Optimization Methods and Software

ISSN: 1055-6788 (Print) 1029-4937 (Online) Journal homepage: http://www.tandfonline.com/loi/goms20

An accelerated non-Euclidean hybrid proximal extragradient-type algorithm for convex–concave saddle-point problems

O. Kolossoski & R.D.C. Monteiro

To cite this article: O. Kolossoski & R.D.C. Monteiro (2017) An accelerated non-Euclidean hybrid proximal extragradient-type algorithm for convex–concave saddle-point problems, Optimization Methods and Software, 32:6, 1244-1272, DOI: 10.1080/10556788.2016.1266355

To link to this article: http://dx.doi.org/10.1080/10556788.2016.1266355

Optimization Methods & Software, 2017
Vol. 32, No. 6, 1244–1272, https://doi.org/10.1080/10556788.2016.1266355

An accelerated non-Euclidean hybrid proximal extragradient-type algorithm for convex–concave saddle-point problems

O. Kolossoski[a]* and R.D.C. Monteiro[b]

[a]*Departamento de Matemática, Universidade Federal do Paraná, Curitiba, PR 81531-990, Brazil;*
[b]*School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205, USA*

(*Received 18 September 2015; accepted 24 November 2016*)

This paper describes an accelerated HPE-type method based on general Bregman distances for solving convex–concave saddle-point (SP) problems. The algorithm is a special instance of a non-Euclidean hybrid proximal extragradient framework introduced by Svaiter and Solodov [*An inexact hybrid generalized proximal point algorithm and some new results on the theory of Bregman functions*, Math. Oper. Res. 25(2) (2000), pp. 214–230] where the prox sub-inclusions are solved using an accelerated gradient method. It generalizes the accelerated HPE algorithm presented in He and Monteiro [*An accelerated HPE-type algorithm for a class of composite convex–concave saddle-point problems*, SIAM J. Optim. 26 (2016), pp. 29–56] in two ways, namely: (a) it deals with general monotone SP problems instead of bilinear structured SPs and (b) it is based on general Bregman distances instead of the Euclidean one. Similar to the algorithm of He and Monteiro [*An accelerated HPE-type algorithm for a class of composite convex–concave saddle-point problems*, SIAM J. Optim. 26 (2016), pp. 29–56], it has the advantage that it works for any constant choice of proximal stepsize. Moreover, a suitable choice of the stepsize yields a method with the best known iteration-complexity for solving monotone SP problems. Computational results show that the new method is superior to Nesterov's [*Smooth minimization of non-smooth functions*, Math. Program. 103(1) (2005), pp. 127–152] smoothing scheme.

Keywords: convex programming; complexity; ergodic convergence; maximal monotone operator; hybrid proximal extragradient method; accelerated gradient method; inexact proximal method; saddle-point problem; Bregman distances

2010 Mathematics Subject Classification: 90C25; 90C30; 47H05

1. Introduction

Given nonempty closed convex sets $X \subset \mathcal{X}$ and $Y \subset \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$ are two inner product spaces, and a convex–concave map $\hat{\Phi} : X \times Y \to \mathbb{R}$, our goal in this paper is to develop algorithms for finding (approximate) saddle-points of $\hat{\Phi}$, i.e. pairs $(\bar{x}, \bar{y}) \in X \times Y$ such that

$$\hat{\Phi}(\bar{x}, y) \leq \hat{\Phi}(\bar{x}, \bar{y}) \leq \hat{\Phi}(x, \bar{y}), \quad \forall (x, y) \in X \times Y, \tag{1}$$

*Corresponding author. Email: oliver.kolossoski@ufpr.br

or equivalently, a zero of the operator $T : \mathcal{X} \times \mathcal{Y} \rightrightarrows \mathcal{X} \times \mathcal{Y}$ defined as

$$T(x,y) = \begin{cases} \partial[\hat{\Phi}(\cdot,y) - \hat{\Phi}(x,\cdot)](x,y) & \text{if } (x,y) \in X \times Y, \\ \emptyset & \text{otherwise.} \end{cases} \tag{2}$$

Under mild assumptions on Φ (see Proposition 2.4), the operator T is maximal monotone, and hence an approximate zero of T can be computed by using an inexact proximal-type algorithm such as one of the algorithms presented in [13,16, 18–21,27–31].

In particular, He and Monteiro [13] presented an inexact proximal-point method for solving the special case of the saddle-point problem in which $\hat{\Phi}$ is of the form

$$\hat{\Phi}(x,y) = f(x) + \langle Ax, y\rangle + g_1(x) - g_2(y), \tag{3}$$

where $A : \mathcal{X} \to \mathcal{Y}$ is a linear operator, g_1 and g_2 are proper closed convex functions, $X \times Y = \text{dom } g_1 \times \text{dom } g_2$, and f is a differentiable convex function with Lipschitz continuous gradient on X. The method is a special instance of the hybrid proximal extragradient (HPE) framework introduced in [27]. Any instance of the HPE framework is essentially an inexact proximal point method which allows for a relative error in the prox sub-inclusions. More specifically, given a maximal monotone operator $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$, where $\mathcal{Z}$ is a finite dimensional inner product space, consider the problem of finding z such that

$$0 \in T(z). \tag{4}$$

Recall that for a given $z_- \in \mathcal{Z}$, the exact proximal point method determines a stepsize $\lambda > 0$ and then computes the next iterate z as $z = (\lambda T + I)^{-1}(z_-)$, or equivalently, as the (unique) solution of

$$\frac{z_- - z}{\lambda} \in T(z). \tag{5}$$

An instance of the HPE framework on the other hand allows an approximate solution of (4) satisfying the (relative) HPE error criterion, namely, for some tolerance $\sigma \in [0,1]$, a stepsize $\lambda > 0$ and a triple $(\tilde{z}, z, \varepsilon)$ are computed in such a way as to satisfy

$$\frac{z_- - z}{\lambda} \in T^{\varepsilon}(\tilde{z}), \quad \frac{1}{2}\|\tilde{z} - z\|^2 + \lambda\varepsilon \le \frac{1}{2}\sigma\|\tilde{z} - z_-\|^2, \tag{6}$$

where T^{ε} denotes the ε-enlargement [2] of T. (It has the property that $T^{\varepsilon}(u) \supset T(u)$ for each u with equality holding when $\varepsilon = 0$.) Clearly, when $\sigma = 0$ in (6), then $z = \tilde{z}$ and $\varepsilon = 0$, and the inclusion in (6) reduces to (5). As opposed to other HPE-type methods in the literature (see for instance [12,20]) which have to choose λ relatively small, the HPE method of [13] for solving (4) with T as in (2) can choose an arbitrarily sized $\lambda > 0$ and computes the triple $(\tilde{z}, z, \varepsilon)$ satisfying the HPE error condition (6) with the aid of an accelerated gradient method (e.g. see [22,32]) applied to a certain regularized convex–concave min–max problem related to $\hat{\Phi}$ in (3).

The main goal of this paper is to develop a non-Euclidean HPE (NE-HPE) method which extends the one of [13] in two relevant ways. First, it solves saddle-point problems with general convex–concave functions $\hat{\Phi}$ such that $\nabla_x\hat{\Phi}$ is Lipschitz continuous instead of those with $\hat{\Phi}$ given in (3). Second, the method is a special instance of a more general non-Euclidean HPE framework which is based on a general Bregman distance instead of the specific Euclidean one. More specifically, let $\text{d}w_z(z') = w(z') - w(z) - \langle \nabla w(z), z' - z\rangle$ for every z, z', where w is a differentiable convex function. Then, the Euclidean distance is obtained by setting $w(\cdot) = \|\cdot\|^2/2$

in which case $\mathrm{d}w_z(z') = \|z' - z\|^2/2$ for all z', z and (6) can be written as follows:

$$\frac{1}{\lambda}\nabla(\mathrm{d}w)_z(z_-) \in T^\epsilon(\tilde{z}), \quad (\mathrm{d}w)_z(\tilde{z}) + \lambda\varepsilon \le \sigma(\mathrm{d}w)_{z_-}(\tilde{z}). \tag{7}$$

The non-Euclidean HPE framework generalizes the HPE one in that it allows an approximate solution of (4) satisfying the more general NE-HPE error condition (7) where w is an arbitrary convex function. As an important step towards analysing the new NE-HPE method, we establish the ergodic iteration-complexity of the NE-HPE framework for solving inclusion (4) where T is a maximal monotone operator. Similar to the method in [13], the new NE-HPE method chooses an arbitrary $\lambda > 0$ and computes a triple $(\tilde{z}, z, \varepsilon)$ satisfying the HPE error condition (7) with the aid of an accelerated gradient method applied to a certain regularized convex–concave min–max problem. Under the assumption that the feasible set of the saddle-point problem is bounded, an ergodic iteration-complexity bound is developed for the total number of inner (accelerated gradient) iterations performed by the new NE-HPE method. Finally, it is shown that if the stepsize λ and Bregman distance are properly chosen, then the derived ergodic iteration-complexity reduces to the one obtained in [22] for Nesterov's smoothing scheme which finds approximate solutions of a bilinear structured convex–concave saddle-point problem. Such complexity bound is known to be optimal (see, for example, the discussion in paragraph (1) of Section 1.1 of [7]).

Our paper is organized as follows. Section 2 contains two subsections which provide the necessary background material for our presentation. Section 2.1 introduces some notation, presents basic definitions and properties of operators and convex functions, and discusses the saddle-point problem and some of its basic properties. Section 2.2 reviews an accelerated gradient method for solving composite convex optimization problems. Section 3 reviews the notion of distance generating functions, then presents the NE-HPE framework for solving (4) and establishes its ergodic iteration-complexity. Section 4 describes the new accelerated NE-HPE method for solving the saddle-point problem, i.e. inclusion (4) with the operator given in (2). It contains three subsections as follows. Section 4.1 presents a scheme based on the accelerated gradient method of Section 2.2 for finding an approximate solution of the prox sub-inclusion according to the NE-HPE error criterion (7) and states its iteration-complexity result. Section 4.2 completely describes the new accelerated NE-HPE method for solving the saddle-point problem and establishes its overall ergodic inner iteration-complexity. It also discusses a way of choosing the prox stepsize λ so that the overall ergodic inner iteration-complexity bound reduces to the one obtained for Nesterov's smoothing scheme [22]. Section 4.3 provides the proof of the iteration-complexity result stated in Section 4.1. Finally, numerical results are presented in Section 5 showing that the new method outperforms the scheme of [22] on three classes of convex–concave saddle-point problems and that it can handle problems which are not of the form (3).

1.1 *Previous related works*

In the context of variational inequalities, Nemirovski [21] established the ergodic iteration-complexity of an extension of Korpelevich's method [16], namely, the mirror-prox algorithm, under the assumption that the feasible set of the problem is bounded. More recently, Dang and Lan [8] established the iteration-complexity of a class of non-Euclidean extragradient methods for solving variational inequalities when the operators are not necessarily monotone. Also, Lan et al. [6] established the iteration-complexity of an accelerated mirror-prox method which finds weak solutions of a class of variational inequalities. They obtained optimal complexity for the case where the feasible set of the problem is bounded.

Nesterov [22] developed a smoothing scheme for solving bilinear structured saddle-point problems under the assumption that X and Y are compact convex sets. It consists of first

approximating the objective function of the associated convex–concave saddle-point problem by a convex differentiable function with Lipschitz continuous gradient and then applying an accelerated gradient-type method (see e.g. [22, 32]) to the resulting approximation problem.

The HPE framework and its convergence results are studied in [27] and its iteration-complexity is established in [18] (see also [19,20]). The complexity results in [18] 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., saddle-point problems) are discussed in [18] and in the subsequent papers [19,20]. More specifically, by viewing Korpelevich's method [16] as well as Tseng's modified forward–backward splitting (MF-BS) method [31] as special cases of the HPE framework, the authors have established in [18,19] the pointwise and ergodic iteration complexities of these methods applied to either: 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 saddle-point problems.

Solodov and Svaiter [29] has studied a more specialized version of the NE-HPE framework which allows approximate solutions of (4) according to (7) but with $\varepsilon = 0$. Finally, extensions of the proximal method to the context of Bregman distances have been studied in [4,5,10,11,14,15]. However, none of the works cited in this paragraph deal with iteration-complexity results.

2. Background material

This section provides background material necessary for the paper presentation. Section 2.1 presents the notation and basic definitions that will be used in the paper. Section 2.2 reviews a variant of Nesterov's accelerated method for the composite convex optimization problem.

2.1 *Basic notation, definitions and results*

This subsection establishes notation and gives basic results that will be used throughout the paper.

The set of real numbers is denoted by $\mathbb{R}$. The set of non-negative real numbers and the set of positive real numbers are denoted, respectively, as $\mathbb{R}_+$ and $\mathbb{R}_{++}$. Let $\lceil z \rceil$ denote the smallest integer not less than $z \in \mathbb{R}$.

2.1.1 *Convex functions, monotone operators and their enlargements.*

Let $\mathcal{Z}$ denote a finite dimensional inner product space with inner product denoted by $\langle \cdot, \cdot \rangle$. For a set $Z \subset \mathcal{Z}$, its relative interior is denoted by $\mathrm{ri}(Z)$ and its closure as $\mathrm{cl}(Z)$. A relation $T \subseteq \mathcal{Z} \times \mathcal{Z}$ can be identified with an operator $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ in which

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

Note that the relation T is then the same as the graph of the operator T defined as

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

The domain of T is defined as

$$\mathrm{Dom}\, T := \{z \in \mathcal{Z} : T(z) \neq \emptyset\}.$$

The domain of definition of a point-to-point map F is also denoted by Dom F. An operator $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is *monotone* if

$$\langle v - \tilde{v}, z - \tilde{z} \rangle \geq 0, \quad \forall (z, v), (\tilde{z}, \tilde{v}) \in \mathrm{Gr}(T).$$

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. $S : \mathcal{Z} \rightrightarrows \mathcal{Z}$ monotone and $\mathrm{Gr}(S) \supset Gr(T)$ implies that $S = T$.

Given a scalar ε, the ε-enlargement of an operator $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is the point-to-set operator $T^\varepsilon : \mathcal{Z} \rightrightarrows \mathcal{Z}$ defined as

$$T^\varepsilon(z) := \{v \in \mathcal{Z} : \langle z - \tilde{z}, v - \tilde{v} \rangle \geq -\varepsilon, \ \forall \tilde{z} \in \mathcal{Z}, \ \forall \tilde{v} \in T(\tilde{z})\}, \quad \forall z \in \mathcal{Z}. \tag{8}$$

The following result gives some useful properties of ε-enlargements.

Proposition 2.1 *Let $T, T' : \mathcal{Z} \rightrightarrows \mathcal{Z}$ be given. Then, the following statement holds:*

(a) $T^{\varepsilon_1}(z) + (T')^{\varepsilon_2}(z) \subset (T + T')^{\varepsilon_1 + \varepsilon_2}(z)$ *for every* $z \in \mathcal{Z}$ *and* $\varepsilon_1, \varepsilon_2 \in \mathbb{R}$.
(b) *If T is maximal monotone and $\varepsilon \geq 0$, then* Dom $T^\varepsilon \subset \mathrm{cl}(\mathrm{Dom}\ T)$.

Proof The proof of (a) follows directly from definition (8). For a proof of (b), see Corollary 2.2 of [3]. ■

Let $f : \mathcal{Z} \to [-\infty, \infty]$ be given. The effective domain of f is defined as

$$\mathrm{dom} f := \{z \in \mathcal{Z} : f(z) < \infty\}.$$

Given a scalar $\varepsilon \geq 0$, the ε-subdifferential of f is the operator $\partial_\varepsilon f : \mathcal{Z} \rightrightarrows \mathcal{Z}$ defined as

$$\partial_\varepsilon f(z) := \{v : f(\tilde{z}) \geq f(z) + \langle \tilde{z} - z, v \rangle - \varepsilon, \ \forall \tilde{z} \in \mathcal{Z}\}, \quad \forall z \in \mathcal{Z}. \tag{9}$$

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 if f is proper. If f is a proper closed convex function, then ∂f is maximal monotone [24].

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

$$\mathcal{I}_\Omega(z) := \begin{cases} 0, & z \in \Omega, \\ \infty, & z \notin \Omega. \end{cases} \tag{10}$$

The following simple but useful result shows that adding a maximal monotone operator T to the subdifferential of the indicator function of a convex set containing Dom T does not change T.

Proposition 2.2 *Assume that T is a maximal monotone operator and $\Omega \subset \mathcal{Z}$ is a convex set containing* Dom T. *Then, $T + \partial \mathcal{I}_\Omega = T$.*

Proof Clearly, $\partial \mathcal{I}_\Omega$ is monotone since, by assumption, the set Ω, and hence the indicator function $\mathcal{I}_\Omega$, is convex. Since T is also monotone by assumption, it follows that $T + \partial \mathcal{I}_\Omega$ is monotone in view of Proposition 6.1.1(b) of [1]). Clearly, $T \subset T + \partial \mathcal{I}_\Omega$ due to the assumption that Dom $T \subset \Omega$ and the fact that $0 \in \partial \mathcal{I}_\Omega(x)$ for every $x \in \Omega$. The conclusion of the proposition now follows from the above two observations and the assumption that T is maximal monotone. ■

2.1.2 *The saddle-point problem.*

Let $\mathcal{X}$ and $\mathcal{Y}$ be finite dimensional inner product spaces with inner products denoted, respectively, by $\langle\cdot,\cdot\rangle_{\mathcal{X}}$ and $\langle\cdot,\cdot\rangle_{\mathcal{Y}}$ and endow the product space $\mathcal{X}\times\mathcal{Y}$ with the canonical inner product defined as

$$\langle(x,y),(x',y')\rangle = \langle x,x'\rangle_{\mathcal{X}} + \langle y,y'\rangle_{\mathcal{Y}}, \quad \forall (x,y),(x',y') \in \mathcal{X}\times\mathcal{Y}. \tag{11}$$

Let $X \subseteq \mathcal{X}$ and $Y \subseteq \mathcal{Y}$ be nonempty sets and define

$$\mathcal{Z} := \mathcal{X}\times\mathcal{Y}, \quad Z := X \times Y.$$

For a given function $\hat{\Phi} : Z \to \mathbb{R}$, a pair $(\bar{x},\bar{y}) \in Z$ is called a saddle-point of $\hat{\Phi}$ if it satisfies (1). The problem of determining such a pair is called the saddle-point problem determined by $\hat{\Phi}$ and is denoted by $\mathrm{SP}(\hat{\Phi})$.

Define $T_{\hat{\Phi}} : \mathcal{Z} \rightrightarrows \mathcal{Z}$ as

$$T_{\hat{\Phi}}(z) := \begin{cases} \partial(\hat{\Phi}_z)(z) & \text{if } z \in Z, \\ \emptyset & \text{otherwise,} \end{cases} \tag{12}$$

where, for every $z = (x,y) \in Z$, the function $\hat{\Phi}_z : \mathcal{Z} \to (-\infty,+\infty]$ is defined as

$$\hat{\Phi}_z(z') = \begin{cases} \hat{\Phi}(x',y) - \hat{\Phi}(x,y'), & \forall z' = (x',y') \in Z, \\ +\infty & \text{otherwise.} \end{cases} \tag{13}$$

Clearly, $z = (x,y)$ is a saddle-point of $\hat{\Phi}$ if and only if z is a solution of the inclusion

$$0 \in T_{\hat{\Phi}}(z). \tag{14}$$

The operator $T_{\hat{\Phi}}$ admits the ε-enlargement as in (8). It also admits an ε-saddle-point enlargement which exploits its natural saddle-point nature, namely, $\partial_\varepsilon(\hat{\Phi}_z)(z)$ for $z \in Z$. The following result whose proof can be found for example in Lemma 3.2 of [13] follows straightforwardly from definitions (8) and (9).

PROPOSITION 2.3 *For every $z \in Z$ and $\varepsilon \geq 0$, the inclusion $\partial_\varepsilon(\hat{\Phi}_z)(z) \subset [T_{\hat{\Phi}}]^\varepsilon(z)$ holds.*

The following result (see for example Theorem 6.3.2 in [1]) gives sufficient conditions for the operator $T_{\hat{\Phi}}$ in (12) to be maximal monotone.

PROPOSITION 2.4 *The following statements hold:*

(a) *$T_{\hat{\Phi}}$ is monotone;*
(b) *if the function $\hat{\Phi}_z : \mathcal{Z} \to (-\infty,+\infty]$ is closed convex for every $z \in Z$, then Z is convex and $T_{\hat{\Phi}}$ is maximal monotone.*

Note that, due to the definition of T^ε, the verification of the inclusion $v \in T^\varepsilon(x)$ requires checking an infinite number of inequalities. This verification is feasible only for specially structured instances of operators T. However, it is possible to compute points in the graph of T^ε using the following *weak transportation formula* [3].

PROPOSITION 2.5 *Suppose that $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is maximal monotone. Moreover, for $i = 1, \ldots, k$, let $z_i, r_i \in \mathcal{Z}$ and $\varepsilon_i, \alpha_i \in \mathbb{R}_+$ satisfying $\sum_{i=1}^k \alpha_i = 1$ be given and define*

$$z^a = \sum_{i=1}^k \alpha_i z_i, \quad r^a = \sum_{i=1}^k \alpha_i r_i, \quad \varepsilon^a = \sum_{i=1}^k \alpha_i [\varepsilon_i + \langle z_i - z^a, r_i - r^a \rangle].$$

Then, the following statements hold:

(a) *if $r_i \in T^{\varepsilon_i}(z_i)$ for every $i = 1, \ldots, k$, then $\varepsilon^a \geq 0$ and $r^a \in T^{\varepsilon^a}(z^a)$;*
(b) *if $T = T_{\hat{\Phi}}$, where $\hat{\Phi}$ is a saddle function satisfying the assumptions of Proposition* 2.4(b) *and the stronger inclusion $r_i \in \partial_{\varepsilon_i}(\hat{\Phi}_{z_i})(z_i)$ holds for every $i = 1, \ldots, k$, then*

$$r^a \in \partial_{\varepsilon^a}(\hat{\Phi}_{z^a})(z^a).$$

2.2 *Accelerated method for composite convex optimization*

This subsection reviews a variant of Nesterov's accelerated first-order method [22,32] for minimizing a (possibly strongly) convex composite function. In what follows, we refer to convex functions as 0-strongly convex functions. This terminology has the benefit of allowing us to treat both the convex and strongly convex cases simultaneously.

Let $\mathcal{X}$ denote a finite dimensional inner product space with an inner product denoted by $\langle \cdot, \cdot \rangle_{\mathcal{X}}$ and a norm denoted by $\| \cdot \|_{\mathcal{X}}$ which is not necessarily the one induced by the inner product. Consider the following composite optimization problem:

$$\inf\{f(x) + g(x) : x \in X\}, \tag{15}$$

where $f : X \subset \mathcal{X} \to \mathbb{R}$ and $g : \mathcal{X} \to [-\infty, +\infty]$ satisfy the following conditions:

(A.1) g is a proper closed μ-strongly convex function;
(A.2) X is a convex set such that $X \supset \operatorname{dom} g$;
(A.3) there exist a constant $L > 0$ and a function $\nabla f : X \to X$ such that for every $x, x' \in X$,

$$f(x') + \langle \nabla f(x'), x - x' \rangle_{\mathcal{X}} \leq f(x) \leq f(x') + \langle \nabla f(x'), x - x' \rangle_{\mathcal{X}} + \frac{L}{2}\|x - x'\|_{\mathcal{X}}^2.$$

Even though the map ∇f is not necessarily the gradient of f, it plays a similar role to it and hence our notation.

The accelerated method for solving problem (15) stated below requires the specification of a point $x_0 \in \operatorname{dom} g$ and a function $h : \mathcal{X} \to (-\infty, \infty]$ satisfying

(A.4) h is a proper closed convex function such that $\operatorname{dom} h \supset \operatorname{dom} g$;
(A.5) h is 1-strongly convex on $\operatorname{dom} g$;
(A.6) $x_0 = \operatorname{argmin}\{h(x) : x \in \operatorname{dom} g\}$.

Clearly, if $\operatorname{dom} g$ is closed then the above optimization problem always has an unique global minimum which can be taken to be the point x_0. The special case of the method below with $\mu = 0$ is the same as the accelerated variant stated in Algorithm 3 of [32]. Its proof for $\mu > 0$ is not given in [32] but follows along the same line as the one for Algorithm 3 of [32] (see also Section 2.2 of [13] for the proof of the case where $\mu > 0$, X is closed and $h(\cdot) = \| \cdot - u_0 \|^2/2$ for some $u_0 \in \mathcal{X}$).

ALGORITHM 1 **A variant of Nesterov's accelerated method:**

(0) *Set* $A_0 := 0$, $\Theta_0 :\equiv 0$, $k = 1$ *and* $\tilde{x}_0 = x_0$ *where* x_0 *is as in* A.6;
(1) *compute*

$$A_k := A_{k-1} + \frac{(1+\mu A_{k-1}) + \sqrt{(1+\mu A_{k-1})^2 + 4L(1+\mu A_{k-1})A_{k-1}}}{2L}, \tag{16}$$

$$\check{x}_k := \frac{A_{k-1}}{A_k}\tilde{x}_{k-1} + \frac{A_k - A_{k-1}}{A_k}x_{k-1}, \tag{17}$$

$$\Theta_k := \frac{A_{k-1}}{A_k}\Theta_{k-1} + \frac{A_k - A_{k-1}}{A_k}[f(\check{x}_k) + \langle \nabla f(\check{x}_k), \cdot - \check{x}_k\rangle_{\mathcal{X}}]; \tag{18}$$

(2) *iterate* x_k *and* $\tilde{x}_k$ *as*

$$x_k := \operatorname{argmin}\left\{\Theta_k(x) + g(x) + \frac{1}{A_k}h(x)\right\}, \tag{19}$$

$$\tilde{x}_k := \frac{A_{k-1}}{A_k}\tilde{x}_{k-1} + \frac{A_k - A_{k-1}}{A_k}x_k; \tag{20}$$

(3) *set* $k \leftarrow k+1$ *and go to step 1.*

end

The main technical result which yields the convergence rate of the above accelerated method is as follows.

PROPOSITION 2.6 *The sequences* $\{A_k\}$, $\{\tilde{x}_k\}$ *and* $\{\Theta_k\}$ *generated by Algorithm* 1 *satisfy the following inequalities for any* $k \geq 1$:

$$A_k \geq \frac{1}{L}\max\left\{\frac{k^2}{4}, \left(1+\sqrt{\frac{\mu}{4L}}\right)^{2(k-1)}\right\}, \tag{21}$$

$$\Theta_k \leq f, \quad (f+g)(\tilde{x}_k) \leq \Theta_k(x) + g(x) + \frac{1}{A_k}[h(x) - h(x_0)], \quad \forall x \in \operatorname{dom} g. \tag{22}$$

3. The NE-HPE framework

This section describes an extension of the non-Euclidean HPE framework introduced in [27] for finding an approximate solution of the monotone inclusion problem (4), and establishes ergodic convergence rate bounds for it. It also presents a specialization of the latter result in the context of the saddle-point SP($\hat{\Phi}$).

It is assumed throughout this section that $\mathcal{Z}$ is an inner product space with inner product $\langle\cdot,\cdot\rangle$ and that $\|\cdot\|$ is a (general) norm in $\mathcal{Z}$ which is not necessarily the inner product induced norm.

Before presenting the framework, we introduce the notions of distance generating functions and Bregman distances used in our presentation.

DEFINITION 3.1 *A proper closed convex function* $w : \mathcal{Z} \to [-\infty, \infty]$ *is called a distance generating function if it satisfies the following conditions:*

(i) $W := \operatorname{dom} w$ *is closed and* $W^0 := \operatorname{int}(W) = \{z \in \mathcal{Z} : \partial w(z) \neq \emptyset\}$;

(ii) *w restricted to W is continuous and w is continuously differentiable on W^0.*

Moreover, w induces the Bregman distance $\mathrm{d}w : \mathcal{Z} \times W^0 \to \mathbb{R}$ *defined as*

$$(\mathrm{d}w)(z';z) := w(z') - w(z) - \langle \nabla w(z), z' - z\rangle, \quad \forall (z', z) \in \mathcal{Z} \times W^0. \tag{23}$$

Clearly, $W^0 \neq \emptyset$ due to Definition 3.1(i) and Theorem 23.4 of [23]. For simplicity, for every $z \in W^0$, the function $(\mathrm{d}w)(\cdot; z)$ will be denoted by $(\mathrm{d}w)_z$ so that

$$(\mathrm{d}w)_z(z') = (\mathrm{d}w)(z';z), \quad \forall z' \in \mathcal{Z}.$$

The following useful identities follow straightforwardly from (20):

$$\nabla(\mathrm{d}w)_z(z') = -\nabla(\mathrm{d}w)_{z'}(z) = \nabla w(z') - \nabla w(z), \quad \forall z, z' \in W^0, \tag{24}$$

$$(\mathrm{d}w)_v(z') - (\mathrm{d}w)_v(z) = \langle \nabla(\mathrm{d}w)_v(z), z' - z\rangle + (\mathrm{d}w)_z(z'), \quad \forall z' \in \mathcal{Z}, \ \forall v, z \in W^0. \tag{25}$$

We next describe the NE-HPE framework for solving (4) under the assumption that $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ is a maximal monotone operator. Its description is based on a distance generating function w whose domain W satisfies the following condition:

(B.1) $\mathrm{ri}(\mathrm{Dom}\, T) \subset W^0$.

Clearly, $\mathrm{cl}(\mathrm{Dom}\, T) \subset W$ due to B.1 and the fact that, by Definition 3.1, W is closed.

NE-HPE framework:

(0) Let $z_0 \in W^0$ be given and set $j = 1$;

(1) choose $\sigma_j \in [0, 1]$, and find $\lambda_j > 0$ and $(\tilde{z}_j, z_j, \varepsilon_j) \in W \times W^0 \times \mathbb{R}_+$ such that

$$r_j := \frac{1}{\lambda_j} \nabla(\mathrm{d}w)_{z_j}(z_{j-1}) \in T^{\varepsilon_j}(\tilde{z}_j), \tag{26}$$

$$(\mathrm{d}w)_{z_j}(\tilde{z}_j) + \lambda_j \varepsilon_j \leq \sigma_j (\mathrm{d}w)_{z_{j-1}}(\tilde{z}_j); \tag{27}$$

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

end

We now make several remarks about the NE-HPE framework. First, the NE-HPE framework does not specify how to find λ_j and $(\tilde{z}_j, z_j, \varepsilon_j)$ satisfying (26) and (27). The particular scheme for computing λ_j and $(\tilde{z}_j, z_j, \varepsilon_j)$ depends on the instance of the framework under consideration and the properties of the operator T. Second, if $\sigma_j = 0$ and the Bregman distance is nondegenerate in sense that $(\mathrm{d}w)_z(\tilde{z}) = 0$ implies that $z = \tilde{z}$, then (27) implies that $\varepsilon_j = 0$ and $z_j = \tilde{z}_j$, and hence that $r_j \in T(z_j)$ in view of (26). Therefore, the HPE error conditions (26) and (27) can be viewed as a relaxation of an iteration of the exact non-Euclidean proximal point method [10], namely,

$$\frac{1}{\lambda_j} \nabla(\mathrm{d}w)_{z_j}(z_{j-1}) \in T(z_j).$$

Third, if w is strongly convex on $\mathrm{ri}(\mathrm{Dom}\, T)$, then it follows from Proposition A.2 of Appendix that the above inclusion has a unique solution z_j. Hence, for any given $\lambda_j > 0$, there always exists a triple $(\tilde{z}_j, z_j, \varepsilon_j)$ of the form $(z_j, z_j, 0)$ satisfying (26) and (27) with $\sigma_j = 0$. Clearly, the computation of such (exact) triple is usually not possible, and the use of (inexact) triples $(\tilde{z}_j, z_j, \varepsilon_j)$ satisfying the HPE (relative) error conditions with $\sigma_j > 0$ provide much greater coverage and computational flexibility.

It is possible to show that some methods such as the ones in [12,13,16,18–21, 27,31] can be viewed as special instances of the NE-HPE framework. Section 4 presents another instance of the above framework in the context of the saddle-point problem (and hence with $T = T_{\hat{\Phi}}$) in which the stepsize λ_j is chosen in an interval of the form $[\tau\lambda, \lambda]$ for arbitrary constants $\lambda > 0$ and $\tau \in (0, 1)$, and the triple $(\tilde{z}_j, z_j, \varepsilon_j)$ is obtained by means of the accelerated gradient method of Section 2.2 applied to a certain optimization problem.

In the remaining part of this subsection, we focus our attention on establishing ergodic convergence rate bounds for the NE-HPE framework. We start by deriving some preliminary technical results.

LEMMA 3.2 *For every $j \geq 1$, the following statements hold:*

(a) *for every $z \in W$, we have*

$$(\mathrm{d}w)_{z_{j-1}}(z) - (\mathrm{d}w)_{z_j}(z) = (\mathrm{d}w)_{z_{j-1}}(\tilde{z}_j) - (\mathrm{d}w)_{z_j}(\tilde{z}_j) + \lambda_j\langle r_j, \tilde{z}_j - z\rangle;$$

(b) *for every $z \in W$, we have*

$$(\mathrm{d}w)_{z_{j-1}}(z) - (\mathrm{d}w)_{z_j}(z) \geq (1-\sigma_j)(\mathrm{d}w)_{z_{j-1}}(\tilde{z}_j) + \lambda_j(\langle r_j, \tilde{z}_j - z\rangle + \varepsilon_j).$$

Proof (a) Using (25) twice and using the definition of r_j given in (26), we have that

$$\begin{aligned}(\mathrm{d}w)_{z_{j-1}}(z) - (\mathrm{d}w)_{z_j}(z) &= (\mathrm{d}w)_{z_{j-1}}(z_j) + \langle \nabla(\mathrm{d}w)_{z_{j-1}}(z_j), z - z_j\rangle \\ &= (\mathrm{d}w)_{z_{j-1}}(z_j) + \langle \nabla(\mathrm{d}w)_{z_{j-1}}(z_j), \tilde{z}_j - z_j\rangle + \langle \nabla(\mathrm{d}w)_{z_{j-1}}(z_j), z - \tilde{z}_j\rangle \\ &= (\mathrm{d}w)_{z_{j-1}}(\tilde{z}_j) - (\mathrm{d}w)_{z_j}(\tilde{z}_j) + \langle \nabla(\mathrm{d}w)_{z_{j-1}}(z_j), z - \tilde{z}_j\rangle \\ &= (\mathrm{d}w)_{z_{j-1}}(\tilde{z}_j) - (\mathrm{d}w)_{z_j}(\tilde{z}_j) + \lambda_j\langle r_j, \tilde{z}_j - z\rangle.\end{aligned}$$

(b) This statement follows as an immediate consequence of (a) and (27). ■

The following result follows as an immediate consequence of Lemma 3.2(b).

LEMMA 3.3 *For every $j \geq 1$ and $z \in W$, we have that*

$$(\mathrm{d}w)_{z_0}(z) - (\mathrm{d}w)_{z_j}(z) \geq \sum_{i=1}^{j}(1-\sigma_i)(\mathrm{d}w)_{z_{i-1}}(\tilde{z}_i) + \sum_{i=1}^{j}\lambda_i[\varepsilon_i + \langle r_i, \tilde{z}_i - z\rangle].$$

Proof The lemma follows by adding the inequality in Lemma 3.2(b) from 1 to j. ■

LEMMA 3.4 *For every $j \geq 1$, define $\Lambda_j := \sum_{i=1}^{j}\lambda_i$,*

$$\tilde{z}_j^a := \frac{1}{\Lambda_j}\sum_{i=1}^{j}\lambda_i\tilde{z}_i, \quad r_j^a := \frac{1}{\Lambda_j}\sum_{i=1}^{j}\lambda_i r_i, \quad \varepsilon_j^a := \frac{1}{\Lambda_j}\sum_{i=1}^{j}\lambda_i[\varepsilon_i + \langle r_i, \tilde{z}_i - \tilde{z}_j^a\rangle].$$

where r_i is defined in (26)*. Then, we have*

$$\varepsilon_j^a \geq 0, \quad r_j^a \in T^{\varepsilon_j^a}(\tilde{z}_j^a), \tag{28}$$

$$\varepsilon_j^a + \langle r_j^a, \tilde{z}_j^a - z\rangle \leq \frac{(\mathrm{d}w)_{z_0}(z)}{\Lambda_j}, \quad \forall z \in W. \tag{29}$$

Proof The relations on (28) follow from (26) and Proposition 2.5(a). Moreover, Lemma 3.3, the assumption that $\sigma_j \in [0,1]$, and the definitions of ε_j^a and r_j^a, imply that for every $z \in W$,

$$(\mathrm{d}w)_{z_0}(z) - (\mathrm{d}w)_{z_j}(z) \geq \sum_{i=1}^{j} \lambda_i[\varepsilon_i + \langle r_i, \tilde{z}_i - z\rangle]$$

$$= \sum_{i=1}^{j} \lambda_i[\varepsilon_i + \langle r_i, \tilde{z}_i - \tilde{z}_j^a\rangle + \langle r_i, \tilde{z}_j^a - z\rangle] = \Lambda_j[\varepsilon_j^a + \langle r_j^a, \tilde{z}_j^a - z\rangle],$$

and hence that (29) holds. ■

For any nonempty compact convex set $\Omega \subset W$, define

$$R(z_0;\Omega) := \max\{(\mathrm{d}w)_{z_0}(z) : z \in \Omega\}. \tag{30}$$

Clearly, $R(z_0;\Omega)$ is finite due to the fact that $(\mathrm{d}w)_{z_0}(\cdot)$ is a continuous function on W (see Definition 3.1(ii)).

We are now ready to state the main result of this subsection which establishes an ergodic convergence rate bound for the NE-HPE framework.

THEOREM 3.5 *For every $j \geq 1$, define Λ_j, $\tilde{z}_j^a$, r_j^a and ε_j^a as in Lemma* 3.4, *and also*

$$\tilde{\varepsilon}_j := \varepsilon_j^a + \max\{\langle r_j^a, \tilde{z}_j^a - z\rangle : z \in \Omega\}, \tag{31}$$

where $\Omega \subset W$ is a nonempty compact convex set. Then, for every $j \geq 1$, we have

$$\tilde{\varepsilon}_j \leq \frac{R}{\Lambda_j}, \tag{32}$$

$$\tilde{z}_j^a \in \Omega \Longrightarrow 0 \in (T_\Omega)^{\tilde{\varepsilon}_j}(\tilde{z}_j^a), \tag{33}$$

where $T_\Omega := T + \partial\mathcal{I}_\Omega$ and $R := R(z_0;\Omega)$. Moreover, if Dom T *is bounded and $\Omega = \mathrm{cl}(\mathrm{Dom}\, T)$, then $0 \in T^{\tilde{\varepsilon}_j}(\tilde{z}_j^a)$ for every $j \geq 1$.*

Proof Inequality (29), the definition of $R = R(z_0;\Omega)$ in (30) and the definition of $\tilde{\varepsilon}_j$ in (31) clearly imply (32). Now, assume that $\tilde{z}_j^a \in \Omega$ and let $\delta_j := \tilde{\varepsilon}_j - \varepsilon_j^a$. Then, (31), the inclusion $\tilde{z}_j^a \in \Omega$, and the definitions of the ε-subdiffential and the indicator function in (9) and (10), respectively, imply that $-r_j^a \in \partial_{\delta_j}(\mathcal{I}_\Omega)(\tilde{z}_j^a)$. This inclusion, the inclusion in (28), and Proposition 2.1(a), then imply that

$$0 \in T^{\varepsilon_j^a}(\tilde{z}_j^a) + (\partial\mathcal{I}_\Omega)^{\delta_j}(\tilde{z}_j^a) \subset (T + \partial\mathcal{I}_\Omega)^{\varepsilon_j^a+\delta_j}(\tilde{z}_j^a) = (T_\Omega)^{\tilde{\varepsilon}_j}(\tilde{z}_j^a),$$

where the last equality is due to the definitions of δ_j and T_Ω.

To prove the last assertion of the theorem, assume that $\Omega = \mathrm{cl}(\mathrm{Dom}\, T)$ and note that Ω is convex due to Theorem 1 of [25]. Also, in view of Proposition 2.1(b) and the fact that $\tilde{z}_j \in \mathrm{Dom}\, T^{\varepsilon_j}$ (see (26)), we conclude that $\tilde{z}_j \in \mathrm{cl}(\mathrm{Dom}\, T) = \Omega$ for every $j \geq 1$. Hence, the convexity of Ω and the definition of $\tilde{z}_j^a$ in Lemma 3.4 imply that $\tilde{z}_j^a \in \Omega$, and hence that $0 \in (T_\Omega)^{\tilde{\varepsilon}_j}(\tilde{z}_j^a)$ due to (33). The assertion now follows from Proposition 2.2. ■

We now make a few remarks about Theorem 3.5. First, $\tilde{\varepsilon}_j$ in (31) can be easily computed for those instances of (31) for which the minimization of a linear function on Ω can be trivially performed. Second, if Λ_j grows to ∞, relation (32) implies that any limit point of $\tilde{z}_j^a$ is a solution

of (4). Third, relation (32) implies that the convergence rate of $\tilde{z}_j^a$, measured in terms of the size of $\tilde{\varepsilon}_j$, is on the order of $\mathcal{O}(1/\Lambda_j)$. Clearly, this convergence rate reduces to $\mathcal{O}(1/j)$ for the case in which the sequence of stepsizes $\{\lambda_j\}$ is constant.

To state a variation of Theorem 3.5 in the context of the saddle-point problem, consider the following notion of an approximate solution of the saddle-point inclusion (14).

DEFINITION 3.6 *A pair $(z, \varepsilon) \in Z \times \mathbb{R}_+$ is called a near-saddle-point of $\hat{\Phi}$ if $0 \in \partial_\varepsilon(\hat{\Phi}_z)(z)$. Moreover, for a given $\bar{\varepsilon} \geq 0$, (z, ε) is called an $\bar{\varepsilon}$-saddle-point of $\hat{\Phi}$ if $0 \in \partial_\varepsilon(\hat{\Phi}_z)(z)$ and $\varepsilon \leq \bar{\varepsilon}$.*

Clearly, z is a saddle-point of $\hat{\Phi}$ if and only if $(z, 0)$ is a near-saddle-point of $\hat{\Phi}$.

COROLLARY 3.7 *Assume now that $T = T_{\hat{\Phi}}$, where $\hat{\Phi}$ is a saddle function satisfying the assumptions of Proposition* 2.4(b) *and consider an instance of the NE-HPE framework in which every $(\tilde{z}_j, z_j, \varepsilon_j)$ satisfies the stronger inclusion*

$$r_j \in \partial_{\varepsilon_j}(\hat{\Phi}_{\tilde{z}_j})(\tilde{z}_j), \quad \forall j \geq 1, \tag{34}$$

where r_j is defined in (26). *Also, for every $j \geq 1$, define Λ_j, $\tilde{z}_j^a$, r_j^a and ε_j^a as in Lemma* 3.4 *and $\tilde{\varepsilon}_j$ as in* (31) *where $\Omega := \Omega_X \times \Omega_Y \subset W$ is a nonempty convex compact set. Then, the following statements hold:*

(a) *every pair $(\tilde{z}_j^a, \tilde{\varepsilon}_j)$ such that $\tilde{z}_j^a \in \Omega$ is a near-saddle-point of the map $\hat{\Phi}$ restricted to $\Omega \cap (X \times Y)$ and $\tilde{\varepsilon}_j$ is bounded according to* (32) *where $R := R(z_0; \Omega)$;*

(b) *if $X \times Y$ is bounded, $\Omega = \mathrm{cl}(X \times Y)$ and $\{\lambda_j\} \subset [\bar{\lambda}, \infty)$ for some $\bar{\lambda} > 0$, then for any given $\bar{\varepsilon} \geq 0$, there exists $j_0 = \mathcal{O}(\lceil R/(\bar{\lambda}\bar{\varepsilon})\rceil)$ such that $(\tilde{z}_j^a, \tilde{\varepsilon}_j)$ is an $\bar{\varepsilon}$-saddle-point of $\hat{\Phi}$ for every $j \geq j_0$.*

Proof (a) The justification of the bound on $\tilde{\varepsilon}_j$ is similar to that given in the proof of Theorem 3.5. By (34) and Proposition 2.5(b), we have that $r_j^a \in \partial_{\varepsilon_j^a}(\Phi_{\tilde{z}_j^a})(\tilde{z}_j^a)$. Now using this inclusion, the assumption that $\tilde{z}_j^a \in \Omega$, and similar arguments as in the proof of Theorem 3.5, we conclude that

$$0 \in (\partial_{\varepsilon_j^a}\hat{\Phi}_{\tilde{z}_j^a} + \partial_{\delta_j}\mathcal{I}_\Omega)(\tilde{z}_j^a) \subset \partial_{\tilde{\varepsilon}_j}(\hat{\Phi}_{\tilde{z}_j^a} + \mathcal{I}_\Omega)(\tilde{z}_j^a),$$

and hence that $(\tilde{z}_j^a, \tilde{\varepsilon}_j)$ is a near-saddle-point of the map $\hat{\Phi}$ restricted to $\Omega \cap (X \times Y)$.

(b) Note that the assumption that $\Omega = \mathrm{cl}(X \times Y)$ easily implies that $\hat{\Phi}_{\tilde{z}_j^a} + \mathcal{I}_\Omega = \hat{\Phi}_{\tilde{z}_j^a}$ and $\tilde{z}_j^a \in \Omega$ for every $j \geq 1$. Thus, this statement follows directly from statement (a), bound (32) and the assumption that $\{\lambda_j\} \subset [\bar{\lambda}, \infty)$. ■

We end this section by noting that the validity of the conclusion of Corollary 3.7(a) requires the condition $\tilde{z}_j^a \in \Omega$ which is generally not guaranteed to hold. Statement (b) guarantees the latter condition by taking $\Omega = \mathrm{cl}(X \times Y)$ but this choice requires us to assume that $X \times Y$ is bounded. Hence, the most polished form of Corollary 3.7 (in the sense that the conclusion of (b) holds) can only be guaranteed when the feasible region of the saddle-point $\mathrm{SP}(\hat{\Phi})$ is bounded.

4. An accelerated instance of the NE-HPE framework

This section presents and establishes the (inner) iteration-complexity of a particular instance of the NE-HPE framework for solving the saddle-point problem where the triple $(\tilde{z}_j, z_j, \varepsilon_j)$ in step 1 of the framework is computed with the aid of the accelerated gradient method of Section 2.2.

Throughout this section, we assume that $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$, X, Y , Z, $\langle\cdot,\cdot\rangle_{\mathcal{X}}$, $\langle\cdot,\cdot\rangle_{\mathcal{Y}}$, $\langle\cdot,\cdot\rangle$ and $\hat{\Phi}$ are as in Section 2.1.2. Moreover, let $\|\cdot\|_{\mathcal{X}}$ and $\|\cdot\|_{\mathcal{Y}}$ be norms in $\mathcal{X}$ and $\mathcal{Y}$, respectively, which are not necessarily the ones induced by their corresponding inner products. Our problem of interest is the saddle-point problem $\mathrm{SP}(\hat{\Phi})$ endowed with a certain composite structure on the space $\mathcal{X}$ which consists of the existence of a proper closed convex function $\phi : \mathcal{X} \to (-\infty, +\infty]$ and a function $\Phi : \mathrm{dom}\,\Phi \supset Z \to \mathbb{R}$ satisfying

$$\mathrm{dom}\,\phi = X, \tag{35}$$

$$\hat{\Phi}(x,y) = \Phi(x,y) + \phi(x), \quad \forall (x,y) \in Z, \tag{36}$$

and the following additional conditions:

(C.1) Z is a nonempty bounded convex set;
(C.2) for every $z \in Z$, the function $\hat{\Phi}_z$ given in (13) is closed and convex;
(C.3) for every $y \in Y$, the function $\Phi(\cdot, y)$ is differentiable on X and there exist nonnegative constants L_{xx} and L_{xy} such that

$$\|\nabla_x \Phi(x',y') - \nabla_x \Phi(x,y)\|_{\mathcal{X}}^* \le L_{xx}\|x - x'\|_{\mathcal{X}} + L_{xy}\|y - y'\|_{\mathcal{Y}}, \quad \forall (x,y),(x',y') \in X \times Y,$$

where $\|\cdot\|_{\mathcal{X}}^*$ denotes the dual norm of $\|\cdot\|_{\mathcal{X}}$ defined as

$$\|x\|_{\mathcal{X}}^* := \max_{\|x'\|_{\mathcal{X}}=1} \{\langle x, x'\rangle_{\mathcal{X}} : x' \in \mathcal{X}\}, \quad \forall x \in \mathcal{X}.$$

We now make two remarks about the above conditions. First, condition C.2 and Proposition 2.4(b) imply that the operator $T_{\hat{\Phi}}$ given by (12) is maximal monotone. Second, problem (3) with the condition that ∇f is Lipschitz continuous on X, i.e. $\|\nabla f(x) - \nabla f(x')\|_{\mathcal{X}}^* \le L\|x - x'\|_{\mathcal{X}}$ for every $x, x' \in X$, is a special case of the saddle-point problem considered in this section in which $X \times Y = \mathrm{dom}\, g_1 \times \mathrm{dom}\, g_2$, $\phi = g_1$, $\Phi(x,y) = f(x) + \langle y, Ax\rangle - g_2(y)$ for every $(x,y) \in \mathrm{dom}\, f \times \mathrm{dom}\, g_2$, $L_{xx} = L$ and $L_{xy} = \max\{\|Ax\|_{\mathcal{Y}}^* : \|x\|_{\mathcal{X}} \le 1\}$.

Our goal in this section is to develop an accelerated instance of the NE-HPE framework for (approximately) solving (in the sense of Definition 3.6) the saddle-point problem $\mathrm{SP}(\hat{\Phi})$, or equivalently, inclusion (14), under the above assumptions.

We start by describing the structure of the distance generating function used by our instance. Let $w_1 : \mathcal{X} \to [-\infty, \infty]$ and $w_2 : \mathcal{Y} \to [-\infty, \infty]$ be distance generating functions with domain $W_1 \subset \mathcal{X}$ and $W_2 \subset \mathcal{Y}$, respectively. Letting $W = W_1 \times W_2$ and $W^0 = \mathrm{int}(W)$, the function $w : \mathcal{Z} \to [-\infty, \infty]$ defined as

$$w(z) := w_1(x) + w_2(y), \quad \forall z = (x,y) \in \mathcal{Z}, \tag{37}$$

is a distance generating function whose domain is W and which induces the Bregman distance

$$(\mathrm{d}w)_z(z') := (\mathrm{d}w_1)_x(x') + (\mathrm{d}w_2)_y(y'), \quad \forall z = (x,y) \in W^0, \ \forall z' = (x',y') \in \mathcal{Z}. \tag{38}$$

We further assume that the following conditions hold throughout this section:

(C.4) $\mathrm{ri}(Z) \subset W^0$;
(C.5) w_1 (resp., w_2) is η_1-strongly (resp., η_2-strongly) convex on X (resp., Y) for some $\eta_1 > 0$ (resp., $\eta_2 > 0$).

A few remarks are in order. First, C.4 and the closeness of Z guarantee that $X \times Y = Z \subset W$ so that $X \times Y \subset \mathrm{dom}\, w_1 \times \mathrm{dom}\, w_2$. Second, C.5 requires that w restricted to $X \times Y$ is strongly

convex. Third, C.4 ensures that the operator $T = T_{\hat{\Phi}}$ given in (12) satisfies condition B.1, and hence that the results of Section 3 carry over to the present context.

To describe our instance, it suffices to explain how step 1 of the NE-HPE framework is implemented. This will be the subject of Section 4.1 below which describes a scheme for implementing this step based on the acceleration gradient method of Section 2.2. For now, we just mention that the stepsize λ_j is not chosen to be constant but rather is computed within an interval of the form $[\tau\lambda, \lambda]$, where $\lambda > 0$ and $\tau \in (0,1)$ are fixed throughout our instance. In addition, the scheme of Section 4.1 also describes how to compute a triple $(\tilde{z}_j, z_j, \varepsilon_j)$ satisfying condition (27) with dw given in (38), and the stronger inclusion (34).

More specifically, Section 4.1 describes a scheme for solving the following problem.

(P1) *Given a pair* $z_- = (x_-, y_-) \in W^0$, *and scalars* $\sigma \in (0,1]$, $\lambda > 0$ *and* $\tau \in (0,1)$, *the problem is to find* $\tilde{\lambda} \in [\tau\lambda, \lambda]$ *and a triple* $(\tilde{z}, z, \varepsilon) \in W \times W^0 \times \mathbb{R}_+$ *such that*

$$r := \frac{1}{\tilde{\lambda}} \nabla(\mathrm{d}w)_z(z_-) \in \partial_\varepsilon(\hat{\Phi}_{\tilde{z}})(\tilde{z}), \tag{39}$$

$$(\mathrm{d}w)_z(\tilde{z}) + \tilde{\lambda}\varepsilon \le \sigma(\mathrm{d}w)_{z_-}(\tilde{z}). \tag{40}$$

with $\hat{\Phi}_{\tilde{z}}$ *given in (13).*

Note that problem (P1) is based on condition (34) instead of (26) with $T = T_{\hat{\Phi}}$. Recall that the first condition implies the latter one in view of Proposition 2.3.

In addition to Section 4.1, this section contains two other subsections. Section 4.2 completely describes the accelerated instance of the NE-HPE framework for solving SP($\hat{\Phi}$) and its corresponding iteration-complexity result. It also discusses optimal ways of choosing the prox stepsize in order to minimize the overall inner iteration-complexity of the instance. Finally, Section 4.3 gives the proof of the inner iteration-complexity result stated in Section 4.1.

4.1 *An accelerated scheme for solving (P1)*

This subsection presents a scheme for finding a solution of problem (P1) based on the accelerated gradient method of Section 2.2 applied to a certain regularized convex–concave min–max problem.

With the above goal in mind, consider the regularized convex–concave min–max problem

$$\min_{x\in X} \max_{y\in Y} \hat{\Phi}(x,y) + \frac{1}{\lambda}(\mathrm{d}w_1)_{x_-}(x) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y). \tag{41}$$

It is easy to see that the exact solution of (41) determines a solution of (P1) with $\sigma = 0$ in which $\tilde{\lambda} = \lambda$. Letting

$$f_\lambda(x) := \max_{y\in Y} \left\{ \Phi(x,y) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y) \right\}, \quad \forall x \in X, \tag{42}$$

$$g_\lambda(x) := \frac{1}{\lambda}(\mathrm{d}w_1)_{x_-}(x) + \phi(x), \quad \forall x \in \mathcal{X}, \tag{43}$$

it follows from (36), (42) and (43) that (41) is equivalent to (15) with $(f,g) = (f_\lambda, g_\lambda)$. Moreover, conditions A.1 and A.2 are satisfied with $\mu = \eta_1/\lambda$ due to (42) and assumption C.5 which requires w_1 to be η_1-strongly convex on X. Also, the following result establishes the validity of A.3.

PROPOSITION 4.1 *The following statements hold for every* $\lambda > 0$:

(a) *for every* $x \in X$, *the point* $y_\lambda(x)$ *defined as*

$$y_\lambda(x) := \operatorname{argmax}_{y \in Y} \left\{ \Phi(x,y) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y) \right\} \tag{44}$$

is well defined and lies in $Y \cap \operatorname{int}(W_2)$;

(b) *the constant* $L = L_\lambda$ *and function* $\nabla f = \nabla f_\lambda : X \to X$ *defined as*

$$L_\lambda := 2\left(L_{xx} + \frac{\lambda}{\eta_2} L_{xy}^2\right), \quad \nabla f_\lambda(x) := \nabla_x \Phi(x, y_\lambda(x)), \quad \forall x \in X, \tag{45}$$

respectively, satisfy condition A.3 *with* $f = f_\lambda$.

Proof (a) This statement follows from Proposition A.1 of Appendix with $w = (1/\lambda)w_2$, $z_- = y_-$ and $\psi = -\Phi(x,\cdot) + \mathcal{I}_Y$.

(b) This statement follows from Proposition 4.1 of [17] with the function Ψ given by

$$\Psi(x,y) = \Phi(x,y) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y), \quad \forall (x,y) \in X \times Y,$$

and with $\eta = 0$ and $\beta = \eta_2/\lambda$. ∎

Next, we present a scheme for solving (P1) under the assumption that the input z_- lies in $W^0 \cap Z$. The scheme consists on applying the accelerated method of Section 2.2, namely Algorithm 1, to problem (15) with $(f,g) = (f_\lambda, g_\lambda)$, where f_λ and g_λ are as in (42) and (43), respectively, and choosing the function h as $h = (1/\eta_1)(\mathrm{d}w_1)_{x_-}$. Note that g and h defined in this manner satisfy condition A.1 with $\mu = \eta_1/\lambda$ and condition A.4.

ALGORITHM 2 **Accelerated scheme for solving (P1)**
Input: $\sigma \in (0,1]$, $\lambda > 0$, $\tau \in (0,1)$ *and* $z_- = (x_-, y_-) \in W^0 \cap Z$.

(0) *Set* $A_0 = 0$, $k = 1$, $\tilde{\Theta}_0 \equiv 0$, $\tilde{y}_0 = 0$, L_λ *as in* (45), *and* $x_0 = \tilde{x}_0 := x_-$;

(1) *compute* A_k *as in* (16) *with* $\mu = \eta_1/\lambda$, *iterate* $\breve{x}_k$ *as in* (17), *compute* $y_\lambda(\breve{x}_k)$ *according to* (44), *and the affine function* $\tilde{\Theta}_k$ *as*

$$\tilde{\Theta}_k := \frac{A_{k-1}}{A_k}\tilde{\Theta}_{k-1} + \frac{A_k - A_{k-1}}{A_k}[\Phi(\breve{x}_k, y_\lambda(\breve{x}_k)) + \langle \nabla_x \Phi(\breve{x}_k, y_\lambda(\breve{x}_k)), \cdot - \breve{x}_k \rangle_{\mathcal{X}}] \tag{46}$$

(2) *set*

$$\lambda_k = \left(\frac{1}{\lambda} + \frac{1}{\eta_1 A_k}\right)^{-1}, \tag{47}$$

and compute iterates x_k *and* $\tilde{y}_k$ *as*

$$x_k = \operatorname{argmin}\left\{ \tilde{\Theta}_k(x) + \phi(x) + \frac{1}{\lambda_k}(\mathrm{d}w_1)_{x_-}(x) \right\}, \tag{48}$$

$$\tilde{y}_k = \frac{A_{k-1}}{A_k}\tilde{y}_{k-1} + \frac{A_k - A_{k-1}}{A_k} y_\lambda(\breve{x}_k), \tag{49}$$

and $\tilde{x}_k$ *as in* (20);

(3) *if* $\lambda_k \geq \max\{1-\sigma, \tau\}\lambda$, *then compute* $y_k := y_{\lambda_k}(\tilde{x}_k)$ *according to* (44), *set* $\tilde{\lambda} = \lambda_k$, $\tilde{z} = \tilde{z}_k := (\tilde{x}_k, \tilde{y}_k)$, $z = z_k := (x_k, y_k)$ *and*

$$\varepsilon = \varepsilon_k := \hat{\Phi}(\tilde{x}_k, y_k) - \tilde{\Theta}_k(x_k) - \phi(x_k) - \frac{1}{\lambda_k}\langle \nabla(\mathrm{d}w)_{z_k}(z_-), \tilde{z}_k - z_k\rangle,$$

output $\tilde{\lambda}$ *and the triple* $(\tilde{z}, z, \varepsilon)$, *and terminate; otherwise, set* $k \leftarrow k+1$ *and go to step 1.*

end

We now make several remarks about Algorithm 2. First, due to the stopping criterion and (47), Algorithm 2 outputs $\tilde{\lambda} \in [\tau\lambda, \lambda]$. Second, it follows from Proposition A.1 of Appendix with $\psi = \tilde{\Theta}_k + \phi$, $w = w_1/\lambda_k$ and $z_- = x_-$ that x_k given in (48) is well defined and lies in $\mathrm{int}(W_1) \cap X$. Third, due to Proposition A.1 and relations (35), (44) and (48), the output z lies in $W^0 \cap Z$. Fourth, steps 1 and 2 of Algorithm 2 are specializations of steps 1 and 2 of Algorithm 1 to the instance of (15) in which (f, g) is given by (f_λ, g_λ) with f_λ and g_λ as in (42) and (43), respectively. The only difference is the extra computation of $\tilde{y}_k$ in (49) which is used to compute the component $\tilde{z}$ of the output. Fifth, even though the affine function $\tilde{\Theta}_k$ given in (46) and the affine function Θ_k given in (18) with $f = f_\lambda$ are not the same, they both have the same gradient due to (45), and hence the subproblems (48) and (19) are equivalent. Sixth, each iteration of Algorithm 2 before the last one requires solving two subproblems, namely, (48) and one of the form (44), while the last one requires one additional subproblem of the form (44) in step 3. Seventh, when the termination criterion in step 3 is met, this extra step computes the output $\tilde{\lambda}$ and $(\tilde{z}, z, \varepsilon)$ which solve (P1) (see Proposition 4.2). Eighth, another possible way to terminate Algorithm 2 would be to compute the triple $(\tilde{z}, z, \varepsilon) = (\tilde{z}_k, z_k, \varepsilon_k)$ as described in its step 3 at every iteration and check whether $\tilde{\lambda} = \lambda_k$ and this triple satisfy the HPE error criterion (40). (They always satisfy (39) due to Proposition 4.2(a).) The drawback of this stopping criterion is that it requires solving an additional subproblem of the form (44) at every iteration. Our computational benchmark presented in Section 5 is based on the stopping criterion of Algorithm 2.

The following result establishes the correctness and iteration-complexity of Algorithm 2. Its proof is given in Section 4.3.

PROPOSITION 4.2 *The following statements hold for every* $k \geq 1$:

(a) *the scalar* $\tilde{\lambda} = \lambda_k$ *and the triple* $(\tilde{z}, z, \varepsilon) = (\tilde{z}_k, z_k, \varepsilon_k)$ *satisfy inclusion* (39);
(b) *if* $\lambda_k \geq (1-\sigma)\lambda$, *then* $\tilde{\lambda} = \lambda_k$ *and* $(\tilde{z}, z, \varepsilon) = (\tilde{z}_k, z_k, \varepsilon_k)$ *satisfy condition* (40).

COROLLARY 4.3 *By performing at most*

$$\mathcal{O}\left(\left\lceil \sqrt{\frac{\lambda\left(L_{xx} + \frac{\lambda}{\eta_2}L_{xy}^2\right)}{\eta_1}} \right\rceil\right) \tag{50}$$

iterations, Algorithm 2 *terminates with a stepsize* $\tilde{\lambda} > 0$ *and a triple* $(\tilde{z}, z, \varepsilon)$ *which solve* (P1).

Proof When Algorithm 2 terminates in step 3, it easily follows from (47) that the generated output $\tilde{\lambda} = \lambda_k$ satisfies $\tilde{\lambda} \in [\tau\lambda, \lambda]$. Hence, in view of Proposition 4.2, we conclude that Algorithm 2 outputs $\tilde{\lambda}$ and $(\tilde{z}, z, \varepsilon)$ which solves (P1). Moreover, using the estimate $A_k \geq k^2/4L_\lambda$ given in (21), and the definitions of L_λ and λ_k given in (45) and (47), respectively, it is easy to verify that the number of iterations until the stopping criterion of Algorithm 2 occurs is bounded by (50) (when τ and σ are viewed as universal constants such that $\max\{1-\sigma, \tau\}$ is neither close to zero nor one). ■

4.2 *An accelerated NE-HPE instance for solving SP($\hat{\Phi}$)*

This subsection describes an accelerated instance of the NE-HPE framework for solving the saddle-point problem SP($\hat{\Phi}$) and its corresponding iteration-complexity result. It also discusses optimal ways of choosing the prox stepsize in order to minimize the overall inner iteration-complexity of the instance.

We start by stating an accelerated instance of the NE-HPE framework for solving SP($\hat{\Phi}$) which computes the required stepsize λ_j and triple $(\tilde{z}_j, z_j, \varepsilon_j)$ in its step 1 with the aid of Algorithm 2.

Accelerated NE-HPE method for the saddle-point problem

(0) Let $z_0 \in W^0$, $\lambda > 0$, $\sigma \in (0,1]$ and $\tau \in (0,1)$ be given and set $j = 1$;
(1) invoke Algorithm 2 with input σ, λ, τ and $z_- = z_{j-1}$ to obtain a stepsize $\lambda_j \in [\tau\lambda, \lambda]$ and a triple $(\tilde{z}_j, z_j, \varepsilon_j)$ satisfying (34) and (27);
(2) set $j \leftarrow j + 1$, and go to step 1.

end

In view of Proposition 4.2, the accelerated NE-HPE method satisfies the error conditions (34) and (27) of step 1 of the NE-HPE framework. Therefore, the accelerated NE-HPE method is clearly a special case of the NE-HPE framework. It follows that the ergodic (outer) convergence rate bound for the accelerated NE-HPE method is as described in Theorem 3.5.

THEOREM 4.4 *Consider the sequence* $\{(\tilde{z}_j, z_j, \varepsilon_j)\}$ *generated by the accelerated NE-HPE method applied to a saddle-point problem SP*($\hat{\Phi}$) *which has the composite structure* (35) *and* (36) *and satisfies the assumptions* C.1–C.3. *Consider also the ergodic sequence* $\{(\tilde{z}_j^a, r_j^a, \varepsilon_j^a)\}$ *computed as in Lemma* 3.4 *and the sequence* $\{\tilde{\varepsilon}_j\}$ *computed as in* (31) *with* $\Omega = Z$. *Also, let* $R := R(z_0; Z)$ *where* $R(z_0; Z)$ *is as in* (30). *Then, the following statements hold:*

(a) *for every positive scalar* $\bar{\varepsilon}$, *there exists*

$$j_0 = \mathcal{O}\left(\left\lceil \frac{R}{\lambda\bar{\varepsilon}} \right\rceil\right)$$

such that for every $j \geq j_0$, $(\tilde{z}_j^a, \tilde{\varepsilon}_j)$ *is an* $\bar{\varepsilon}$*-saddle-point of* $\hat{\Phi}$;
(b) *each iteration of the accelerated NE-HPE method performs at most*

$$\mathcal{O}\left(\left\lceil \sqrt{\frac{\lambda\left(L_{xx} + \frac{\lambda}{\eta_2}L_{xy}^2\right)}{\eta_1}} \right\rceil\right)$$

inner iterations.

As a consequence, the accelerated NE-HPE method finds an $\bar{\varepsilon}$*-saddle-point of* $\hat{\Phi}$ *by performing no more than*

$$\mathcal{O}\left(\left\lceil \sqrt{\frac{\lambda\left(L_{xx} + \frac{\lambda}{\eta_2}L_{xy}^2\right)}{\eta_1}} \right\rceil \left\lceil \frac{R}{\lambda\bar{\varepsilon}} \right\rceil\right) \tag{51}$$

inner iterations.

Proof Since the accelerated NE-HPE method is a special instance of the NE-HPE framework, (a) follows from Corollary 3.7 (b) and from the fact that $\lambda_j \geq \tau\lambda$ for every $j \geq 1$. Statement

(b) follows from Proposition 4.2. The last assertion of the theorem follows immediately from (a) and (b). ■

We end this subsection by making a remark about the complexity bound (51) in light of the one obtained in relation (4.4) of [22]. Clearly, when $\lambda = R/\bar{\varepsilon}$, the complexity bound (51) reduces to

$$\mathcal{O}\left(1 + \frac{RL_{xy}}{\bar{\varepsilon}\sqrt{\eta_1\eta_2}} + \sqrt{\frac{RL_{xx}}{\bar{\varepsilon}\eta_1}}\right). \tag{52}$$

It turns out that, by suitably scaling the distance generating functions w_1 and w_2, this bound reduces to

$$\mathcal{O}\left(1 + \frac{\sqrt{R_1R_2}L_{xy}}{\bar{\varepsilon}\sqrt{\eta_1\eta_2}} + \sqrt{\frac{R_1L_{xx}}{\bar{\varepsilon}\eta_1}}\right), \tag{53}$$

where

$$R_1 := \max\{(\mathrm{d}w_1)_{x_0}(x) : x \in X\}, \quad R_2 := \max\{(\mathrm{d}w_2)_{y_0}(y) : y \in Y\}.$$

The latter bound generalizes the one in relation (4.4) of [22] which was shown to be valid only for a special bilinear structured case of $\mathrm{SP}(\hat{\Phi})$.

To obtain the bound (53), consider the distance generating functions

$$w_{1,\theta} := \theta w_1, \quad w_{2,\theta} := \theta^{-1}w_2,$$

where $\theta > 0$ is a fixed parameter. Clearly, $w_{1,\theta}$ (resp., $w_{2,\theta}$) is a distance generating function with domain W_1 (resp., W_2) which is $\theta\eta_1$-strongly convex on X (resp., $\theta^{-1}\eta_2$-strongly convex on Y). In this case, R becomes

$$R = \theta R_1 + \theta^{-1}R_2.$$

Hence, choosing $\theta = (R_2/R_1)^{1/2}$, the quantities R, η_1 and η_2 in this case reduce to

$$R = 2\sqrt{R_1R_2}, \quad \eta_1 = \sqrt{\frac{R_2}{R_1}}\eta_1, \quad \eta_2 = \sqrt{\frac{R_1}{R_2}}\eta_2,$$

and hence (52) reduces to (53).

4.3 *Proof of Proposition 4.2*

This subsection proves Proposition 4.2.

Given the input of problem (P1) and a point $\tilde{z} \in Z \cap W^0$, the following result describes a way of generating a pair (z, ε) in terms of $\hat{\Phi}_{\tilde{z}}$ (as in (13)) and a convex function minorizing it so that (39) holds and (40) is satisfied whenever a suitable sufficient condition holds. This result will be used to show that Algorithm 2 obtains a scalar $\tilde{\lambda}$ and a triple $(\tilde{z}, z, \varepsilon)$ which solve (P1).

LEMMA 4.5 *Consider the distance generating function w as in* (37) *and let $\tilde{\lambda} > 0$, $z_- \in W^0$ and $\tilde{z} \in Z \cap W^0$ be given. Assume that there exists a proper closed convex function $\Gamma_{\tilde{z}}$ such that*

dom $\Gamma_{\tilde z} = Z$ *and* $\Gamma_{\tilde z} \le \hat\Phi_{\tilde z}$ *where* $\hat\Phi_{\tilde z}$ *is as in* (13). *Moreover, define the quantities*

$$z := \operatorname{argmin}_u \left\{ \Gamma_{\tilde z}(u) + \frac{1}{\tilde\lambda}(\mathrm{d}w)_{z_-}(u) \right\}, \tag{54}$$

$$\varepsilon := -\Gamma_{\tilde z}(z) - \langle r, \tilde z - z\rangle, \tag{55}$$

where

$$r := \frac{1}{\tilde\lambda}\nabla(\mathrm{d}w)_z(z_-). \tag{56}$$

Then, the following statements hold:

(a) z *is well defined and* $z \in Z \cap W^0$, *and hence* r *is well defined ;*
(b) $\varepsilon \in [0,\infty)$ *and* (39) *holds.*
(c) *if, in addition, for a given scalar* $\sigma \ge 0$, *we have*

$$\frac{1-\sigma}{\tilde\lambda}(\mathrm{d}w)_{z_-}(\tilde z) \le \inf\left\{ \Gamma_{\tilde z}(u) + \frac{1}{\tilde\lambda}(\mathrm{d}w)_{z_-}(u) : u \in \mathcal{Z} \right\}, \tag{57}$$

then (40) *holds.*

Proof (a) The assumptions of the lemma clearly imply that $\tilde z \in Z \cap W^0 = \operatorname{dom}\ \Gamma_{\tilde z} \cap W^0$. Hence, using the latter inclusion, assumption C.5 that w is strongly convex over $Z = \operatorname{dom}\hat\Gamma_{\tilde z}$ and relation (56), we conclude from Proposition A.1 of Appendix with $\psi = \tilde\lambda\Gamma_{\tilde z}$ that z is well-defined, $z \in Z \cap W^0$ and satisfies

$$r \in \partial\Gamma_{\tilde z}(z). \tag{58}$$

(b) Clearly, $\varepsilon < \infty$ due to (55) and the fact that $\Gamma_{\tilde z}$ is proper. Using the assumption that $\Gamma_{\tilde z} \le \hat\Phi_{\tilde z}$, and relations (55) and (58), we conclude that

$$\hat\Phi_{\tilde z}(u) \ge \Gamma_{\tilde z}(u) \ge \Gamma_{\tilde z}(z) + \langle r, u - z\rangle = \langle r, u - \tilde z\rangle - \varepsilon, \quad \forall u \in \mathcal{Z}. \tag{59}$$

The latter conclusion together with the fact that (13) implies that $\hat\Phi_{\tilde z}(\tilde z) = 0$ then yield (39). Clearly, (59) with $u = \tilde z$ implies that $\varepsilon \ge 0$.

(c) Note that (54) and (57) imply that

$$\frac{1-\sigma}{\tilde\lambda}(\mathrm{d}w)_{z_-}(\tilde z) \le \Gamma_{\tilde z}(z) + \frac{1}{\tilde\lambda}(\mathrm{d}w)_{z_-}(z). \tag{60}$$

Moreover, relations (24)–(56) imply that

$$(\mathrm{d}w)_{z_-}(\tilde z) - (\mathrm{d}w)_{z_-}(z) = (\mathrm{d}w)_z(\tilde z) + \langle\nabla(\mathrm{d}w)_{z_-}(z), \tilde z - z\rangle = (\mathrm{d}w)_z(\tilde z) - \tilde\lambda\langle r, \tilde z - z\rangle. \tag{61}$$

Now, using (55), (60) and (61), we conclude that

$$\begin{aligned}(\mathrm{d}w)_z(\tilde z) + \tilde\lambda\varepsilon &= (\mathrm{d}w)_z(\tilde z) - \tilde\lambda[\Gamma_{\tilde z}(z) + \langle r, \tilde z - z\rangle] \\ &= (\mathrm{d}w)_{z_-}(\tilde z) - (\mathrm{d}w)_{z_-}(z) - \tilde\lambda\Gamma_{\tilde z}(z) \le \sigma(\mathrm{d}w)_{z_-}(\tilde z),\end{aligned}$$

and hence that (40) holds. ∎

We now establish the following technical lemma which guarantees that the output of Algorithm 2 satisfies the conditions described in Lemma 4.5.

LEMMA 4.6 *For every $k \geq 1$, the scalar λ_k, triple $(\tilde{z}_k, z_k, \varepsilon_k)$ and function $\tilde{\Theta}_k$ generated by Algorithm* 2 *satisfy the following statements:*

(a) $\tilde{z}_k, z_k \in W^0 \cap Z$;
(b) $\Gamma_{\tilde{z}_k} : \mathcal{Z} \to [-\infty, \infty]$ *defined as*

$$\Gamma_{\tilde{z}_k}(z) := \begin{cases} \tilde{\Theta}_k(x) + \phi(x) - \hat{\Phi}(\tilde{x}_k, y), & \forall z = (x, y) \in Z, \\ +\infty & \text{otherwise} \end{cases} \tag{62}$$

is a proper closed convex function such that dom $\Gamma_{\tilde{z}_k} = Z$ *and* $\Gamma_{\tilde{z}_k} \leq \hat{\Phi}_{\tilde{z}_k}$ *where the latter function is defined in* (13);
(c) $\tilde{\lambda} = \lambda_k$, $(\tilde{z}, z, \varepsilon) = (\tilde{z}_k, z_k, \varepsilon_k)$ *and* $\Gamma_{\tilde{z}} = \Gamma_{\tilde{z}_k}$ *as in* (62) *satisfy* (54) *and* (55).

As a consequence, $\tilde{\lambda} = \lambda_k$ *and* $(\tilde{z}, z, \varepsilon) = (\tilde{z}_k, z_k, \varepsilon_k)$ *satisfy* (39) *for every* $k \geq 1$.

Proof (a) First observe that the second remark following Algorithm 2 and Proposition 4.1(a) implies that $x_k \in X \cap \text{int}(W_1)$ and $y_\lambda(\breve{x}_k) \in Y \cap \text{int}(W_2)$ for every $k \geq 1$. Also, note that by (20) (resp., (49)), we have

$$\tilde{x}_k = \sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_k} x_i, \quad \tilde{y}_k = \sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_k} y_\lambda(\breve{x}_i) \tag{63}$$

and hence $\tilde{x}_k$ (resp., $\tilde{y}_k$) is a convex combination of the points x_i (resp., $y_\lambda(\breve{x}_i)$), $i = 1, \ldots, k$. Since the set $X \cap \text{int}(W_1)$ (resp., $Y \cap \text{int}(W_2)$) is convex, we conclude that $\tilde{z}_k = (\tilde{x}_k, \tilde{y}_k) \in Z \cap W^0$. The conclusion that $z_k \in Z \cap W^0$ follows from the inclusion $x_k \in X \cap \text{int}(W_1)$, the definition of y_k in step 3 of Algorithm 2 and Proposition 4.1(a).

(b) Using (13), the definition of $\Gamma_{\tilde{z}_k}$ in (62), condition C.2, the assumption that ϕ is a proper closed convex function, and the fact that $\tilde{\Theta}_k$ is an affine function, we easily that $\Gamma_{\tilde{z}_k}$ is a proper closed convex function with dom$\Gamma_{\tilde{z}_k} = Z$. Using the definitions of $\tilde{\Theta}_k$ and $\tilde{y}_k$ given in (46) and (48) as well as the fact that $\Phi(\cdot, y) - \Phi(x, \cdot)$ is convex for every $(x, y) \in Z$, we see that for every $x \in X$

$$\begin{aligned} \tilde{\Theta}_k(x) &= \sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_k} [\Phi(\breve{x}_i, y_\lambda(\breve{x}_i)) + \langle \nabla_x \Phi(\breve{x}_i, y_\lambda(\breve{x}_i)), x - \breve{x}_i \rangle_{\mathcal{X}}] \\ &\leq \sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_k} [\Phi(x, y_\lambda(\breve{x}_i))] \leq \Phi\left(x, \sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_k} y_\lambda(\breve{x}_i)\right) = \Phi(x, \tilde{y}_k). \end{aligned}$$

By (13), (36) and (62), we see that $\Gamma_{\tilde{z}_k}$ minorizes $\hat{\Phi}_{\tilde{z}_k}$ if and only if $\tilde{\Theta}_k \leq \Phi(\cdot, \tilde{y}_k)$, and hence (b) follows.

(c) This statement follows directly from the definitions of z_k and ε_k and relations (44), (48) and (62).

The last conclusion of the lemma follows directly from Lemma 4.5(b). ■

LEMMA 4.7 *If at the kth step of Algorithm* 2, *the condition* $\lambda_k \geq (1 - \sigma)\lambda$ *is satisfied, then* $\tilde{\lambda} = \lambda_k$ *and* $(\tilde{z}, z, \varepsilon) = (\tilde{z}_k, z_k, \varepsilon_k)$ *satisfy* (57), *and, as a consequence, condition* (40).

Proof The last conclusion of the lemma follows from Lemma 4.5(c). Moreover, in view of (38) and (62), the first conclusion of lemma is equivalent to the condition that

$$\frac{1-\sigma}{\lambda_k}[(\mathrm{d}w_1)_{x_-}(\tilde{x}_k) + (\mathrm{d}w_2)_{y_-}(\tilde{y}_k)] \\ \leq \tilde{\Theta}_k(x) + \phi(x) - \hat{\Phi}(\tilde{x}_k, y) + \frac{1}{\lambda_k}[(\mathrm{d}w_1)_{x_-}(x) + (\mathrm{d}w_2)_{y_-}(x)] \quad \forall (x,y) \in Z. \tag{64}$$

To show the latter condition, assume that $\lambda_k \geq (1-\sigma)\lambda$. For the purpose of applying Proposition 2.6, let

$$x_0 = x_-, \quad f = f_\lambda, \quad g = g_\lambda, \quad h = (1/\eta_1)(\mathrm{d}w_1)_{x_-}, \tag{65}$$

where f_λ and g_λ are as in (42) and (43), respectively. Note that the assumption that w_1 is strongly convex on X (see C.5) implies that x_0, g and h satisfy conditions A.4–A.6. Since the set X and the functional pair (f, g) satisfy conditions A.1–A.3, and Algorithm 2 corresponds to Algorithm 1 applied to (15) with x_0, f, g and h as above (see the fourth remark following Algorithm 2), it follows from Proposition 2.6, and relations (36), (42), (43) and (65), that

$$\sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_i}[f_\lambda(\breve{x}_i) + \langle \nabla f_\lambda(\breve{x}_i), x - \breve{x}_i\rangle_{\mathcal{X}}] + g_\lambda(x) + \frac{1}{\eta_1 A_k}(\mathrm{d}w_1)_{x_-}(x) \geq (f_\lambda + g_\lambda)(\tilde{x}_k) \\ \geq \Phi(\tilde{x}_k, y) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y) + \frac{1}{\lambda}(\mathrm{d}w_1)_{x_-}(\tilde{x}_k) + \phi(\tilde{x}_k) \\ = \hat{\Phi}(\tilde{x}_k, y) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y) + \frac{1}{\lambda}(\mathrm{d}w_1)_{x_-}(\tilde{x}_k), \quad \forall (x,y) \in Z. \tag{66}$$

Using (46), (63) and the convexity of $(\mathrm{d}w_2)_{y_-}(\cdot)$, we have

$$\tilde{\Theta}_k(x) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(\tilde{y}_k) \\ \geq \sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_i}\left[\Phi(\breve{x}_i, y_\lambda(\breve{x}_i)) + \langle \nabla_x \Phi(\breve{x}_i, y_\lambda(\breve{x}_i)), x - \breve{x}_i\rangle_{\mathcal{X}} - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y_\lambda(\breve{x}_i))\right] \\ = \sum_{i=1}^{k} \frac{A_i - A_{i-1}}{A_i}[f_\lambda(\breve{x}_i) + \langle \nabla f_\lambda(\breve{x}_i), x - \breve{x}_i\rangle_{\mathcal{X}}], \tag{67}$$

where the last equality is due to (42), (44) and (45). Combining (66) and (67), and using (43) and (47), we then conclude that

$$\hat{\Phi}(\tilde{x}_k, y) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(y) + \frac{1}{\lambda}(\mathrm{d}w_1)_{x_-}(\tilde{x}_k) \\ \leq \tilde{\Theta}_k(x) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(\tilde{y}_k) + g_\lambda(x) + \frac{1}{\eta_1 A_k}(\mathrm{d}w_1)_{x_-}(x) \\ \leq \tilde{\Theta}_k(x) - \frac{1}{\lambda}(\mathrm{d}w_2)_{y_-}(\tilde{y}_k) + \phi(x) + \frac{1}{\lambda_k}(\mathrm{d}w_1)_{x_-}(x), \quad \forall (x,y) \in Z.$$

Now, using (47) and the assumption that $\lambda_k \geq (1-\sigma)\lambda$, we have that $(1-\sigma)/\lambda_k \leq 1/\lambda \leq 1/\lambda_k$. Combining these inequalities with the previous relation, we obtain (64). ■

We now end this subsection by observing that Proposition 4.2 follows immediately from Lemmas 4.6 and 4.7.

5. Numerical experiments

This section presents computational results showing the numerical performance of the accelerated NE-HPE method on a collection of saddle-point problems. All the computational results were obtained using MATLAB R2014a on a Windows 64 bit machine with processor Intel 2.16 GHz with 4 GB memory.

The accelerated NE-HPE method (referred to as ACC-HPE) is compared with Nesterov's smoothing scheme [22] (referred to as NEST). We have implemented both algorithms based on the Euclidean distance and the Bregman distance induced by the Kullback–Leibler divergence, namely, $\mathrm{d}w_{z^2}(z^1) = \sum_i z_i^1 \log(z_i^1/z_i^2) + z_i^1 - z_i^2$. Our computational results then consider four variants, namely, E-ACC-HPE, KL-ACC-HPE, E-NEST and KL-NEST, where the ones starting with E- (resp., KL-) are the ones based on the Euclidean (resp., Kullback–Leibler log distance). The implementation of all these variants are based on Nesterov's accelerated gradient method discussed in Section 2.2, namely, Algorithm 1. However, a restarting feature is added to the implementation of both E-NEST and KL-NEST. Essentially, Algorithm 1 is restarted from the most recent iterate whenever the objective function of the smoothing subproblem fails to decrease. We have observed that adding the restarting feature to both E-NEST and KL-NEST resulted in a minor improvement of their performance (which is probably due to the smoothing nature of these two variants). The restarting feature was not added to the implementation of the HPE variants since it did not improve their performance (which is probably due to the fact that they perform a relatively small number of inner iterations to solve each of their subproblems). Finally, we could have compared the aforementioned variants with the accelerated method of [7]. However, since the E-ACC-HPE variant was compared with the accelerated method of [7] in reference [13] and the two methods were found to have similar performance, we have decided to leave the accelerated method of [7] out of our computational study in this section.

To improve the performance of the KL-variants, we have used the adaptive scheme for choosing the parameter L given in [32], i.e. the initial value of L is set to a fraction of the true Lipschitz constant value and is increased by a factor of 2 whenever it fails to satisfy a certain convergence criterion (see Equations (23) and (45) of [32]). The fraction $1/2^9$ was used in our experiments. The same scheme was not used for the E-variants since we have observed that it does not improve their performance. The value of L at the last iteration divided by the true Lipschitz constant varied between 1/64 and 1 in our experiments. More specifically, this ratio was 1/64 for 1 instance, 1/32 for 3 instances, 1/8 for 1 instance, 1/4 for 4 instances, 1/2 for 12 instances and 1 for the remaining instances.

The following four subsections report computational results on the following classes of problems: (a) zero-sum matrix game, (b) quadratic game, (c) vector–matrix saddle-point and (d) minimizing the maximum of convex quadratic functions. The results are reported in tables and in performance profiles (see [9]). We recall the following definition of a performance profile. For a given instance, a method A is said to be at most x times slower than method B, if the time taken by method A is at most x times the time taken by method B. A point (x, y) is in the performance profile curve of a method if it can solve exactly 100% of all the tested instances x times slower than any other competing method.

For the three first problem classes, the stopping criterion used to terminate all methods at the kth iteration is

$$\max_{y\in Y} \hat{\Phi}(\tilde{x}_k, y) - \min_{x\in X} \hat{\Phi}(x, \tilde{y}_k) \leq \bar{\varepsilon}.$$

The use of this criterion for the second and third problem classes is not the best strategy from the computational point of view, since the computation of the dual function involves solving a quadratic programming problem over the unit simplex. Note that our method has the ability to

compute at every iteration a pair $((\tilde{x}_k, \tilde{y}_k), \varepsilon_k)$ such that the above inequality holds with $\bar{\varepsilon} = \varepsilon_k$ and hence the above termination criterion will be satisfied whenever $\varepsilon_k \leq \bar{\varepsilon}$. Since the usual description of Nesterov's smoothing scheme generates $(\tilde{x}_k, \tilde{y}_k)$ but not ε_k, we have opted for the gap criterion but adopted the convention of excluding the effort to evaluate the dual functions from the reported cpu times. Since we do not implement Nesterov's smoothing schemes for the fourth class of problems, we use for this class the stopping criterion $\tilde{\varepsilon}_j \leq \bar{\varepsilon}$ where $\tilde{\varepsilon}_j$ is as in (31).

We let $\mathbb{R}^n$ denote the n-dimensional Euclidean space and $\mathcal{S}^n$ denote the linear space of $n \times n$ real symmetric matrices. The unit simplex in $\mathbb{R}^n$ is defined as

$$\Delta_n := \left\{ x \in \mathbb{R}^n : \sum_{i=1}^{n} x_i = 1,\ x \geq 0 \right\}. \tag{68}$$

5.1 *Zero-sum matrix game problem*

This subsection compares the performance of the four variants on instances of the zero-sum matrix game problem

$$\min_{x \in \Delta_n} \max_{y \in \Delta_m} \langle x, Ay \rangle,$$

where A is a real $n \times m$ matrix and Δ_n, Δ_m are given in (68). The matrices were generated so that its elements are non-zero with probability p and the non-zero ones are randomly generated in the interval $[-1, 1]$. We have tested the methods for a set of problems with different sizes of matrices and different values of p. The tolerance used here was $\bar{\varepsilon} = 10^{-4}$.

Table 1 reports the results of the four variants applied to several instances of this problem with different sizes of matrices and different values of p.

Figure 1 gives the performance profile for the same set of instances. Overall, it shows that the accelerated NE-HPE variants perform better than NEST variants on this set of zero-sum games instances.

Finally, we have observed that the instances in Table 1 which E-ACC-HPE performed too many iterations to solve (namely the first, second and last instances) are ones that satisfy $n > m$. By running the algorithm with the transpose matrix so that now $n < m$, we have observed a significant decrease in the number of iterations performed by E-ACC-HPE. We have, however, not reported these findings although it can be seen from Table 1 that instances with $n < m$ are usually easy for E-ACC-HPE to solve.

Table 1. Test results for the zero-sum matrix game problem ($\bar{\epsilon} = 1\text{e} - 4$).

Size			E-ACC-HPE		E-NEST		KL-ACC-HPE		KL-NEST	
n	m	p	Time	Iter.	Time	Iter.	Time	Iter.	Time	Iter.
1000	100	0.01	174.3864	131,359	138.119	106,259	91.5114	32,465	123.529	69,887
1000	100	0.02	305.3871	254,661	194.055	95,060	74.2447	55,461	80.7436	52,114
1000	1000	0.01	1.5908	978	339.334	205,336	167.1044	82,073	184.826	79,856
1000	1000	0.02	3.2378	1836	396.728	192,183	126.0967	56,404	139.102	62,115
1000	10,000	0.01	2.9961	495	195.446	24,701	451.2670	49,351	452.116	68,855
1000	10,000	0.02	7.1158	978	440.115	44,226	456.4244	51,364	526.755	58,229
10,000	100	0.01	19.2045	6363	582.755	169,582	177.5826	54,432	193.402	52,268
10,000	100	0.02	8.8340	2493	1075.75	295,855	176.9932	41,964	244.735	60,115
10,000	1000	0.01	2.8091	531	1219.55	405,663	356.6100	73,020	1204.42	70,718
10,000	1000	0.02	11,717.8	594,684	2967.5	379,056	1104.8	77,524	1048.45	69,998

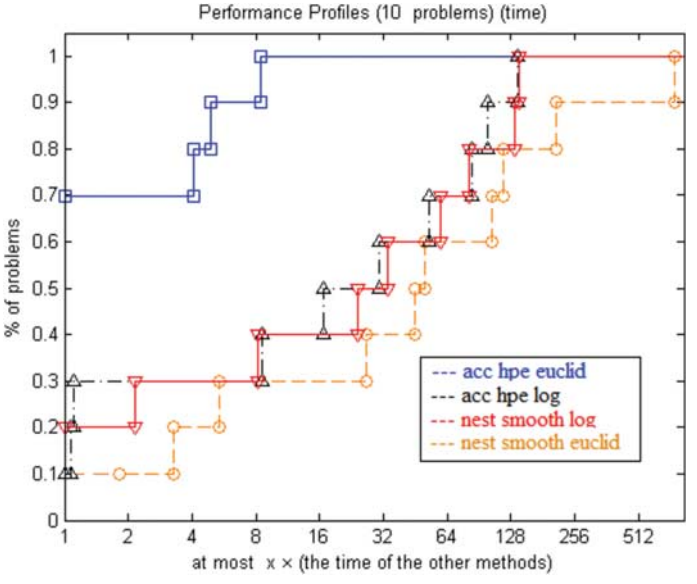

Figure 1. Performance profile for the zero-sum matrix problem ($\bar{\epsilon} = 1\text{e} - 4$).

5.2 *Quadratic game problem*

This subsection compares the four variants on instances of the quadratic game problem

$$\min_{x\in\Delta_n}\max_{y\in\Delta_m}\frac{1}{2}\|Bx\|^2 + \langle x, Ay\rangle$$

for different sizes of matrices and different values of p. The matrices were generated in the same way as in the zero-sum matrix game problem (see Section 5.1). The tolerance used was $\bar{\varepsilon} = 10^{-3}$.

Table 2 reports the results of the four variants applied to several instances of this problem with different sizes of matrices and different values of p.

Figure 2 gives the performance profile for the same set of instances. It shows that the accelerated NE-HPE variant based on the Euclidean (resp., Kullback–Leibler log) distance performs better than the NEST variant based on the Euclidean (resp., Kullback–Leibler log) distance on this set of quadratic game instances.

Table 2. Test results for the quadratic game problem ($\bar{\epsilon} = 1\text{e} - 3$).

Size			E-ACC-HPE		E-NEST		KL-ACC-HPE		KL-NEST	
n	m	p	Time	Iter.	Time	Iter.	Time	Iter.	Time	Iter.
100	100	0.01	0.1814	25	0.4271	210	0.3035	725	7.7894	2760
100	1000	0.01	0.2905	45	1.2477	3265	0.7193	1260	9.3556	2275
1000	100	0.01	0.3812	765	0.7559	210	15.3816	1925	10.2989	2055
1000	1000	0.01	0.3706	60	11.0226	4820	12.0675	1245	12.2159	2255
100	100	0.1	0.4913	75	11.6642	3060	6.7952	720	9.2554	1685
100	1000	0.1	0.3342	70	13.8469	6630	7.8718	1055	12.0627	1635
1000	100	0.1	10.4097	1140	13.7982	5065	10.4403	1940	11.8599	1925
1000	1000	0.1	11.5278	1800	15.7258	6150	9.7203	1045	11.9566	1750
100	100	0.2	5.5546	460	9.2208	4690	8.7195	695	11.5903	1465
100	1000	0.2	0.3995	95	26.9528	9605	9.4831	1025	12.1928	1390
1000	100	0.2	13.7596	2100	22.1286	6870	8.3569	685	17.6890	1795
1000	1000	0.2	19.4915	3805	25.7224	4370	9.7182	1145	17.5225	2625

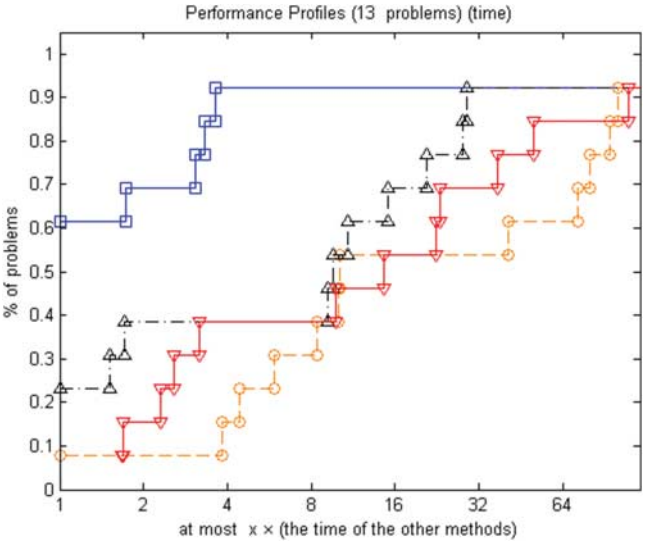

Figure 2. Performance profile for the quadratic game problem ($\bar{\epsilon} = 1e-3$).

5.3 *Vector–matrix saddle-point problem*

This subsection compares the four variants on instances of the vector–matrix saddle-point problem. Given $c \in \mathbb{R}^n$, a real $n \times n$ matrix B and a linear operator $\mathcal{A} : \mathbb{R}^n \to \mathcal{S}^m$, the vector–matrix saddle-point problem is

$$\min_{x \in \Delta_n} \frac{1}{2}\|Bx + c\|^2 + \theta_{\max}(\mathcal{A}(x)),$$

where $\theta_{\max}(\mathcal{A}(x))$ denotes the largest eigenvalue of $\mathcal{A}(x)$. Such problem is equivalent to the saddle-point problem

$$\min_{x \in \Delta_n} \max_{y \in \Omega} \frac{1}{2}\|Bx + c\|^2 + \langle \mathcal{A}(x), y\rangle,$$

where $\Omega := \{y \in \mathcal{S}^m : \mathrm{tr}(y) = 1, y \text{ is positive definite}\}$ and Δ_n is given in (68). We have tested the four variants on a set of problems where the matrices B and $\mathcal{A}_i := \mathcal{A}(e_i)$, $i = 1, \ldots, n$, were generated so that its elements are non-zero with probability 0.1 and the non-zero ones are randomly generated in the interval $[-1, 1]$. (Here, e_i denotes the ith unit n dimensional vector.) The tolerance used was $\bar{\varepsilon} = 10^{-3}$.

Table 3 reports the results of the four variants applied to several instances of this problem with different sizes of matrices.

Figure 3 gives the performance profile for the same set of instances. It also shows that the accelerated NE-HPE variants perform better than NEST variants on this set of vector–matrix saddle-point instances.

5.4 *Minimizing the maximum of convex quadratic functions*

Consider m convex quadratic functions given as

$$f_i(x) = \frac{1}{2}x^{\mathrm{T}}Q_i x + c_i^t x + \gamma_i, \quad \forall x \in \mathbb{R}^n$$

where for each $i = 1, \ldots, m$, Q_i are $n \times n$ positive semidefinite symmetric matrices, $c_i \in \mathbb{R}^n$ and $\gamma_i \in \mathbb{R}$. Define $f(x) = \max\{f_i(x) : i = 1, \ldots, m\}$. In this subsection, we use the accelerated

Table 3. Test results for the vector-matrix saddle-point game problem ($\bar{\epsilon} = 1\text{e}-3$).

Size		E-ACC-HPE		E-NEST		KL-ACC-HPE		KL-NEST	
n	m	Time	Iter.	Time	Iter.	Time	Iter.	Time	Iter.
50	50	31.7197	3090	6.9952	871	21.638	3355	23.0662	2338
50	100	5.0795	207	25.8155	1061	26.4985	1585	25.2686	1786
50	200	273.052	1870	265.815	1856	108.54	1575	189.779	2085
100	50	2.8717	257	18.7195	1651	15.2115	2240	30.4799	2677
100	100	1.7303	103	14.7265	876	63.6484	2300	57.2244	2087
100	200	169.718	860	271.8164	1396	129.0064	1755	198.1275	2163
200	50	31.9003	1587	78.1515	3329	19.7539	2520	34.9962	2728
200	100	2.8319	149	15.8466	811	49.9772	1238	80.1758	2654
200	200	92.403	710	118.0432	986	147.0955	1930	209.543	2027

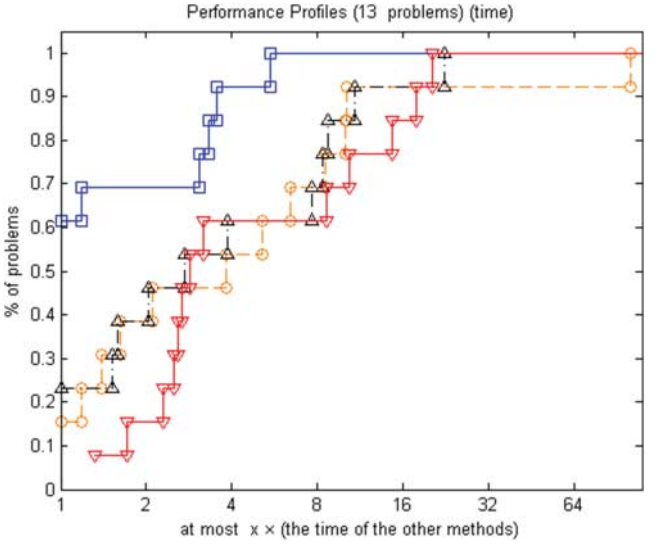

Figure 3. Performance profile for the vector–matrix saddle-point problem ($\bar{\epsilon} = 1\text{e}-3$).

NE-HPE method to solve the optimization problem

$$\min\{f(x) : x \in \Delta_n\},$$

where Δ_n is defined in (68). Clearly, the above problem is equivalent to the convex–concave saddle-point problem

$$\min_{x\in\Delta_n} \max_{y\in\Delta_m} \sum_{i=1}^{m} y_i f_i(x), \tag{69}$$

which is generally not a special case of the class of problems (3). However, it is a special case of the class of problems considered in Section 4 in which $X \times Y = \Delta_n \times \Delta_m$, $\phi = \mathcal{I}_{\Delta_n}$ and $\Phi(x, y) = \sum_{i=1}^{m} y_i f_i(x)$ for all $(x, y) \in \mathbb{R}^n \times \Delta_m$. Hence, problem (69) can be solved by the accelerated NE-HPE method of Section 4.2.

Results are reported in Table 4 for eight instances of the above problem in which Q_i and c_i, $i = 1, \ldots, m$, are randomly generated for different pairs (n, m). The tolerance used was $\bar{\varepsilon} = 10^{-4}$.

Table 4. Test results for the maximum of convex quadratic functions problem ($\bar{\epsilon} = 1e-4$).

Size		E-ACC-HPE		KL-ACC-HPE	
n	m	Time	Iter.	Time	Iter.
10	10	3.4656	145	22.0530	711
10	20	16.4593	304	23.7624	4476
20	10	3.4124	446	16.7082	2197
20	20	41.2514	3342	131.5449	9342
50	10	25.8993	3730	88.3269	1294
50	20	64.4898	5726	196.6463	15,805
100	10	85.4456	9416	138.7094	15,708
100	20	255.2760	13,715	676.754	43,975

6. Concluding remarks

In this section, we make some final remarks about the computational results described in this section. We have shown in Section 4.2 that the accelerated NE-HPE method has the same complexity as Nesterov's smoothing technique of [22]. The experiment results of this section involving three problem sets have shown that the accelerated NE-HPE variants outperform the variants of Nesterov's smoothing scheme. The experiments have also shown that the accelerated NE-HPE variant based on the Euclidean distance performs better than the accelerated NE-HPE variant based on the Kullback–Leibler log distance.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

The work of O. Kolossoski was partially supported by Capes [grant number 99999.003842/2014-02]. The work of R.D.C. Monteiro was partially supported by NSF [grant number CMMI-1300221].

References

[1] A. Auslender and M. Teboulle, *Asymptotic Cones and Functions in Optimization and Variational Inequalities*, Springer Monographs in Mathematics, Springer, New York, 2003.
[2] R.S. Burachik, A.N. Iusem, and B.F. Svaiter, *Enlargement of monotone operators with applications to variational inequalities*, Set-Valued Anal. 5(2) (1997), pp. 159–180.
[3] R.S. Burachik, C.A. Sagastizábal, and B.F. Svaiter, *ϵ-enlargements of maximal monotone operators: Theory and applications, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods (Lausanne, 1997)*, Appl. Optim., Vol. 22, Kluwer Academics, Dordrecht, 1999, pp. 25–43.
[4] Y. Censor and S.A. Zenios, *Proximal minimization algorithm with D-functions*, J. Optim. Theory Appl. 73 (1992). 451–464.
[5] G. Chen and M. Teboulle, *Convergence analysis of a proximal-like minimization algorithm using Bregman functions*, SIAM J. Optim. 3 (1993), pp. 538–543.
[6] Y. Chen, G. Lan, and Y. Ouyang, *Accelerated schemes for a class of variational inequalities*, Math. Program. (2014). submitted to.
[7] Y. Chen, G. Lan, and Y. Ouyang, *Optimal primal-dual methods for a class of saddle point problems*, SIAM J. Optim. 24(4) (2014), pp. 1779–1814.
[8] C.D. Dang and G. Lan, *On the convergence properties of non-euclidean extragradient methods for variational inequalities with generalized monotone operators*, Comput. Optim. Appl. 60(2) (2015), pp. 277–310.
[9] E.D. Dolan and J.J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program. 91(2) (2002), pp. 201–213.
[10] J. Eckstein, *Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming*, Math. Oper. Res. 18(1) (1993), pp.202–226.

[11] P.P.B. Eggermont, *Multiplicative iterative algorithms for convex programming*, Linear Algebra Appl. 130 (1990), pp. 25–42.
[12] Y. He and R.D.C. Monteiro, *Accelerating block-decomposition first-order methods for solving composite saddle-point and two-player Nash equilibrium problems*, SIAM J. Optim. 25 (2015), pp. 2182–2211.
[13] Y. He and R.D.C. Monteiro, *An accelerated HPE-type algorithm for a class of composite convex–concave saddle-point problems*, SIAM J. Optim. 26 (2016), pp. 29–56.
[14] A.N. Iusem and M.V. Solodov, *Newton-type methods with generalized distances for constrained optimization*, Optimization 41 (1997), pp. 257–278.
[15] K.C. Kiwiel, *Proximal minimization methods with generalized Bregman functions*, SIAM J. Control Optim. 35 (1997), pp. 1142–1168.
[16] G.M. Korpelevič, *An extragradient method for finding saddle points and for other problems*, Èkonom. i Mat. Metody 12(4) (1976), pp. 747–756.
[17] R.D.C. Monteiro and B.F. Svaiter, *Convergence rate of inexact proximal point methods with relative error criteria for convex optimization*, Tech. Rep., 2010; available at http://www.optimization-online.org/DB_FILE/2010/08/2714.pdf.
[18] 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(6) (2010), pp. 2755–2787.
[19] 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(4) (2011), pp. 1688–1720.
[20] 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(1) (2013), pp. 475–507.
[21] 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(1) (2004), pp. 229–251.
[22] Y. Nesterov, *Smooth minimization of non-smooth functions*, Math. Program. 103(1) (2005), pp. 127–152.
[23] R.T. Rockafellar, *Convex Analysis*, Princeton Math. Series, vol. 28, 1970.
[24] R.T. Rockafellar, *On the maximal monotonicity of subdifferential mappings*, Pacific J. Math. 33 (1970), pp. 209–216.
[25] R.T. Rockafellar, *On the virtual convexity of the domain and range of a nonlinear maximal monotone operator*, Math. Annalen 185 (1970), pp. 81–90.
[26] R.T. Rockafellar and R.J.-B. Wets, *Variational Analysis*, Springer, Berlin, 1998.
[27] 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(4) (1999), pp. 323–345.
[28] M.V. Solodov and B.F. Svaiter, *A hybrid projection-proximal point algorithm*, J. Convex Anal. 6(1) (1999), pp. 59–70.
[29] 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(2) (2000), pp. 214–230.
[30] M.V. Solodov and B.F. Svaiter, *A unified framework for some inexact proximal point algorithms*, Numer. Funct. Anal. Optim. 22(7–8) (2001), pp. 1013–1035.
[31] P. Tseng, *A modified forward–backward splitting method for maximal monotone mappings*, SIAM J. Control Optim. 38(2) (2000), pp. 431–446.
[32] P. Tseng, *On accelerated proximal gradient methods for convex–concave optimization*, Tech. Rep., 2008; available at http://www.mit.edu/dimitrib/PTseng/papers/apgm.pdf.

Appendix

This appendix presents two existence/uniqueness results about solutions of certain regularized convex minimization and/or monotone inclusion problems.

We begin by stating without proof a well-known result about regularized convex minimization problems.

PROPOSITION A.1 *Let $\psi : \mathcal{Z} \to [-\infty, \infty]$ be a proper closed convex function and w be a distance generating function such that* $\operatorname{dom} \psi \cap \operatorname{int}(\operatorname{dom} w) \neq \emptyset$ *and w is strongly convex on* $\operatorname{dom} \psi$. *Then, for any* $z_- \in \operatorname{dom} \psi \cap \operatorname{int}(\operatorname{dom} w)$, *the problem*

$$\inf\{\psi(u) + (\mathrm{d}w)_{z_-}(u) : u \in \mathcal{Z}\}$$

has a unique optimal solution z, which necessarily lies in $\operatorname{dom} \psi \cap \operatorname{int}(\operatorname{dom} w)$. *Moreover, z is the unique zero of the inclusion* $\nabla w(z_-) \in (\partial\psi + \partial w)(z)$.

The next result gives a version of Proposition A.1 in the context of regularized monotone operators.

PROPOSITION A.2 *Let $T : \mathcal{Z} \rightrightarrows \mathcal{Z}$ be a maximal monotone operator and $w : \mathcal{Z} \to [-\infty, \infty]$ be a distance generating function such that* $\operatorname{int}(\operatorname{dom} w) \supset \operatorname{ri}(\operatorname{Dom} T)$ *and w is strongly convex on* $\operatorname{ri}(\operatorname{Dom} T)$. *Then, for every* $z_- \in \operatorname{int}(\operatorname{dom} w)$, *the*

inclusion $0 \in (T + \partial(\mathrm{d}w)_{z_-})(z)$ *has a unique solution* z*, which must necessarily be on* $\mathrm{Dom}\, T \cap \mathrm{int}(\mathrm{dom}\, w)$ *and hence satisfy the inclusion* $0 \in (T + \nabla(\mathrm{d}w)_{z_-})(z)$.

Proof Let $z_- \in \mathrm{int}(\mathrm{dom}\, w)$ be given. First note that

$$\mathrm{int}(\mathrm{dom}\, w) \subset \mathrm{Dom}(\partial w) = \mathrm{Dom}(\partial(\mathrm{d}w)_{z_-}) \subset \mathrm{dom}\, w,$$

from which we conclude that $\mathrm{ri}(\mathrm{Dom}(\partial(\mathrm{d}w)_{z_-}) = \mathrm{int}(\mathrm{dom}\, w)$. Moreover, by Proposition 2.40 and Theorem 12.41 of [26], we have that $\mathrm{ri}(\mathrm{Dom}\, T) \neq \emptyset$. These two observations then imply that

$$\mathrm{ri}(\mathrm{Dom}\, T) \cap \mathrm{ri}(\mathrm{Dom}(\partial(\mathrm{d}w)_{z_-}) = \mathrm{ri}(\mathrm{Dom}\, T) \cap \mathrm{int}(\mathrm{dom}\, w) = \mathrm{ri}(\mathrm{Dom}\, T) \neq \emptyset. \tag{A1}$$

Clearly, $(\mathrm{d}w)_{z_-}(\cdot)$ is a proper lower semicontinuous function due to Definition 3.1 and (20), and hence $\partial(\mathrm{d}w)_{z_-}$ is maximal monotone in view of Theorem 12.17 of [26]. Thus, it follows from (A1), the last conclusion, the assumption that T is maximal monotone and Corollary 12.44 of [26] that $T + \partial(\mathrm{d}w)_{z_-}$ is maximal monotone. Moreover, since w is strongly convex on $\mathrm{ri}(\mathrm{Dom}\, T)$, it follows that $\partial(\mathrm{d}w)_{z_-}$ is strongly monotone on $\mathrm{ri}(\mathrm{Dom}\, T)$. By using a simple limit argument and the fact that ∂w is a continuous map on $\mathrm{int}(\mathrm{dom}\, w)$ due to Definition 3.1, we conclude that $\partial(\mathrm{d}w)_{z_-}$ is strongly monotone on the larger set $\mathrm{Dom}\, T \cap \mathrm{int}(\mathrm{dom}\, w)$. Since the latter set is equal to $\mathrm{Dom}(T + \partial(\mathrm{d}w)_{z_-})$ and T is monotone, we conclude that $T + \partial(\mathrm{d}w)_{z_-}$ is strongly monotone. The first conclusion of the proposition now follows from Proposition 12.54 of [26]. The second conclusion follows immediately from the first one and the fact that, by Definition 3.1, $\mathrm{Dom}\,(\partial w) = \mathrm{int}(\mathrm{dom}\, w)$. ■