1$, and go to step 1.

end

We now make a few remarks about the relationship between the above method and Algorithm 3 of [23]. First, the latter method considers the case in which g is convex, while Algorithm 1 handles the more general case in which g is a strongly convex function with modulus μ. The remarks that follow all refer to the special case of Algorithm 1 above with $\mu = 0$. Second, Algorithm 3 of [23] computes w_j as in (2.10) but with the quadratic term $\|u - u_0\|_{\mathcal{X}}^2/2$ replaced by a general strongly convex function $h(u)$. Third, Algorithm 3 of [23] assumes that X is closed and computes $\tilde{u}_0$ and w_0 as $\tilde{u}_0 = w_0 := \operatorname{argmin}\{h(u) : u \in X\}$. Clearly, in view of the latter two remarks, Algorithm 1 with the assumption that $\mu = 0$ and X is closed is a special case of Algorithm 3 of [23] in which the sequences $\{\theta_j\}$ and $\{\vartheta_j\}$ used by it are given by $\theta_j = \vartheta_j = [1/(L\Gamma_{j+1})]^{1/2}$.

We now state the main technical result from which the convergence rate of the above Nesterov accelerated variant immediately follows. Although its proof is similar to that of Corollary 3(a) of [23], we provide its proof in the appendix for the sake of completeness.

PROPOSITION 2.3. *The sequences $\{\Gamma_j\}$, $\{\tilde{u}_j\}$, and $\{u_j\}$ generated by Algorithm 1 satisfy the following inequalities for any $j \geq 1$:*

$$\Gamma_j \geq \frac{1}{L}\max\left\{\frac{j^2}{4}, \quad \left(1 + \sqrt{\frac{\mu}{4L}}\right)^{2(j-1)}\right\} \tag{2.12}$$

and

$$\Gamma_j p(\tilde{u}_j) \leq \sum_{i=1}^{j} (\Gamma_i - \Gamma_{i-1})[l_\psi(u; u_i) + g(u)] + \frac{1}{2}\|u - u_0\|_{\mathcal{X}}^2 \quad \forall u \in X. \tag{2.13}$$

As a consequence, the sequence $\{l_{\psi,j}\}$ of affine functions defined as

$$l_{\psi,j}(u) := \sum_{i=1}^{j} \frac{\Gamma_i - \Gamma_{i-1}}{\Gamma_j} l_\psi(u; u_i) \quad \forall u \in \mathcal{X} \tag{2.14}$$

satisfies

$$l_{\psi,j} \leq \psi, \qquad p(\tilde{u}_j) \leq l_{\psi,j}(u) + g(u) + \frac{1}{2\Gamma_j}\|u - u_0\|_{\mathcal{X}}^2 \quad \forall u \in X. \tag{2.15}$$

2.3. HPE framework for the monotone inclusion problem. Let $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ be a maximal monotone operator. The monotone inclusion problem for T consists of finding $z \in \mathcal{Z}$ such that

$$0 \in T(z). \tag{2.16}$$

We also assume throughout this subsection that this problem has a solution, that is, that $T^{-1}(0) \neq \emptyset$.

We next review the HPE framework introduced in [18] for solving the above problem and state the iteration complexity results obtained for it in [11].

[HPE] Hybrid proximal extragradient framework:

(0) Let $z_0 \in \mathcal{Z}$ and $0 \leq \sigma < 1$ be given, and set $k = 1$;

(1) choose $\lambda_k > 0$, and find $\tilde{z}_k, \tilde{r}_k \in \mathcal{Z}$, $\sigma_k \in [0, \sigma]$, and $\varepsilon_k \geq 0$ such that

$$\tilde{r}_k \in T^{\varepsilon_k}(\tilde{z}_k), \quad \|\lambda_k \tilde{r}_k + \tilde{z}_k - z_{k-1}\|^2 + 2\lambda_k \varepsilon_k \leq \sigma_k^2 \|\tilde{z}_k - z_{k-1}\|^2; \tag{2.17}$$

(2) set $z_k = z_{k-1} - \lambda_k \tilde{r}_k$, set $k \leftarrow k + 1$, and go to step 1.

end

We now make several remarks about the HPE framework. First, the HPE framework does not specify how to choose λ_k and how to find $\tilde{z}_k$, $\tilde{r}_k$, and ε_k as in (2.17). The particular choice of λ_k and the algorithm used to compute $\tilde{z}_k$, $\tilde{r}_k$, and ε_k will depend on the particular implementation of the method and the properties of the operator T. Second, if $\tilde{z} := (\lambda_k T + I)^{-1} z_{k-1}$ is the *exact* proximal point iterate or, equivalently,

$$\tilde{r} \in T(\tilde{z}), \tag{2.18}$$

$$\lambda_k \tilde{r} + \tilde{z} - z_{k-1} = 0 \tag{2.19}$$

for some $\tilde{r} \in \mathcal{Z}$, then $(\tilde{z}_k, \tilde{r}_k) = (\tilde{z}, \tilde{r})$ and $\varepsilon_k = 0$ satisfies (2.17). Therefore, the error criterion (2.17) relaxes the inclusion (2.18) to $\tilde{r} \in T^{\varepsilon}(\tilde{z})$ and relaxes (2.19) by allowing a small error relative to $\|\tilde{z}_k - z_{k-1}\|$.

We define a sequence of ergodic means $\{\tilde{z}_k^a\}$ associated with $\{\tilde{z}_k\}$ as

$$\tilde{z}_k^a := \frac{1}{\Lambda_k} \sum_{i=1}^{k} \lambda_i \tilde{z}_i, \qquad \text{where} \qquad \Lambda_k := \sum_{i=1}^{k} \lambda_i, \tag{2.20}$$

and define the sequences of ergodic residuals $\{\tilde{r}_k^a\}$ and $\{\varepsilon_k^a\}$ as

$$\tilde{r}_k^a := \frac{1}{\Lambda_k} \sum_{i=1}^{k} \lambda_i \tilde{r}_i, \qquad \varepsilon_k^a := \frac{1}{\Lambda_k} \sum_{i=1}^{k} \lambda_i (\varepsilon_i + \langle \tilde{z}_i - \tilde{z}_k^a, \tilde{r}_i - \tilde{r}_k^a \rangle). \tag{2.21}$$

The following result describes the pointwise and ergodic convergence rate properties of the HPE framework. Its proof can be found in Theorem 4.4, Lemma 4.5, and Theorem 4.7 of [11].

THEOREM 2.4. *Let d_0 denote the distance of z_0 to $T^{-1}(0)$. Then, for every $k \in \mathbb{N}$, the following statements hold:*

(a) *(Pointwise convergence rate) $\tilde{r}_k \in T^{\varepsilon_k}(\tilde{z}_k)$, and there exists an index $i \leq k$ such that*

$$\|\tilde{r}_i\| \leq d_0 \sqrt{\frac{1+\sigma}{1-\sigma} \left(\frac{1}{\sum_{j=1}^{k} \lambda_j^2} \right)}, \qquad \varepsilon_i \leq \frac{\sigma^2 d_0^2 \lambda_i}{2(1-\sigma^2) \sum_{j=1}^{k} \lambda_j^2}.$$

(b) *(Ergodic convergence rate)* $\varepsilon_k^a \geq 0$, $\tilde{r}_k^a \in T^{\varepsilon_k^a}(\tilde{z}_k^a)$, *and*

$$\|\tilde{r}_k^a\| \leq \frac{2d_0}{\Lambda_k}, \qquad \varepsilon_k^a \leq \frac{2d_0^2}{\Lambda_k}\left(1 + \frac{\sigma}{\sqrt{(1-\sigma^2)}}\right).$$

3. HPE framework for saddle-point problem. The section reviews the definition of the saddle-point (SP) problem, its connection to the composite convex-concave min-max problem, and the notion of an approximate saddle-point. Moreover, this section specializes the HPE framework to the context of the SP problem and states its convergence properties.

Throughout this paper, we let $\mathcal{X}$ be the finite dimensional inner product space as described in subsection 2.2 and $\mathcal{Y}$ denote a finite dimensional inner product space with associated inner product denoted by $\langle \cdot, \cdot \rangle_{\mathcal{Y}}$ and associated norm denoted by $\| \cdot \|_{\mathcal{Y}}$. We endow the product space $\mathcal{X} \times \mathcal{Y}$ with the canonical inner product defined as

$$\langle (x,y), (x',y') \rangle = \langle x, x' \rangle_{\mathcal{X}} + \langle y, y' \rangle_{\mathcal{Y}} \quad \forall (x,y), (x',y') \in \mathcal{X} \times \mathcal{Y}. \tag{3.1}$$

The associated norm, denoted by $\| \cdot \|$ for brevity, is then given by

$$\|(x,y)\| = \sqrt{\|x\|_{\mathcal{X}}^2 + \|y\|_{\mathcal{Y}}^2} \quad \forall (x,y) \in \mathcal{X} \times \mathcal{Y}.$$

We will now review the SP problem and some of its basic properties. Given two nonempty convex sets $X \subseteq \mathcal{X}$ and $Y \subseteq \mathcal{Y}$, we consider throughout this section a function $\Psi : \mathcal{X} \times \mathcal{Y} \to [-\infty, +\infty]$ satisfying the following condition.

(B.1) $\Psi(x,y)$ is finite-valued on $X \times Y$ and

$$\Psi(x,y) = \begin{cases} \infty, & x \notin X, \\ -\infty, & x \in X, y \notin Y. \end{cases} \tag{3.2}$$

The SP problem determined by the triple $(\Psi; X, Y)$, denoted by $SP(\Psi; X, Y)$, consists of finding a pair $(x,y) \in X \times Y$ such that

$$\Psi(x,y') \leq \Psi(x,y) \leq \Psi(x',y) \quad \forall (x',y') \in X \times Y. \tag{3.3}$$

Clearly, (x,y) is a saddle-point of $SP(\Psi; X, Y)$ if and only if $(x,y) \in X \times Y$ and

$$(0,0) \in T(x,y) := \partial[\Psi(\cdot, y) - \Psi(x, \cdot)](x,y). \tag{3.4}$$

Define the primal and dual functions $p : X \to (-\infty, +\infty]$ and $d : Y \to [-\infty, +\infty)$, respectively, as

$$p(\tilde{x}) = \sup_{\tilde{y} \in Y} \Psi(\tilde{x}, \tilde{y}), \qquad d(\tilde{y}) = \inf_{\tilde{x} \in X} \Psi(\tilde{x}, \tilde{y}) \qquad \forall (\tilde{x}, \tilde{y}) \in X \times Y, \tag{3.5}$$

and consider the pair of optimization problems associated with $SP(\Psi; X, Y)$:

$$p_* := \inf_{\tilde{x} \in X} p(\tilde{x}) = \inf_{\tilde{x} \in X} \sup_{\tilde{y} \in Y} \Psi(\tilde{x}, \tilde{y}) \tag{3.6}$$

and

$$d_* := \sup_{\tilde{y} \in Y} d(\tilde{y}) = \sup_{\tilde{y} \in Y} \inf_{\tilde{x} \in X} \Psi(\tilde{x}, \tilde{y}). \tag{3.7}$$

Then, the weak duality inequality says that

$$p(\tilde{x}) \geq d(\tilde{y}) \quad \forall (\tilde{x}, \tilde{y}) \in X \times Y. \tag{3.8}$$

Moreover, it is well known that (x, y) is a saddle-point if and only if $(x, y) \in X \times Y$ and $p(x) = d(y)$. In view of (3.8), the latter condition is equivalent to $x \in X$ and $y \in Y$ being optimal solutions of (3.6) and (3.7), respectively, and the optimal duality gap $p_* - d_*$ being equal to zero.

We now give a definition of an approximate saddle-point.

DEFINITION 3.1. *Given $(\rho, \varepsilon) \in \Re_+ \times \Re_+$, $z = (x, y) \in X \times Y$, $r \in \mathcal{X} \times \mathcal{Y}$, and $\tilde{\varepsilon} \in \Re_+$, the triple $(z, r, \tilde{\varepsilon})$ is called a (ρ, ε)-saddle-point of Ψ if $\|r\| \leq \rho$, $\tilde{\varepsilon} \leq \varepsilon$, and*

$$r \in \partial_{\tilde{\varepsilon}}[\Psi(\cdot, y) - \Psi(x, \cdot)](x, y). \tag{3.9}$$

Moreover, the pair $(z, \tilde{\varepsilon})$ is a called an ε-saddle-point if $(z, 0, \tilde{\varepsilon})$ is a $(0, \varepsilon)$-saddle-point.

Before describing a special case of the HPE framework for solving the SP problem, we introduce two more assumptions.

(B.2) $\Psi(\cdot, y)$ and $-\Psi(x, \cdot)$ are proper closed convex functions for every $(x, y) \in X \times Y$;

(B.3) the inclusion (3.4) has a solution, i.e., $T^{-1}(0) \neq \emptyset$.

A function $\Psi : \mathcal{X} \times \mathcal{Y} \to [-\infty, +\infty]$ satisfying conditions (B.1) and (B.2) for some nonempty convex sets X and Y is called a *closed convex-concave function on $X \times Y$*. It is well known that its associated map T defined in (3.4) is maximal monotone (see, for example, Theorem 6.3.2 in [1]).

We are ready to state a special case of the HPE framework for solving the monotone inclusion problem (3.4), and hence the SP problem $SP(\Psi; X, Y)$.

[SP-HPE] Hybrid proximal extragradient framework for solving $SP(\Psi; X, Y)$:

(0) Let $(x_0, y_0) \in \mathcal{X} \times \mathcal{Y}$, $\lambda > 0$, and $0 \leq \sigma < 1$ be given, and set $k = 1$;

(1) find $(\tilde{x}_k, \tilde{y}_k) \in \mathcal{X} \times \mathcal{Y}$, $\tilde{r}_k = (\tilde{r}_k^x, \tilde{r}_k^y) \in \mathcal{X} \times \mathcal{Y}$, and $\varepsilon_k \geq 0$ such that

$$(\tilde{r}_k^x, \tilde{r}_k^y) \in \partial_{\varepsilon_k}[\Psi(\cdot, \tilde{y}_k) - \Psi(\tilde{x}_k, \cdot)](\tilde{x}_k, \tilde{y}_k), \tag{3.10}$$

$$\begin{aligned} &\|\lambda \tilde{r}_k^x + \tilde{x}_k - x_{k-1}\|_{\mathcal{X}}^2 + \|\lambda \tilde{r}_k^y + \tilde{y}_k - y_{k-1}\|_{\mathcal{Y}}^2 + 2\lambda\varepsilon_k \\ &\leq \sigma^2 \left(\|\tilde{x}_k - x_{k-1}\|_{\mathcal{X}}^2 + \|\tilde{y}_k - y_{k-1}\|_{\mathcal{Y}}^2 \right); \end{aligned} \tag{3.11}$$

(2) set $x_k = x_{k-1} - \lambda \tilde{r}_k^x$, $y_k = y_{k-1} - \lambda \tilde{r}_k^y$, and $k \leftarrow k + 1$, and go to step 1.

end

We now make several remarks about the SP-HPE framework. First, due to Lemma 3.2 below, the SP-HPE framework is a special case of the HPE framework in which $\lambda_k := \lambda$. In fact, the SP-HPE framework could be stated in terms of a sequence of variable stepsizes $\{\lambda_k\}$, but we assume for simplicity that $\lambda_k = \lambda$. Second, similar to the HPE framework, the SP-HPE framework does not specify how to find $(\tilde{x}_k, \tilde{y}_k)$, $\tilde{r}_k$, and ε_k satisfying the HPE error condition in (3.10) and (3.11). Section 5 describes a special instance of the SP-HPE framework in which $(\tilde{x}_k, \tilde{y}_k)$, $\tilde{r}_k$, and ε_k are obtained by a variant of Nesterov's accelerated method. Third, using the fact that the inclusion in (3.10) is stronger than the inclusion in (2.17), we derive in Theorem 3.4 a finer version of Theorem 2.4 with $\lambda_k = \lambda$ specialized to the context of the SP problem (3.3).

Before stating the pointwise and ergodic convergence rate results for the SP-HPE framework, we give two preliminary technical results.

LEMMA 3.2. *For each $(x,y) \in X \times Y$ and $\varepsilon \geq 0$, we have*

$$\partial_\varepsilon(\Psi(\cdot,y) - \Psi(x,\cdot))(x,y) \subseteq T^\varepsilon(x,y),$$

where T is defined in (3.4).

Proof. Let $r \in \partial_\varepsilon(\Psi(\cdot,y) - \Psi(x,\cdot))(x,y)$ be given. This clearly implies that

$$\Psi(\tilde{x},y) - \Psi(x,\tilde{y}) \geq \langle (\tilde{x}-x, \tilde{y}-y), r\rangle - \varepsilon \quad \forall (\tilde{x},\tilde{y}) \in \mathcal{X} \times \mathcal{Y}.$$

On the other hand, it follows from the definition of T in (3.4) that any $\tilde{r} \in T(\tilde{x},\tilde{y})$ satisfies

$$\Psi(x,\tilde{y}) - \Psi(\tilde{x},y) \geq \langle (x-\tilde{x}, y-\tilde{y}), \tilde{r}\rangle.$$

Summing up the above two inequalities, we then conclude that

$$\langle (x-\tilde{x}, y-\tilde{y}), r-\tilde{r}\rangle \geq -\varepsilon \qquad \forall (\tilde{x},\tilde{y}) \in \mathcal{X} \times \mathcal{Y},\ \forall \tilde{r} \in T(\tilde{x},\tilde{y}),$$

and hence that $r \in T^\varepsilon(x,y)$ in view of the the definition of $T^\varepsilon(\cdot)$ in (2.3). ∎

LEMMA 3.3. *Let $X \subseteq \Re^n$ and $Y \subseteq \Re^m$ be given convex sets, and let $\Gamma : X \times Y \to \Re$ be a function such that, for each pair $(x,y) \in X \times Y$, the function $\Gamma(\cdot,y) - \Gamma(x,\cdot) : X \times Y \to \Re$ is convex. Suppose that, for $i = 1,\ldots,k$, $(x_i,y_i) \in X \times Y$ and $(v_i,w_i) \in \Re^n \times \Re^m$ satisfies*

$$(v_i,w_i) \in \partial_{\varepsilon_i}\Big(\Gamma(\cdot,y_i) - \Gamma(x_i,\cdot)\Big)(x_i,y_i).$$

Let $\alpha_1,\ldots,\alpha_k \geq 0$ be such that $\sum_{i=1}^k \alpha_i = 1$, and define

$$(x^a,y^a) = \sum_{i=1}^k \alpha_i(x_i,y_i), \quad (v^a,w^a) = \sum_{i=1}^k \alpha_i(v_i,w_i),$$
$$\varepsilon^a := \sum_{i=1}^k \alpha_i[\varepsilon_i + \langle x_i - x^a, v_i\rangle + \langle y_i - y^a, w_i\rangle].$$

Then, $\varepsilon^a \geq 0$ and

$$(v^a,w^a) \in \partial_{\varepsilon^a}\Big(\Gamma(\cdot,y^a) - \Gamma(x^a,\cdot)\Big)(x^a,y^a). \tag{3.12}$$

The proof of Lemma 3.3 can be found in Proposition 5.1 of [10].

The following result describes the pointwise and ergodic convergence rate properties of the SP-HPE framework.

THEOREM 3.4. *Consider the sequences $\{(\tilde{x}_k,\tilde{y}_k)\}$, $\{\tilde{r}_k\} = \{(\tilde{r}_k^x,\tilde{r}_k^y)\}$, and $\{\varepsilon_k\}$ generated by the SP-HPE framework, and define, for every $k \in \mathbb{N}$,*

$$(\tilde{x}_k^a,\tilde{y}_k^a) := \frac{1}{k}\sum_{i=1}^k (\tilde{x}_i,\tilde{y}_i), \qquad \tilde{r}_k^a := \frac{1}{k}\sum_{i=1}^k (\tilde{r}_i^x,\tilde{r}_i^y), \tag{3.13}$$

and

$$\varepsilon_k^a := \frac{1}{k}\sum_{i=1}^{k}[\varepsilon_i + \langle(\tilde{x}_i - \tilde{x}_k^a, \tilde{y}_i - \tilde{y}_k^a), (\tilde{r}_i^x, \tilde{r}_i^y) - \tilde{r}_k^a\rangle]. \tag{3.14}$$

Let d_0 denote the distance of (x_0, y_0) to the solution set of $SP(\Psi; X, Y)$. Then, for every $k \in \mathbb{N}$, the following statements hold:

(a) *(Pointwise convergence rate) the triple $((\tilde{x}_k, \tilde{y}_k), \tilde{r}_k, \varepsilon_k)$ is an $(\|\tilde{r}_k\|, \varepsilon_k)$-saddle-point of Ψ, or, equivalently,* (3.10) *holds, and there exists an index $i \le k$ such that*

$$\|\tilde{r}_i\| \le \frac{d_0}{\lambda}\sqrt{\frac{1+\sigma}{k(1-\sigma)}}, \qquad \varepsilon_i \le \frac{\sigma^2 d_0^2}{2k\lambda(1-\sigma^2)}. \tag{3.15}$$

(b) *(Ergodic convergence rate) $\varepsilon_k^a \ge 0$, the triple $((\tilde{x}_k^a, \tilde{y}_k^a), \tilde{r}_k^a, \varepsilon_k^a)$ is an $(\|\tilde{r}_k^a\|, \varepsilon_k^a)$-saddle-point of Ψ, or, equivalently,*

$$\tilde{r}_k^a \in \partial_{\varepsilon_k^a}(\Psi(\cdot, \tilde{y}_k^a) - \Psi(\tilde{x}_k^a, \cdot))(\tilde{x}_k^a, \tilde{y}_k^a), \tag{3.16}$$

and

$$\|\tilde{r}_k^a\| \le \frac{2d_0}{\lambda k}, \qquad \varepsilon_k^a \le \frac{2d_0^2}{\lambda k}\left(1 + \frac{\sigma}{\sqrt{(1-\sigma^2)}}\right). \tag{3.17}$$

Proof. The first claim in (a) is obvious. Since, by (3.10) and Lemma 3.2, we have $\tilde{r}_k \in T^{\varepsilon_k}(\tilde{x}_k, \tilde{y}_k)$, where T is defined in (3.4), we conclude that the SP-HPE framework is a special instance of the HPE framework applied to (3.4), where $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$ is endowed with the inner product defined in (3.1). The second claim in (a) then follows Theorem 2.4(a). Moreover, inclusion (3.16) follows from (3.10) and Lemma 3.3, and the bounds in (3.17) follow from Theorem 2.4(b) with $\lambda_k = \lambda$. ∎

4. Solving the HPE error condition. This section presents a scheme, together with its iteration-complexity analysis, for finding a solution of the HPE error condition (3.10)–(3.11) with Ψ given by (1.4) (and w.l.o.g. $\lambda = 1$). The scheme is based on the Nesterov accelerated variant of subsection 2.2 applied to an associated composite convex-concave min-max problem.

This section considers the following problem corresponding to the special case of step 1 of the SP-HPE framework in which $\lambda = 1$.

($P1$) Given convex sets $X \subset \mathcal{X}$ and $Y \subset \mathcal{Y}$, a closed convex-concave function Ψ on $X \times Y$, a pair $(u_0, v_0) \in \mathcal{X} \times \mathcal{Y}$, and a scalar $\sigma \in (0, 1]$, the problem is to find $(\tilde{u}, \tilde{v}) \in \mathcal{X} \times \mathcal{Y}, (\tilde{r}^u, \tilde{r}^v) \in \mathcal{X} \times \mathcal{Y}$ and $\tilde{\varepsilon} \ge 0$ such that

$$(\tilde{r}^u, \tilde{r}^v) \in \partial_{\tilde{\varepsilon}}\left[\Psi(\cdot, \tilde{v}) - \Psi(\tilde{u}, \cdot)\right](\tilde{u}, \tilde{v}), \tag{4.1}$$

$$\|\tilde{r}^u + \tilde{u} - u_0\|_{\mathcal{X}}^2 + \|\tilde{r}^v + \tilde{v} - v_0\|_{\mathcal{Y}}^2 + 2\tilde{\varepsilon} \le \sigma^2\left(\|\tilde{u} - u_0\|_{\mathcal{X}}^2 + \|\tilde{v} - v_0\|_{\mathcal{Y}}^2\right). \tag{4.2}$$

This section presents a scheme based on the Nesterov accelerated variant of subsection 2.2 for solving problem ($P1$) where Ψ has the bilinear structure

$$\Psi(u, v) = f(u) + \langle Au, v\rangle + g_1(u) - g_2(v) \quad \forall (u, v) \in X \times Y \tag{4.3}$$

and the following conditions hold.

(C.1) $A : \mathcal{X} \to \mathcal{Y}$ is a linear operator.
(C.2) $g_1 : \mathcal{X} \to [-\infty, \infty]$ and $g_2 : \mathcal{Y} \to [-\infty, \infty]$ are proper closed convex functions such that $\operatorname{dom} g_1 = X$ and $\operatorname{dom} g_2 = Y$.
(C.3) f is convex on a closed convex set $\Omega \supseteq X$.
(C.4) f is differentiable on Ω, and ∇f is L_f-Lipschitz continuous on Ω.

We now make two remarks about problem $(P1)$. First, finding the solution of the exact version of problem $(P1)$, i.e., the one in which $\sigma = 0$, is equivalent to finding the unique saddle-point of

$$\min_{u \in X} \max_{v \in Y} \Psi(u, v) + \frac{1}{2}\|u - u_0\|^2 - \frac{1}{2}\|v - v_0\|^2 \tag{4.4}$$

where Ψ is given by (4.3). More specifically, if $(\tilde{u}, \tilde{v})$ is the exact saddle-point of the above problem, then $(\tilde{u}, \tilde{v})$ and the quantities $(\tilde{r}^u, \tilde{r}^v) := (u_0 - \tilde{u}, v_0 - \tilde{v})$ and $\tilde{\varepsilon} := 0$ satisfy (4.1) and (4.2) with $\sigma = 0$. Second, although the above SP problem has essentially the same structure as the one we are interested in solving, namely, (1.4), its primal function (see (3.5)) has the key property that it is the composite sum of the easy convex nonsmooth function g_1 and a smooth convex function with Lipschitz continuous gradient. Hence, approximate solutions of (4.4) can be obtained by using a Nesterov accelerated variant for composite convex optimization problems (e.g., the one in subsection 2.2).

In view of the two observations above, it is reasonable to expect that approximate solutions of (4.4) yield solutions of problem $(P1)$ (with $\sigma > 0$). Rather than tackling the latter issue in an abstract setting, we instead propose a scheme based on the Nesterov accelerated variant of subsection 2.2 applied to (4.4) to obtain a solution of problem $(P1)$ and derive its corresponding iteration complexity.

We next discuss how the composite convex-concave min-max problem (4.4) can be viewed as a composite convex optimization problem (2.7) satisfying conditions (A.1)–(A.3). Clearly, (4.4) is a special case of (2.7) in which

$$\psi(u) := f(u) + \tilde{\phi}(u), \qquad g(u) := g_1(u) + \frac{1}{2}\|u - u_0\|_{\mathcal{X}}^2, \tag{4.5}$$

and

$$\tilde{\phi}(u) := \max_v \left\{ \phi(u, v) := \langle Au, v\rangle - g_2(v) - \frac{1}{2}\|v - v_0\|_{\mathcal{Y}}^2 \right\}. \tag{4.6}$$

It is apparent that the above function g satisfies condition (A.1) with $\mu = 1$. The following result implies that the above ψ satisfies conditions (A.2) and (A.3). Its proof for the case in which Y is compact is well known (see, for example, [14]). Since we are not assuming the latter condition, we include for the sake of completeness a simple proof of the more general version given below. Its statement uses the following notion of the induced norm of a linear operator $A : \mathcal{X} \to \mathcal{Y}$ defined as

$$\|A\| := \max_x \{\|Ax\|_{\mathcal{Y}} : \|x\|_{\mathcal{X}} \le 1\}.$$

PROPOSITION 4.1. *The following statements hold:*

(a) *For every $u \in \mathcal{X}$, the maximization problem in (4.6) has a unique optimal solution $v(u)$, i.e.,*

$$v(u) := \arg\max_v \langle Au, v\rangle - g_2(v) - \frac{1}{2}\|v - v_0\|_{\mathcal{Y}}^2; \tag{4.7}$$

(b) *$\tilde{\phi}$ is convex, differentiable everywhere on $\mathcal{X}$, $\nabla\tilde{\phi}$ is $\|A\|^2$-Lipschitz continuous on $\mathcal{X}$, and*

$$\nabla\tilde{\phi}(u) = A^*v(u) \quad \forall u \in \mathcal{X}; \tag{4.8}$$

(c) *for every $u, \tilde{u} \in \mathcal{X}$,*

$$l_{\tilde{\phi}}(u;\tilde{u}) = \phi(u, v(\tilde{u})). \tag{4.9}$$

Proof. (a) This statement follows immediately from the fact that the negative of the objective function of the max problem in (4.7) is proper, closed, and strongly convex.

(b) Letting $\tilde{g}_2(v) := g_2(v) + \|v - v_0\|^2/2$ and using the definition of $\tilde{\phi}$ in (4.6), we easily see that

$$\tilde{\phi}(u) = \tilde{g}_2^*(Au) \quad \forall u \in \mathcal{X}. \tag{4.10}$$

Moreover, noting that $\tilde{g}_2$ is proper, closed, and strongly convex with modulus one, we conclude from Proposition 2.2 with $f = \tilde{g}_2$ that $\tilde{g}_2^*$ is differentiable everywhere on $\mathcal{Y}$ and $\nabla\tilde{g}_2^*$ is 1-Lipschitz continuous. The above two observations then easily imply that $\tilde{\phi}$ is convex, differentiable everywhere on $\mathcal{X}$, and $\nabla\tilde{\phi}$ is $\|A\|^2$-Lipschitz continuous on $\mathcal{X}$. Moreover, the optimality condition for (4.7) implies that $Au \in \partial\tilde{g}_2(v(u))$ and hence that $v(u) = \nabla\tilde{g}_2^*(Au)$ in view of Proposition 2.1(c). Now, (4.8) follows by differentiating (4.10) and using the latter conclusion.

(c) Using (4.8) and the definitions of $l_{\tilde{\phi}}(\cdot,\cdot)$, $\phi(\cdot,\cdot)$, and $v(u)$ in (2.4), (4.6), and (4.7), respectively, we easily see that

$$l_{\tilde{\phi}}(u;\tilde{u}) = \tilde{\phi}(\tilde{u}) + \langle\nabla\tilde{\phi}(\tilde{u}), u - \tilde{u}\rangle = \phi(\tilde{u}, v(\tilde{u})) + \langle A^*v(\tilde{u}), u - \tilde{u}\rangle = \phi(u, v(\tilde{u})). \qquad \square$$

In view of the above result, we conclude that the function ψ defined in (4.5) satisfies conditions (A.2) and (A.3) of subsection 2.2 with $L = L_f + \|A\|^2$. We can then use Algorithm 1 to approximately solve (4.4) and hence $(P1)$, as will be shown later in this section.

We now state our accelerated scheme for solving problem $(P1)$. It is essentially Algorithm 1 applied to (2.7) with Ψ and g given by (4.5) and (4.6), respectively, parameter μ set to 1, and endowed with two important refinements as follows. The first, due to Nesterov (see (4.2) of [14] or Corollary 3(c) of [23]), computes a dual iterate $\tilde{v}_j$ as in (4.11), which, together with the primal iterate $\tilde{u}_j$, provides the first candidate pair $(\tilde{u}, \tilde{v}) = (\tilde{u}_j, \tilde{v}_j)$ for $(P1)$. The second (see step 2 below) gives a recipe for computing the second candidate pair $(\tilde{r}^u, \tilde{r}^v) \in \mathcal{X} \times \mathcal{Y}$ and scalar $\tilde{\varepsilon} \geq 0$, which, together with the above pair $(\tilde{u}, \tilde{v})$, yield a candidate solution for $(P1)$.

[Algorithm 2] Accelerated method for problem ($P1$):

Input: f, L_f, A, g_1, and g_2 as in conditions (C.1)–(C.4), $(u_0, v_0) \in \mathcal{X} \times \mathcal{Y}$, and $\sigma \in (0, 1]$.

(0) set $L = L_f + \|A\|^2$, $\Gamma_0 = 0$, $\tilde{u}_0 = w_0 = P_\Omega(u_0)$, $\tilde{v}_0 = 0$, and $j = 1$;

(1) compute Γ_j, u_j, and $v(u_j)$ as in (2.8) with $\mu = 1$, (2.9), and (4.7), respectively, $(\tilde{v}_j, w_j) \in Y \times X$ as

$$\tilde{v}_j := \frac{\Gamma_{j-1}}{\Gamma_j}\tilde{v}_{j-1} + \frac{\Gamma_j - \Gamma_{j-1}}{\Gamma_j} v(u_j), \tag{4.11}$$

$$w_j := \operatorname{argmin}_u l_{f,j}(u) + \langle A^*\tilde{v}_j, u\rangle + g_1(u) + \frac{c_j}{2}\|u - u_0\|_{\mathcal{X}}^2, \tag{4.12}$$

and $\tilde{u}_j$ as in (2.11), where

$$c_j := 1 + \frac{1}{\Gamma_j}, \qquad l_{f,j}(u) := \sum_{i=1}^{j} \frac{\Gamma_i - \Gamma_{i-1}}{\Gamma_j} l_f(u; u_i); \tag{4.13}$$

(2) set

$$\tilde{\varepsilon}_j = \frac{1}{2\Gamma_j}\|\tilde{u}_j - u_0\|_{\mathcal{X}}^2, \quad \tilde{r}_j^u := c_j(u_0 - w_j), \quad \tilde{r}_j^v := v_0 - v(\tilde{u}_j); \tag{4.14}$$

(3) if $\|\tilde{r}_j^u + \tilde{u}_j - u_0\|_{\mathcal{X}}^2 + \|\tilde{r}_j^v + \tilde{v}_j - v_0\|_{\mathcal{Y}}^2 + 2\tilde{\varepsilon}_j \le \sigma^2\|\tilde{u}_j - u_0\|_{\mathcal{X}}^2 + \sigma^2\|\tilde{v}_j - v_0\|_{\mathcal{Y}}^2$, then terminate; otherwise, set $j \leftarrow j + 1$, and go to step 1.

Output: Output $(\tilde{u}, \tilde{v}) = (\tilde{u}_j, \tilde{v}_j)$, $(\tilde{r}^u, \tilde{r}^v) = (\tilde{r}_j^u, \tilde{r}_j^v)$, and $\tilde{\varepsilon} = \tilde{\varepsilon}_j$.

The following simple result shows that step 1 of Algorithm 2 corresponds to an iteration of Algorithm 1 applied to (2.10) with ψ and g defined according to (4.5) and (4.6).

LEMMA 4.2. *Let ψ and g be defined according to* (4.5) *and* (4.6). *Then, the following statements hold for every $j \ge 1$:*

(a) *The function $l_{\psi,j}(u) - (l_{f,j}(u) + \langle A^*\tilde{v}_j, u\rangle)$ is constant where $l_{\psi,j}$ and $l_{f,j}$ are defined in* (2.14) *and* (4.13)*;*

(b) (4.12) *is equivalent to* (2.10).

Proof. (a) Relation (4.11) and the fact that $\Gamma_0 = 0$ imply that

$$\tilde{v}_j = \sum_{i=1}^{j} \frac{\Gamma_i - \Gamma_{i-1}}{\Gamma_j} v(u_i). \tag{4.15}$$

Using the first identity in (4.5) and Proposition 4.1(b), we have that $\nabla\psi(u) = \nabla f(u) + A^*v(u)$, which together with definition (2.4) then implies that

$$l_\psi(u; u_i) = l_f(u; u_i) + [\tilde{\phi}(u_i) + \langle A^*v(u_i), u - u_i\rangle] \quad \forall i \ge 1.$$

Statement (a) now follows from the previous identity and relations (2.14), (4.13), and (4.15).

(b) This statement immediately follows from (a), the definition of g in (4.5), and the definition of c_j in (4.13). ∎

Before establishing the iteration-complexity of Algorithm 2 for solving problem ($P1$), we first make the following two remarks. First, ignoring steps 2 and 3 of

Algorithm 2, which are essentially computing $(\tilde{r}^u, \tilde{r}^v) = (\tilde{r}^u_j, \tilde{r}^v_j)$ and $\tilde{\varepsilon} = \tilde{\varepsilon}_j$ satisfying (4.1) and checking whether these entities together with the primal-dual iterate $(\tilde{u}_j, \tilde{v}_j)$ satisfy (4.2), Lemma 4.2 immediately implies that Algorithm 2 is nothing more than Algorithm 1 applied to problem (2.7) with ψ and g given by (4.5). Second, there is no reason for us to have specifically chosen the accelerated gradient variant of subsection 2.2, namely, Algorithm 1, as a basis for developing Algorithm 2. In fact, any accelerated gradient variant for solving the composite convex optimization problem (2.7) satisfying properties (2.12) and (2.13) could have been used in place of Algorithm 1.

We now proceed to establish the iteration-complexity of Algorithm 2. The following technical result follows as a consequence of the first observation above and Proposition 2.3.

LEMMA 4.3. *Consider the sequences* $\{(\tilde{u}_j, \tilde{v}_j)\}$ *generated by Algorithm* 2*, and define*

$$\tilde{\varepsilon}'_j := \frac{1}{2\Gamma_j}\|\tilde{u}_j - u_0\|^2_{\mathcal{X}} + l_{f,j}(\tilde{u}_j) - f(\tilde{u}_j), \tag{4.16}$$

$$\Psi_j(u,v) := l_{f,j}(u) + \langle Au, v\rangle + g_1(u) - g_2(v), \tag{4.17}$$

$$q_j(u,v) := \frac{c_j}{2}\|u - u_0\|^2_{\mathcal{X}} + \frac{1}{2}\|v - v_0\|^2_{\mathcal{Y}}, \tag{4.18}$$

where c_j *and* $l_{f,j}$ *are defined in* (4.13)*. Then,*

$$0 \in \partial_{\tilde{\varepsilon}'_j}\left[\Psi_j(\cdot, \tilde{v}_j) - \Psi_j(\tilde{u}_j, \cdot) + q_j(\cdot,\cdot)\right](\tilde{u}_j, \tilde{v}_j). \tag{4.19}$$

Proof. Consider the functions ψ, g, and ϕ defined in (4.5) and (4.6). It follows from (4.5), (4.6), and Proposition 2.3 that

$$\begin{aligned} f(\tilde{u}_j) + \phi(\tilde{u}_j, v) + g_1(\tilde{u}_j) + \frac{1}{2}\|\tilde{u}_j - u_0\|^2_{\mathcal{X}} &\le (\psi + g)(\tilde{u}_j) \\ &\le l_{\psi,j}(u) + g_1(u) + \frac{c_j}{2}\|u - u_0\|^2_{\mathcal{X}} \quad \forall (u,v) \in \mathcal{X}\times\mathcal{Y}, \end{aligned}$$

where $l_{\psi,j}(\cdot)$ is defined in (2.14). Using the definitions of ψ and $\tilde{\phi}$ in (4.5) and (4.6), relation (2.4), the definitions of $l_{\psi,j}(u)$ and $l_{f,j}(u)$ in (2.14) and (4.13), the identities (4.9) and (4.15), and the fact that $\phi(u,\cdot)$ is concave for any $u \in \mathcal{X}$, we conclude that

$$\begin{aligned} l_{\psi,j}(u) &= \sum_{i=1}^{j} \frac{\Gamma_i - \Gamma_{i-1}}{\Gamma_j}\left(l_f(u;u_i) + l_{\tilde{\phi}}(u;u_i)\right) = l_{f,j}(u) + \sum_{i=1}^{j}\frac{\Gamma_i - \Gamma_{i-1}}{\Gamma_j}\phi(u, v(u_i)) \\ &\le l_{f,j}(u) + \phi\left(u, \sum_{i=1}^{j}\frac{\Gamma_i - \Gamma_{i-1}}{\Gamma_j}v(u_i)\right) = l_{f,j}(u) + \phi(u,\tilde{v}_j) \quad \forall u \in \mathcal{X}. \end{aligned}$$

Combining the above two relations and using the definition of ϕ, Ψ_j, and $\tilde{\varepsilon}'_j$ in (4.6), (4.17), and (4.16), respectively, we then conclude that

$$\begin{aligned} \Psi_j(\tilde{u}_j, v) - \frac{1}{2}\|v - v_0\|^2_{\mathcal{Y}} + \frac{c_j}{2}\|\tilde{u}_j - u_0\|^2_{\mathcal{X}} - \tilde{\varepsilon}'_j & \\ = l_{f,j}(\tilde{u}_j) + \phi(\tilde{u}_j, v) + g_1(\tilde{u}_j) + \frac{c_j}{2}\|\tilde{u}_j - u_0\|^2_{\mathcal{X}} - \tilde{\varepsilon}'_j & \\ \le l_{f,j}(u) + \phi(u,\tilde{v}_j) + g_1(u) + \frac{c_j}{2}\|u - u_0\|^2_{\mathcal{X}} & \\ = \Psi_j(u,\tilde{v}_j) - \frac{1}{2}\|\tilde{v}_j - v_0\|^2_{\mathcal{Y}} + \frac{c_j}{2}\|u - u_0\|^2_{\mathcal{X}} \quad \forall (u,v) \in X \times Y. & \end{aligned} \tag{4.20}$$

Now, using the definition of the ε-differential in (2.6) and the definition of $q_j(\cdot,\cdot)$ in (4.18), the above inequality can be easily seen to be equivalent to (4.19). ∎

The following result quantifies the quality of the entities $(\tilde{u}_j, \tilde{v}_j)$, $\tilde{\varepsilon}_j$, and $(\tilde{r}_j^u, \tilde{r}_j^v)$ generated at the jth iteration of Algorithm 2 as a candidate solution for problem $(P1)$.

LEMMA 4.4. *Consider the sequences* $\{(\tilde{u}_j, \tilde{v}_j)\}$, $\{\tilde{\varepsilon}_j\}$, *and* $\{(\tilde{r}_j^u, \tilde{r}_j^v)\}$ *generated by Algorithm* 2. *Then, for every* $j \geq 1$,

$$(\tilde{r}_j^u, \tilde{r}_j^v) \in \partial_{\tilde{\varepsilon}_j} \left[\Psi(\cdot, \tilde{v}_j) - \Psi(\tilde{u}_j, \cdot)\right](\tilde{u}_j, \tilde{v}_j), \tag{4.21}$$

$$\|\tilde{r}_j^u + \tilde{u}_j - u_0\|_{\mathcal{X}}^2 + \|\tilde{r}_j^v + \tilde{v}_j - v_0\|_{\mathcal{Y}}^2 + 2\tilde{\varepsilon}_j \leq \left(\frac{3}{\Gamma_j} + \frac{4}{\Gamma_j^2}\right)\|\tilde{u}_j - u_0\|_{\mathcal{X}}^2, \tag{4.22}$$

where $\Psi(\cdot)$ *is as defined in* (4.3).

Proof. Equations (4.7) and (4.12) and the definitions of Ψ_j and q_j in (4.17) and (4.18) imply that

$$(w_j, v(\tilde{u}_j)) = \arg\min_{(u,v)} \Psi_j(u, \tilde{v}_j) - \Psi_j(\tilde{u}_j, v) + q_j(u, v).$$

In view of the optimality condition of the above minimization problem, the definitions of $\tilde{r}_j^u$ and $\tilde{r}_j^v$ in (4.14), and the definition of $q_j(\cdot,\cdot)$ in (4.18), we then conclude that

$$(\tilde{r}_j^u, \tilde{r}_j^v) = -\nabla q_j(w_j, v(\tilde{u}_j)) \in \partial[\Psi_j(\cdot, \tilde{v}_j) - \Psi_j(\tilde{u}_j, \cdot)](w_j, v(\tilde{u}_j)).$$

Hence, by Proposition 2.1(a) we have

$$(\tilde{r}_j^u, \tilde{r}_j^v) = -\nabla q_j(w_j, v(\tilde{u}_j)) \in \partial_{\delta_j}[\Psi_j(\cdot, \tilde{v}_j) - \Psi_j(\tilde{u}_j, \cdot)](\tilde{u}_j, \tilde{v}_j), \tag{4.23}$$

where

$$\delta_j := -\left[\Psi_j(w_j, \tilde{v}_j) - \Psi_j(\tilde{u}_j, v(\tilde{u}_j))\right] - \langle -\nabla q_j(w_j, v(\tilde{u}_j)), (\tilde{u}_j, \tilde{v}_j) - (w_j, v(\tilde{u}_j))\rangle \geq 0.$$

On the other hand, in view of Lemma 4.3, inclusion (4.19) holds, or, equivalently, inequality (4.20) holds. The latter inequality with $(u, v) = (w_j, v(\tilde{u}_j))$, together with the definitions of $\tilde{\varepsilon}_j$ and $\tilde{\varepsilon}'_j$ in (4.14) and (4.16), then implies that

$$\begin{aligned} \tilde{\varepsilon}_j \geq \tilde{\varepsilon}'_j &\geq -\Psi_j(w_j, \tilde{v}_j) + \Psi_j(\tilde{u}_j, v(\tilde{u}_j)) + q_j(\tilde{u}_j, \tilde{v}_j) - q_j(w_j, v(\tilde{u}_j)) \\ &= \delta_j + \frac{c_j}{2}\|\tilde{u}_j - w_j\|_{\mathcal{X}}^2 + \frac{1}{2}\|\tilde{v}_j - v(\tilde{u}_j)\|_{\mathcal{Y}}^2, \end{aligned} \tag{4.24}$$

where the last equality comes from the definition of δ_j and the fact that the second-order Taylor expansion of the quadratic function q_j at an arbitrary point agrees with q_j itself. In view of (4.23) and (4.24), we then conclude that

$$(\tilde{r}_j^u, \tilde{r}_j^v) \in \partial_{\tilde{\varepsilon}'_j}[\Psi_j(\cdot, \tilde{v}_j) - \Psi_j(\tilde{u}_j, \cdot)](\tilde{u}_j, \tilde{v}_j).$$

Using the definition of the ε-subdifferential in (2.6), the definitions of $\tilde{\varepsilon}_j$, $\tilde{\varepsilon}'_j$, Ψ, and Ψ_j in (4.14), (4.16), (4.3), and (4.17), respectively, and the fact that $\Psi_j(\cdot, \tilde{v}_j)$ is majorized by $\Psi(\cdot, \tilde{v}_j)$, it is now easy to see that the above inclusion implies (4.21).

Moreover, inequality (4.24), the definitions of $\tilde{r}_j^u$, $\tilde{r}_j^v$, and $\tilde{\varepsilon}_j$ in (4.14), and the fact that $c_j = 1 + 1/\Gamma_j$ imply that

$$\begin{aligned}
&\|\tilde{r}_j^u + \tilde{u}_j - u_0\|_{\mathcal{X}}^2 + \|\tilde{r}_j^v + \tilde{v}_j - v_0\|_{\mathcal{Y}}^2 + 2\tilde{\varepsilon}_j \\
=&\|(u_0 - \tilde{u}_j)/\Gamma_j + c_j(\tilde{u}_j - w_j)\|_{\mathcal{X}}^2 + \|\tilde{v}_j - v(\tilde{u}_j)\|_{\mathcal{X}}^2 + 2\tilde{\varepsilon}_j \\
\le&\frac{2}{\Gamma_j^2}\|\tilde{u}_j - u_0\|_{\mathcal{X}}^2 + 2c_j^2\|\tilde{u}_j - w_j\|_{\mathcal{X}}^2 + \|\tilde{v}_j - v(\tilde{u}_j)\|_{\mathcal{X}}^2 + 2\tilde{\varepsilon}_j \\
\le&\frac{2}{\Gamma_j^2}\|\tilde{u}_j - u_0\|_{\mathcal{X}}^2 + (4c_j + 2)\tilde{\varepsilon}_j = \left(\frac{3}{\Gamma_j} + \frac{4}{\Gamma_j^2}\right)\|\tilde{u}_j - u_0\|_{\mathcal{X}}^2. \qquad \square
\end{aligned}$$

As an immediate consequence of Lemma 4.4, we can now derive the iteration-complexity for Algorithm 2 to solve problem $(P1)$.

PROPOSITION 4.5. *Algorithm 2 terminates with an output that solves problem* $(P1)$ *in at most*

$$\mathcal{O}\left(1 + \min\left\{\frac{\sqrt{L}}{\sigma}, (1+\sqrt{L})\log^+\left(\frac{\sqrt{L}}{\sigma}\right)\right\}\right) \tag{4.25}$$

iterations where $L := L_f + \|A\|^2$.

Proof. The inclusion (4.21) and the termination criterion in step 3 of Algorithm 2 show that the output of Algorithm 2 solves problem $(P1)$. To complete the proof of the proposition, it suffices to show that Algorithm 2 finishes in at most

$$j_0 := \left\lceil \min\left\{4\frac{\sqrt{L}}{\sigma}, 1 + (1+2\sqrt{L})\log^+\left(\frac{2\sqrt{L}}{\sigma}\right)\right\}\right\rceil \tag{4.26}$$

iterations since such j_0 has the order of (4.25). In view of Lemma 4.4, the latter conclusion will follow if we show that

$$\left(\frac{3}{\Gamma_{j_0}} + \frac{4}{\Gamma_{j_0}^2}\right) \le \sigma^2,$$

which in turn easily follows from the inequality $\Gamma_{j_0} \ge 4/\sigma^2$ in view of the assumption that $\sigma \le 1$. To show the latter inequality, observe that (4.26) and the inequality $\log(1+t) \ge t/(t+1)$ with $t = 1/(2\sqrt{L})$ imply that either

$$\frac{j_0^2}{4L} \ge \frac{4}{\sigma^2} \quad \text{or} \quad \frac{1}{L}\left(1 + \frac{1}{2\sqrt{L}}\right)^{2(j_0-1)} \ge \frac{4}{\sigma^2}.$$

Hence, in view of the conclusion (2.12) of Proposition 2.3 with $\mu = 1$, we then conclude that $\Gamma_{j_0} \ge 4/\sigma^2$. $\square$

In our analysis, we are interested in values of σ such that $\max\{\sigma^{-1}, (1-\sigma)^{-1}\} = \mathcal{O}(1)$. Under this assumption, among the two bounds in (4.26), the first dominates (resp., is of the same order as) the second for relatively large (resp, small) values of L, and hence it is the one we use to derive the overall iteration-complexity of the accelerated instance of the SP-HPE framework discussed in the next section.

5. Accelerated SP-HPE method for problem (1.4). This section presents a special instance of the SP-HPE framework introduced in section 3, which we refer to as the Acc-SP-HPE method, for solving the class of composite convex-concave min-max problem (1.4), or, equivalently, the SP problem $SP(\Psi; X, Y)$ with Ψ as defined in (4.3). Each (outer) iteration of the Acc-SP-HPE method, which is essentially a special iteration of the SP-HPE framework, invokes Algorithm 2 to obtain a solution of the inexact prox subproblem (3.10)–(3.11). This section contains two subsections. A complexity bound on the total number of Algorithm 2 iterations (called the inner iterations) performed by the Acc-SP-HPE method to find a (ρ, ε)-saddle-point is derived in subsection 5.1. Moreover, an inner-iteration complexity for the Acc-SP-HPE method to find an ε-saddle-point for the case where the feasible set $X \times Y$ is bounded is derived in subsection 5.2 .

We assume in this section that the solution set of the composite convex-concave min-max problem (1.4) is nonempty and assumptions (C.1)–(C.4) are satisfied.

5.1. The accelerated SP-HPE method and its complexity analysis. This subsection describes the accelerated SP-HPE method and its corresponding complexity results for the case where the feasible set $X \times Y$ is possibly unbounded.

Recall that in section 4 we have motivated the introduction of problem $(P1)$ as a special case of the inexact prox subproblem (3.10)–(3.11) in which $\lambda = 1$. The following result shows, in fact, that problem $(P1)$ is as general as subproblem (3.10)–(3.11) for any value of $\lambda > 0$.

PROPOSITION 5.1. *Let $\lambda > 0$ and a closed convex-concave function Ψ be given, and consider the kth iteration of the SP-HPE framework. If $(\tilde{u}, \tilde{v}) \in \mathcal{X} \times \mathcal{Y}$, $(\tilde{r}^u, \tilde{r}^v) \in \mathcal{X} \times \mathcal{Y}$, and $\tilde{\varepsilon} \geq 0$ solve problem $(P1)$ with input $\Psi = \lambda\Psi$, $(u_0, v_0) = (x_{k-1}, y_{k-1})$, and $\sigma > 0$, then*

$$(\tilde{x}_k, \tilde{y}_k) := (\tilde{u}, \tilde{v}), \quad (\tilde{r}_k^x, \tilde{r}_k^y) := \frac{1}{\lambda}(\tilde{r}^u, \tilde{r}^v), \quad \varepsilon_k := \frac{1}{\lambda}\tilde{\varepsilon}$$

satisfy conditions (3.10) *and* (3.11) *of step* 1 *of the SP-HPE framework.*

Proof. The conclusion follows immediately from the identity

$$\lambda \partial_\varepsilon \left[\Psi(\cdot, \tilde{v}) - \Psi(\tilde{u}, \cdot)\right](\tilde{u}, \tilde{v}) = \partial_{\lambda\varepsilon} \left[\lambda\Psi(\cdot, \tilde{v}) - \lambda\Psi(\tilde{u}, \cdot)\right](\tilde{u}, \tilde{v}),$$

which holds for every $\varepsilon \geq 0$, $\lambda > 0$, and $(\tilde{u}, \tilde{v}) \in \mathcal{X} \times \mathcal{Y}$. ∎

In view of the above result, we can use Algorithm 2 to solve the inexact prox subproblem (3.10)–(3.11). This is the key idea behind the following special case of the SP-HPE framework, referred to as the Acc-SP-HPE method, for solving the SP problem (1.4).

[Acc-SP-HPE] Accelerated SP-HPE method for solving problem (1.4):

(0) Let $(x_0, y_0) \in \mathcal{X} \times \mathcal{Y}$, $\lambda > 0$, and $0 < \sigma < 1$ be given, and set $k = 1$;

(1) invoke Algorithm 2 with input

$$f = \lambda f,\ A = \lambda A,\ g_1 = \lambda g_1,\ g_2 = \lambda g_2,\ (u_0, v_0) = (x_{k-1}, y_{k-1}),\ L_f = \lambda L_f,$$

and set

$$(\tilde{x}_k, \tilde{y}_k) := (\tilde{u}, \tilde{v}), \quad \tilde{r}_k = (\tilde{r}_k^x, \tilde{r}_k^y) := \frac{1}{\lambda}(\tilde{r}^u, \tilde{r}^v), \quad \varepsilon_k := \frac{1}{\lambda}\tilde{\varepsilon},$$

where $(\tilde{u}, \tilde{v})$, $(\tilde{r}^u, \tilde{r}^v)$, and $\tilde{\varepsilon}$ are the output generated by Algorithm 2;

(2) set $x_k = x_{k-1} - \lambda \tilde{r}_k^x$, $y_k = y_{k-1} - \lambda \tilde{r}_k^y$, set $k \leftarrow k + 1$, and go to step 1.

end

PROPOSITION 5.2. *The Acc-SP-HPE method is a special case of the SP-HPE framework for solving the composite convex-concave min-max problem* (1.4).

Proof. In view of Proposition 5.1, the sequences $\{(\tilde{x}_k, \tilde{y}_k)\}$, $\{(\tilde{r}_k^x, \tilde{r}_k^y)\}$, and $\{\varepsilon_k\}$ generated by the Acc-SP-HPE method satisfy the conditions (3.10) and (3.11) of step 1 of the SP-HPE framework. Therefore, the Acc-SP-HPE method is clearly a special case of the SP-HPE framework. ∎

It follows as a consequence of Proposition 5.2 that the pointwise and ergodic (outer) convergence rate bounds for the Acc-SP-HPE method are as described in statements (a) and (b) of Theorem 3.4, respectively. In particular, the following result follows as a consequence of the ergodic convergence rate derived in Theorem 3.4(b).

THEOREM 5.3. *Assume that* $\max\{\sigma^{-1}, (1-\sigma)^{-1}\} = \mathcal{O}(1)$*, and let* d_0 *denote the distance of the initial iterate* (x_0, y_0) *of the Acc-SP-HPE method with respect to the (convex) set of saddle-points of* (1.4)*. Consider the sequences* $\{(\tilde{x}_k, \tilde{y}_k)\}$*,* $\{(\tilde{r}_k^x, \tilde{r}_k^y)\}$*, and* $\{\varepsilon_k\}$ *generated by the Acc-SP-HPE method and the ergodic sequences* $\{(\tilde{x}_k^a, \tilde{y}_k^a)\}$*,* $\{\tilde{r}_k^a\}$*, and* $\{\varepsilon_k^a\}$ *defined in Theorem* 3.4*. Then, the following statements hold:*

(a) *for every pair of positive scalars* (ρ, ε)*, there exists*

$$k_0 = \mathcal{O}\left(\max\left\{1, \frac{d_0}{\lambda\rho}, \frac{d_0^2}{\lambda\varepsilon}\right\}\right)$$

such that for every $k \geq k_0$*, the triple* $((\tilde{x}_k^a, \tilde{y}_k^a), \tilde{r}_k^a, \varepsilon_k^a)$ *is a* (ρ, ε)*-saddle-point of* (1.4)*;*

(b) *each iteration of the Acc-SP-HPE method performs at most*

$$\mathcal{O}\left(\left\lceil \sqrt{\lambda L_f + \lambda^2 \|A\|^2} \right\rceil\right)$$

inner iterations (and hence resolvent evaluations of ∂g_1 *and* ∂g_2*).*

As a consequence, the Acc-SP-HPE method finds a (ρ, ε)*-saddle-point of* (1.4) *by performing no more than*

$$\mathcal{O}\left(\left\lceil \sqrt{(\lambda L_f + \lambda^2 \|A\|^2)} \right\rceil \max\left\{1, \frac{d_0}{\lambda\rho}, \frac{d_0^2}{\lambda\varepsilon}\right\}\right) \tag{5.1}$$

inner iterations (and hence resolvent evaluations of ∂g_1 *and* ∂g_2*).*

Proof. Since by Proposition 5.2 the Acc-SP-HPE method is a special instance of the SP-HPE framework, (a) follows immediately from Theorem 3.4(b). Statement (b)

follows immediately from Proposition 4.5 with $L_f = \lambda L_f$ and $A = \lambda A$, and the fact that each iteration of Algorithm 2 performs one resolvent evaluation of ∂g_1 and two resolvent evaluations of ∂g_2. The last assertion of the theorem follows immediately from (a) and (b). ∎

We now make some remarks about possible values of λ which minimize the complexity bound (5.1) (up to an additive and multiplicative $\mathcal{O}(1)$ constant). Noting that (5.1) is equivalent to

$$\mathcal{O}\left(\max\left\{\frac{1}{\lambda}, \sqrt{\frac{L_f}{\lambda}}, \|A\|\right\} \max\left\{\lambda, \frac{d_0}{\rho}, \frac{d_0^2}{\varepsilon}\right\}\right)$$

and assuming that $A \neq 0$, it is straightforward to see that the following claims hold depending on whether the condition

$$\lambda_1 := \max\left\{\frac{L_f}{\|A\|^2}, \frac{1}{\|A\|}\right\} \leq \max\left\{\frac{d_0}{\rho}, \frac{d_0^2}{\varepsilon}\right\} =: \lambda_2 \tag{5.2}$$

holds (case (1)) or not (case (2)):

(1) if (5.2) holds, then any $\lambda \in [\lambda_1, \lambda_2]$ minimizes (5.1) with minimum value equal to

$$\mathcal{O}\left(\|A\| \max\left\{\frac{d_0}{\rho}, \frac{d_0^2}{\varepsilon}\right\}\right);$$

(2) otherwise, if $\lambda_1 > \lambda_2$, then $\lambda = \lambda_2$ minimizes (5.1) with minimum value equal to

$$\mathcal{O}\left(1 + \sqrt{L_f} \max\left\{\sqrt{\frac{d_0}{\rho}}, \frac{d_0}{\sqrt{\varepsilon}}\right\}\right).$$

Ideally, one should choose λ according to the above discussion in order to minimize the total number of resolvent evaluations of ∂g_1 and ∂g_2. But, since d_0 is usually not known a priori, we cannot compute λ_2 and as a result choose $\lambda = \lambda_2$ as proposed in case (2) above. Note, however, that we can always choose $\lambda = \lambda_1$ since the latter is easily computable. Clearly, this choice is optimal when case (1) holds and, even though not optimal when case (2) holds, we believe it might be a good practical choice in both cases due to the fact that case (2) is quite unlikely.

5.2. Specialized complexity bounds for bounded feasible sets. This subsection considers the special case of problem (1.4) where the feasible set $X \times Y$ is bounded and derives a complexity bound on the number of inner iterations performed by the Acc-SP-HPE method to find an ε-saddle-point of (1.4).

COROLLARY 5.4. *Suppose that the assumptions of Theorem* 5.3 *hold,* $(x_0, y_0) \in X \times Y$*, and the diameter* D *of the set* $X \times Y$ *defined in* (2.2) *is finite. Then, for any* $\varepsilon > 0$*, the Acc-SP-HPE method finds an* ε*-saddle-point of* (1.4) *by performing no more than*

$$\mathcal{O}\left(\max\left\{1, \frac{d_0 D}{\lambda \varepsilon}\right\}\right) \tag{5.3}$$

outer iterations and no more than

$$\mathcal{O}\left(\left\lceil\sqrt{(\lambda L_f + \lambda^2\|A\|^2)}\right\rceil \max\left\{1, \frac{d_0 D}{\lambda \varepsilon}\right\}\right) \leq \mathcal{O}\left(\left\lceil\sqrt{(\lambda L_f + \lambda^2\|A\|^2)}\right\rceil \max\left\{1, \frac{D^2}{\lambda \varepsilon}\right\}\right) \tag{5.4}$$

inner iterations (and hence resolvent evaluations of ∂g_1 and ∂g_2).

Proof. Under the assumption that D is finite, it is straightforward to see from Definition 3.1 and the definition of the subdifferential that an $(\varepsilon/2D, \varepsilon/2)$-saddle-point is always an ε-saddle-point. The first bound in (5.4) now follows immediately from the fact that $d_0 \le D$ in view of the assumption that $(x_0, y_0) \in X \times Y$, and from the bound (5.1) in Theorem 5.3 with $(\rho, \varepsilon) = (\varepsilon/(2D), \varepsilon/2)$. Clearly, $d_0 \le D$ also implies the second bound in (5.4). ∎

We now make a few comments about choosing λ so as to minimize the right-hand side of (5.4) (up to an additive and multiplicative $\mathcal{O}(1)$ constant). Similar to the discussion in the previous subsection, if

$$\widehat{\lambda}_1 := \max\left\{\frac{L_f}{\|A\|^2}, \frac{1}{\|A\|}\right\} \le \frac{D^2}{\varepsilon} =: \widehat{\lambda}_2 \tag{5.5}$$

holds, then any $\lambda \in [\widehat{\lambda}_1, \widehat{\lambda}_2]$ minimizes the right-hand side of (5.4) with minimum value equal to $\mathcal{O}(1 + D^2\|A\|/\varepsilon)$. Otherwise, if $\widehat{\lambda}_1 > \widehat{\lambda}_2$, then $\lambda = \widehat{\lambda}_2$ minimizes the right-hand side of (5.4) with minimum value equal to $\mathcal{O}(1 + D\sqrt{L_f/\varepsilon})$. Observe that regardless of which case holds, the right-hand side of (5.4) assumes its minimum value

$$\mathcal{O}\left(1 + D^2\frac{\|A\|}{\varepsilon} + D\sqrt{\frac{L_f}{\varepsilon}}\right) \tag{5.6}$$

when $\lambda = \min\{\widehat{\lambda}_1, \widehat{\lambda}_2\}$.

Clearly, letting D_X and D_Y denote the diameter of X and Y, we have $D = (D_X^2 + D_Y^2)^{1/2}$. Hence, we have $D_X \le D$ and $D_X D_Y \le D^2/2$, and it is clearly possible that $D_X << D$ and/or $D_X D_Y << D^2/2$. The rest of this subsection shows that the Acc-SP-HPE method applied to problem (1.4) with $\mathcal{X}$ and $\mathcal{Y}$ endowed with suitable scaled inner products has a resolvent complexity similar to (5.6) but with D^2 in the first term replaced by $D_X D_Y$ and D in the second term replaced by D_X.

To achieve the above goal, we endow $\mathcal{X}$ and $\mathcal{Y}$ with new inner products

$$\langle\cdot,\cdot\rangle_{\mathcal{X},\theta} := \theta\langle\cdot,\cdot\rangle, \quad \langle\cdot,\cdot\rangle_{\mathcal{Y},\theta} := \theta^{-1}\langle\cdot,\cdot\rangle, \tag{5.7}$$

respectively, where $\theta > 0$ is a constant. The associated norms then become

$$\|\cdot\|_{\mathcal{X},\theta} := \theta^{1/2}\|\cdot\|_{\mathcal{X}}, \quad \|\cdot\|_{\mathcal{Y},\theta} := \theta^{-1/2}\|\cdot\|_{\mathcal{Y}},$$

and problem (1.4) becomes

$$\min_{x\in X}\max_{y\in Y} \Psi(x,y) = f(x) + \langle A_\theta x, y\rangle_{\mathcal{Y},\theta} + g_1(x) - g_2(y), \tag{5.8}$$

where $A_\theta := \theta A$. Moreover, $\|A_\theta\|_\theta = \|A\|$ where $\|C\|_\theta := \max_x\{\|Cx\|_{\mathcal{Y},\theta} : \|x\|_{\mathcal{X},\theta} \le 1\}$ and the gradient of f with respect to $\langle\cdot,\cdot\rangle_{\mathcal{X},\theta}$ is $L_{f,\theta}$-Lipschitz continuous on Ω where $L_{f,\theta} = \theta^{-1}L_f$. Also, the diameter of the feasible set $X \times Y$ with the product space $\mathcal{X} \times \mathcal{Y}$ endowed with the Cartesian inner product $\langle\cdot,\cdot\rangle_{\mathcal{X},\theta} + \langle\cdot,\cdot\rangle_{\mathcal{Y},\theta}$ is

$$D_\theta^2 := \theta D_X^2 + \theta^{-1}D_Y^2.$$

Using the above observations, and assuming that $A \neq 0$ and $\min\{D_X, D_Y\} > 0$, we immediately see that the Acc-SP-HPE method applied to problem (1.4) where θ and λ are chosen as

$$\theta = \frac{D_Y}{D_X}, \qquad \lambda = \min\left\{\max\left\{\frac{L_f D_X}{\|A\|^2 D_Y}, \frac{1}{\|A\|}\right\}, \frac{2D_X D_Y}{\varepsilon}\right\},$$

and $\mathcal{X}$ and $\mathcal{Y}$ are endowed with the inner products (5.7), computes an ε-saddle-point of (1.4) by performing no more than

$$\mathcal{O}\left(1 + \frac{\|A\|}{\varepsilon} D_X D_Y + \sqrt{\frac{L_f}{\varepsilon}} D_X\right) \tag{5.9}$$

resolvent evaluations of ∂g_1 and ∂g_2.

We end this subsection with some concluding remarks. First, the above complexity is the same as that obtained for the Nesterov smoothing method (see (4.4) in [14]). Second, when λ is sufficiently large, i.e., $\lambda = \Theta(d_0 D/\varepsilon)$, it follows from Corollary 5.4 that the Acc-SP-HPE method performs only one outer iteration and hence basically consists of iterations of Algorithm 2 applied to the perturbed problem (4.4) with $(u_0, v_0) = (x_0, y_0)$. Thus, in the latter case, the Acc-SP-HPE method has some similarity to the Nesterov smoothing method, although it is worth observing that they are based on slightly different perturbation problems. Third, the Acc-SP-HPE method with a wide range of values of λ is shown to have the same complexity of Nesterov's smoothing method for a large class of relevant instances of problem (1.4).

Finally, from a practical point of view, we believe that the convergence guarantee of the Acc-SP-HPE method for any stepsize λ allows for more suitable choices of stepsize (e.g., $\lambda = \Theta(\max\{L_f D_X/(\|A\|^2 D_Y), \|A\|^{-1}\})$) other than those given by $\lambda = \Theta(D_X D_Y/\varepsilon)$, which depends on the tolerance ε.

6. Numerical experiments. This section presents computational results showing the numerical performance of the Acc-SP-HPE method on a collection of convex optimization problems that are either in the form of, or can be easily reformulated as, (1.4). All the computational results were obtained using MATLAB R2013b on a quad-core 3.20GHz Linux machine with 16GB memory.

The Acc-SP-HPE method is compared with three other methods, namely, (i) Nesterov's smooth approximation scheme [14] (referred to as Nest-app), where the smooth approximation is solved by the Nesterov's accelerated variant introduced in Subsection 2.2; (ii) the accelerated primal-dual method (referred to as APD) proposed in [5]; and (iii) the primal-dual splitting method (referred to as PD splitting) proposed in [7]. For the sake of a fair comparison, we have implemented Nest-app and APD with $\mathcal{X}$ and $\mathcal{Y}$ endowed with the Euclidean (or Frobenius) norm $\|\cdot\|_2$ and based on the distance generating function for $\|\cdot\|_2^2/2$.

The following three subsections report computational results on the following classes of convex optimization problems: (a) zero-sum matrix game; (b) quadratic game; and (c) vector-matrix saddle-point. For all problem classes, all methods are terminated whenever an ε-saddle-point (either $\varepsilon = 10^{-3}$ or $\varepsilon = 10^{-4}$) is obtained. Note that for a given pair (x, y), this termination criterion requires computing $p(x)$ and $d(y)$ and checking whether $p(x) - d(y) \leq \varepsilon$. For the second and the third problem classes, computation of the dual function is not a simple operation since it involves solving a quadratic programming problem over the unit simplex. Hence, the use of this criterion is not the best strategy from the computational point of view. Note that had we used the more general termination criterion of Definition 3.1, i.e., that of (ρ, ε)-saddle-point, for terminating our method, we would have avoided evaluating $d(y)$. Since the use of the latter termination criterion is not common despite its computational appeal and the fact that the theories developed for the other methods are based on the first termination criterion, we have opted for the first criterion but adopted the convention of excluding the effort to evaluate the dual functions from the reported CPU times.

We now discuss how we select the parameters for the four methods used in our benchmark. Acc-SP-HPE sets $\lambda = \max\{L_f/\|A\|^2, 1/\|A\|\}$ and $\sigma = 0.99$. Our implementation of Nest-app sets the smoothness parameter $\mu = \varepsilon/2D_Y^2$ (see equation (4.8) of [14]). For APD, we have used the code implemented by the authors of [5] with parameters set for solving problems with bounded feasible sets, i.e., according to equation (2.19) of [5]. Our implementation of PD splitting is based on Algorithm 3.1 of [7] with parameters $\tau = \sigma$, where $\sigma > 0$ is such that $1/\sigma - \sigma\|A\|^2 = L_f/2$ and $\rho_n = 0.5$ for all $n \in \mathbb{N}$ (see Theorem 3.1 of [7]).

6.1. Zero-sum matrix game. This subsection compares Acc-SP-HPE with Nest-app, APD, and PD splitting on a collection of instances of the zero-sum matrix game problem

$$\min_{x\in\Delta_m}\max_{y\in\Delta_n}\Psi(x,y) = \langle Ax, y\rangle, \tag{6.1}$$

where $A \in \Re^{n\times m}$.

In the numerical experiment, the matrix A in problem (6.1) is generated such that each entry is nonzero with probability p and each nonzero entry is generated independently and uniformly in the interval $[-1, 1]$. The methods are terminated whenever the duality gap at the iterate $(\tilde{x}_k, \tilde{y}_k)$ is less than a given tolerance ε, i.e.,

$$\max_i(A\tilde{x}_k)_i - \min_i(A^\top\tilde{y}_k)_i \le \varepsilon. \tag{6.2}$$

Table 1 reports the CPU time and the number of (inner) iterations for each method (Acc-SP-HPE). Table 1 shows that the methods Acc-SP-HPE, APD, and PD splitting have roughly similar performance and that they all perform better than Nest-app on the zero-sum game problem.

TABLE 1

Computational results for the methods Acc-SP-HPE, Nest-app, and APD on two-player zero-sum games with different sizes and sparsities. All methods are terminated using a duality gap criterion with tolerance $\varepsilon = 10^{-3}$. CPU time in seconds and number of (inner) iterations are reported for each method.

Problem size			Acc-SP-HPE			Nest-app		APD		PD splitting	
m	n	p	time	#inner	#outer	time	#iter.	time	#iter.	time	#iter.
1000	100	0.01	0.39	358	303	1.34	1720	1.01	3620	0.28	345
1000	100	0.1	0.17	280	275	1.76	3765	0.75	490	0.16	430
1000	1000	0.01	0.42	65	63	4.94	985	1.70	550	0.54	125
1000	1000	0.1	0.86	132	129	12.33	2445	0.69	150	0.73	170
1000	10000	0.01	4.31	62	62	60.42	1150	3.73	90	3.44	75
1000	10000	0.1	10.04	145	144	159.53	3035	5.04	150	9.44	205
10000	100	0.01	2.05	269	262	14.75	2465	1.80	545	1.66	325
10000	100	0.1	5.70	758	749	57.67	9620	2.64	795	4.64	925
10000	1000	0.01	4.21	62	58	62.79	1220	3.15	95	3.60	80
10000	1000	0.1	10.71	157	152	237.68	4570	3.95	135	9.96	215

6.2. Quadratic game problem. This subsection compares Acc-SP-HPE with Nest-app and APD for solving a collection of instances of the quadratic game problem

$$\min_{x\in\Delta_m}\max_{y\in\Delta_n}\frac{1}{2}\|Bx\|^2 + y^\top Ax, \tag{6.3}$$

where $A \in \Re^{n\times m}$ and $B \in \Re^{m\times m}$. In our numerical experiments, the matrices A and B were randomly generated such that each component is nonzero with probability p and each nonzero component is generated independently and uniformly in the interval

TABLE 2

Computational results for the methods Acc-SP-HPE, Nest-app, and APD on two-player quadratic games with different sizes and sparsities. All methods are terminated using a duality gap criterion with tolerance $\varepsilon = 10^{-4}$. CPU time in seconds and number of (inner) iterations are reported for each method.

Problem size			Lip. const.		Acc-SP-HPE			Nest-app		APD		PD splitting	
m	n	p	$\|B\|^2$	$\|A\|$	time	#inner	#outer	time	#iter.	time	#iter.	time	#iter.
200	200	0.1	27.60	5.51	0.05	218	18	1.60	3785	0.30	1140	0.17	470
200	200	0.2	52.20	7.24	0.07	306	16	2.49	5485	0.34	1375	0.22	770
200	200	0.5	129.03	11.23	0.08	351	11	2.94	6320	0.28	980	0.67	1310
200	500	0.1	27.28	6.94	0.13	289	29	3.22	4990	0.44	1360	0.35	525
200	500	0.2	56.20	9.47	0.13	354	24	3.29	5420	0.50	1610	0.39	860
200	500	0.5	127.53	14.82	0.27	693	33	5.07	8340	0.56	1640	3.74	2455
200	1000	0.1	27.47	8.38	0.35	442	52	5.63	5075	0.83	1675	0.57	715
200	1000	0.2	54.01	11.71	0.35	441	41	7.55	6105	0.63	1150	0.86	990
200	1000	0.5	127.62	18.52	0.42	574	34	11.29	9570	1.21	2410	3.19	1610
500	200	0.1	69.12	6.79	0.15	311	11	2.53	3525	0.52	1115	0.52	900
500	200	0.2	133.35	9.51	0.17	318	8	3.52	4620	0.44	810	1.09	1205
500	200	0.5	322.97	14.82	0.32	508	8	5.42	6635	0.69	1475	6.11	2740
500	500	0.1	68.41	8.22	0.42	304	14	5.74	3710	0.78	1025	1.08	925
500	500	0.2	129.32	11.56	0.38	362	12	7.64	5070	0.80	1010	2.11	1295
500	500	0.5	326.28	18.07	0.59	551	11	11.08	7450	1.13	1435	13.13	3000
500	1000	0.1	67.86	10.00	0.73	308	18	17.18	4375	1.42	1095	2.47	800
500	1000	0.2	133.68	13.93	1.07	391	16	17.56	5945	1.55	1155	4.78	1400
500	1000	0.5	328.57	21.88	1.40	564	14	31.18	8030	1.99	1435	24.81	2885
1000	200	0.1	132.84	8.47	0.44	286	6	6.27	3315	0.69	450	1.62	1010
1000	200	0.2	267.33	11.76	0.71	467	7	7.93	4240	1.41	1010	8.74	1960
1000	200	0.5	653.78	18.37	0.90	616	6	0.44	5730	1.62	1080	30.52	4160
1000	500	0.1	134.86	9.89	0.83	277	7	11.63	3445	1.17	550	3.09	1055
1000	500	0.2	266.78	13.90	1.11	397	7	14.16	4280	1.37	650	9.55	1720
1000	500	0.5	663.44	21.75	1.52	506	6	20.74	6235	2.04	900	40.10	3750
1000	1000	0.1	135.68	11.64	1.82	341	11	21.29	3600	1.95	640	6.92	1185
1000	1000	0.2	264.52	16.45	2.20	409	9	30.14	5050	2.75	895	17.51	1845
1000	1000	0.5	669.49	25.82	3.99	740	10	43.30	7375	4.99	1605	101.97	5130

$[-1, 1]$. Table 2 reports the CPU time and the number of (inner) iterations for each method. It shows that the method Acc-SP-HPE is slightly better than APD and that they both perform better than Nest-app and PD splitting on this class of problem.

6.3. Vector-matrix saddle-point problem. This subsection compares Acc-SP-HPE with Nest-app and APD for solving a collection of instances of the minimization problem

$$\min_{x \in \Delta_m} \frac{1}{2}\|Cx - b\|^2 + \theta_{max}(A(x)), \tag{6.4}$$

where $C \in \Re^{m \times m}$, $b \in \Re^m$, $A_1, \ldots, A_m \in \mathcal{S}^n$, and $A(x) = \sum_{i=1}^m x_i A_i \in \mathcal{S}^{n \times n}$. It is easy to verify that the above problem is equivalent to the vector-matrix SP problem

$$\min_{x \in \Delta_m} \max_{y \in \Omega} \Psi(x, y) = \frac{1}{2}\|Cx - b\|^2 + \langle A(x), y \rangle, \tag{6.5}$$

where $\Omega = \{y \in \mathcal{S}^n : tr(y) = 1, y \succeq 0\}$. Hence, we can apply the above methods to the SP problem (6.5). In our numerical experiments, the matrices $A_1, \ldots, A_m$ and C were randomly generated such that each component is nonzero with probability 0.1 and each nonzero component is generated independently and uniformly in the interval $[-1, 1]$ and $A_1, \ldots, A_m$ are then symmetrized. Table 3 reports the CPU time and the number of eigendecompositions (resolvent evaluation of $\partial \mathcal{I}_\Omega$) for each method. It shows that the method Acc-SP-HPE performs better than Nest-app and significantly better than APD and PD splitting on this class of problem.

TABLE 3

Computational results for the methods Acc-SP-HPE, Nest-app, and APD on vector-matrix saddle-point problems (6.5) *with different sizes. All methods are terminated using a duality gap criterion with tolerance* $\varepsilon = 10^{-3}$. *CPU time in seconds and number of eigendecompositions are reported for each method.*

Problem size		Lip. const.		Acc-SP-HPE			Nest-app		APD		PD splitting	
m	n	$\|C\|^2$	$\|A\|$	time	#eigen.	#outer	time	#eigen.	time	#eigen.	time	#eigen.
50	50	62.73	2.74	0.79	447	7	1.78	795	4.39	3830	4.16	3230
50	100	66.13	4.97	3.50	706	21	16.32	2100	17.54	5170	20.62	3785
50	200	58.53	8.81	37.71	1826	121	107.42	4090	158.72	12855	105.84	5730
100	50	122.22	2.80	1.08	494	4	2.31	805	5.83	4315	13.63	6710
100	100	131.84	4.99	10.80	958	13	17.49	1690	29.03	6045	72.17	8595
100	200	124.93	8.88	46.40	1442	42	104.22	2795	134.48	7930	244.05	8415
200	50	259.87	3.17	1.91	678	3	4.77	870	9.27	4350	73.89	14405
200	100	253.96	5.29	19.24	1542	12	23.81	1470	105.36	14915	376.25	26095
200	200	257.27	9.21	110.11	2186	31	140.78	2310	372.69	14650	1207.7	23765

7. Concluding remarks. In this section we make some final remarks about the theoretical and computational results described in this work.

We have shown in section 5 that the Acc-SP-HPE method has the same complexity as the Nesterov smoothing technique of [14]. The experiment results of section 6 involving the three problem sets have shown that the new method Acc-SP-HPE outperforms Nesterov's smoothing technique of [14] and the PD splitting method for problems in which the ratio $L_f/\|A\|$ is significantly large. They also show that the performance of our method on the first two problem classes (resp., third problem class) is comparable to (resp., substantially better than) that of the accelerated primal-dual method of [5].

Both Nesterov's smoothing technique and the APD method can be implemented using the entropy distance-generating function and with $\mathcal{X}$ and $\mathcal{Y}$ endowed with the L_1-norm. In the future, we plan to design a variant of Acc-SP-HPE that can take advantage of the entropy distance-generating function and compare it with the corresponding variants of Nesterov's method and the APD method.

Appendix. Proof of Proposition 2.3. To prove Proposition 2.3, we first prove an intermediate result in Lemma A.1.

LEMMA A.1. *Define*

$$\Pi_0 := \min_{u\in\Omega} \frac{1}{2}\|u-u_0\|_{\mathcal{X}}^2 \tag{A.1}$$

and, for $j \geq 1$,

$$\Pi_j := \min_{u\in\Omega} \left\{ \sum_{i=1}^{j} (\Gamma_i - \Gamma_{i-1})[l_\psi(u;u_i) + g(u)] + \frac{1}{2}\|u-u_0\|_{\mathcal{X}}^2 \right\}. \tag{A.2}$$

Then, for every $j \geq 0$,

$$\Pi_{j+1} - \Pi_j \geq \Gamma_{j+1} p(\tilde{u}_{j+1}) - \Gamma_j p(\tilde{u}_j). \tag{A.3}$$

Proof. Since $\Gamma_0 = 0$ and $g(u)$ is strongly convex with modulus μ, the function in the minimization problem (A.2) is strongly convex with modulus $\Gamma_j\mu + 1$. Therefore, we have

$$\begin{aligned} \Pi_j + \frac{\Gamma_j\mu+1}{2}\|w_j - w_{j+1}\|_{\mathcal{X}}^2 &\leq \sum_{i=1}^{j} (\Gamma_i - \Gamma_{i-1})[l_\psi(w_{j+1};u_i) + g(w_{j+1})] + \frac{1}{2}\|w_{j+1}-u_0\|_{\mathcal{X}}^2 \\ &= \Pi_{j+1} - (\Gamma_{j+1} - \Gamma_j)[l_\psi(w_{j+1};u_{j+1}) + g(w_{j+1})]. \end{aligned} \tag{A.4}$$

Now, using the definition of $\tilde{u}_j$ in (2.11), the definitions (2.4) and (2.7), the convexity of the function $l_\psi(\cdot; u_{j+1}) + g(\cdot)$, and relation (2.5), we have

$$\begin{aligned}
&\Gamma_{j+1}[l_\psi(\tilde{u}_{j+1}; u_{j+1}) + g(\tilde{u}_{j+1})] \\
&\quad \le (\Gamma_{j+1} - \Gamma_j)[l_\psi(w_{j+1}; u_{j+1}) + g(w_{j+1})] + \Gamma_j[l_\psi(\tilde{u}_j; u_{j+1}) + g(\tilde{u}_j)] \\
&\quad \le (\Gamma_{j+1} - \Gamma_j)[l_\psi(w_{j+1}; u_{j+1}) + g(w_{j+1})] + \Gamma_j p(\tilde{u}_j). \qquad \text{(A.5)}
\end{aligned}$$

Using the relation (2.8) and the definitions of u_j and $\tilde{u}_j$ in (2.9) and (2.11), we have

$$\|\tilde{u}_{j+1} - u_{j+1}\|^2 = \frac{(\Gamma_{j+1} - \Gamma_j)^2}{\Gamma_{j+1}^2}\|w_{j+1} - w_j\|^2 = \frac{\Gamma_j\mu + 1}{\Gamma_{j+1}L}\|w_{j+1} - w_j\|^2.$$

Therefore, the equality above and the inequalities (A.4) and (A.5) imply that

$$\Pi_{j+1} - \Pi_j \ge \Gamma_{j+1}[l_\psi(\tilde{u}_{j+1}; u_{j+1}) + g(\tilde{u}_{j+1})] + \frac{\Gamma_{j+1}L}{2}\|\tilde{u}_{j+1} - u_{j+1}\|^2 - \Gamma_j p(\tilde{u}_j).$$

Since ψ is L-Lipschitz continuous on Ω, we have

$$l_\psi(\tilde{u}_{j+1}; u_{j+1}) + \frac{L}{2}\|\tilde{u}_{j+1} - u_{j+1}\|^2 \ge \psi(\tilde{u}_{j+1}),$$

which, together with the above inequality and the definition (2.7), implies (A.3). ∎

Proof of Proposition 2.3. It follows from (A.3) that the sequence $\{\Pi_j - \Gamma_j p(\tilde{u}_j)\}$ is nondecreasing, which, together with the definitions of Π_0 and Π_j in (A.1) and (A.2) and the fact that $\Gamma_0 = 0$, implies that

$$\Pi_j - \Gamma_j p(\tilde{u}_j) \ge \Pi_0 - \Gamma_0 p(\tilde{u}_0) = \min_{u\in\Omega} \frac{1}{2}\|u - u_0\|_{\mathcal{X}}^2 \ge 0.$$

Inequality (2.13) then follows from the facts that the function in the minimization problem (A.2) is strongly convex with modulus $\Gamma_j\mu + 1$ and that w_j is its solution. Moreover, the bounds in (2.12) can be obtained from relation (2.8) by induction. ∎

REFERENCES

[1] A. Auslender and M. Teboulle, *Asymptotic Cones and Functions in Optimization and Variational Inequalities*, Springer Monogr. Math., Springer-Verlag, New York, 2003.

[2] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.

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

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

[5] Y. Chen, G. Lan, and Y. Ouyang, *Optimal primal-dual methods for a class of saddle point problems*, arXiv preprint, arXiv:1309.5548, 2013.

[6] P. L. Combettes and J.-C. Pesquet, *Primal-dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators*, Set-Valued Var. Anal., 20 (2012), pp. 307–330.

[7] L. Condat, *A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms*, J. Optim. Theory Appl., 158 (2013), pp. 460–479.

[8] Y. He and R. D. C. Monteiro, *Accelerating Block-Decomposition First-Order Methods for Solving Generalized Saddle-Point and Nash Equilibrium Problems*, Optimization-Online preprint, http://www.optimization-online.org/DB_HTML/2013/10/4101.html, 2013.

[9] G. M. Korpelevič, *An extragradient method for finding saddle points and for other problems*, Èkonom. i Mat. Metody, 12 (1976), pp. 747–756.

[10] R. D. C. Monteiro and B. F. Svaiter, *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 J. Optim., 21 (2011), pp. 1688–1720.
[11] R. D. C. Monteiro and B. F. Svaiter, *On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean*, SIAM J. Optim., 20 (2010), pp. 2755–2787.
[12] R. D. C. Monteiro and B. F. Svaiter, *Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers*, SIAM J. Optim., 23 (2013), pp. 475–507.
[13] 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 J. Optim., 15 (2004), pp. 229–251.
[14] Y. Nesterov, *Smooth minimization of non-smooth functions*, Math. Program., 103 (2005), pp. 127–152.
[15] R. T. Rockafellar, *On the maximal monotonicity of subdifferential mappings*, Pacific J. Math., 33 (1970), pp. 209–216.
[16] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.
[17] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Grundlehren Math. Wiss. 317, Springer, Berlin, 1998.
[18] M. V. Solodov and B. F. Svaiter, *A hybrid approximate extragradient-proximal point algorithm using the enlargement of a maximal monotone operator*, Set-Valued Anal., 7 (1999), pp. 323–345.
[19] M. V. Solodov and B. F. Svaiter, *A hybrid projection-proximal point algorithm*, J. Convex Anal., 6 (1999), pp. 59–70.
[20] M. V. Solodov and B. F. Svaiter, *An inexact hybrid generalized proximal point algorithm and some new results on the theory of Bregman functions*, Math. Oper. Res., 25 (2000), pp. 214–230.
[21] M. V. Solodov and B. F. Svaiter, *A unified framework for some inexact proximal point algorithms*, Numer. Funct. Anal. Optim., 22 (2001), pp. 1013–1035.
[22] P. Tseng, *A modified forward-backward splitting method for maximal monotone mappings*, SIAM J. Control Optim., 38 (2000), pp. 431–446.
[23] P. Tseng, *On Accelerated Proximal Gradient Methods for Convex-Concave Optimization*, http://www.mit.edu/~dimitrib/PTseng/papers.html, 2008.
[24] B. Công Vũ, *A splitting algorithm for dual monotone inclusions involving cocoercive operators*, Adv. Comput. Math., 38 (2013), pp. 667–681.