Math. Program., Ser. A (2016) 159:201–235
DOI 10.1007/s10107-015-0955-5

FULL LENGTH PAPER

Gradient sliding for composite optimization

Guanghui Lan[1]

Received: 11 June 2014 / Accepted: 1 October 2015 / Published online: 22 October 2015

Abstract We consider in this paper a class of composite optimization problems whose objective function is given by the summation of a general smooth and nonsmooth component, together with a relatively simple nonsmooth term. We present a new class of first-order methods, namely the gradient sliding algorithms, which can skip the computation of the gradient for the smooth component from time to time. As a consequence, these algorithms require only $\mathcal{O}(1/\sqrt{\epsilon})$ gradient evaluations for the smooth component in order to find an ϵ-solution for the composite problem, while still maintaining the optimal $\mathcal{O}(1/\epsilon^2)$ bound on the total number of subgradient evaluations for the nonsmooth component. We then present a stochastic counterpart for these algorithms and establish similar complexity bounds for solving an important class of stochastic composite optimization problems. Moreover, if the smooth component in the composite function is strongly convex, the developed gradient sliding algorithms can significantly reduce the number of graduate and subgradient evaluations for the smooth and nonsmooth component to $\mathcal{O}(\log(1/\epsilon))$ and $\mathcal{O}(1/\epsilon)$, respectively. Finally, we generalize these algorithms to the case when the smooth component is replaced by a nonsmooth one possessing a certain bi-linear saddle point structure.

Keywords Convex programming · Complexity · Gradient sliding · Nesterov's method · Data analysis

The author of this paper was partially supported by NSF CAREER Award CMMI-1254446, CMMI-1537414 NSF Grant DMS-1319050, and ONR Grant N00014-13-1-0036.

✉ Guanghui Lan
glan@ise.ufl.edu

[1] Department of Industrial and Systems Engineering, University of Florida,
Gainesville, FL 32611, USA

Mathematics Subject Classification 90C25 · 90C06 · 90C22 · 49M37

1 Introduction

In this paper, we consider a class of composite convex programming (CP) problems given in the form of

$$\Psi^* \equiv \min_{x \in X} \{\Psi(x) := f(x) + h(x) + \mathcal{X}(x)\}. \tag{1.1}$$

Here, $X \subseteq \mathbb{R}^n$ is a closed convex set, $\mathcal{X}$ is a relatively simple convex function, and $f : X \to \mathbb{R}$ and $h : X \to \mathbb{R}$, respectively, are general smooth and nonsmooth convex functions satisfying

$$f(x) \le f(y) + \langle \nabla f(y), x - y \rangle + \frac{L}{2}\|x - y\|^2, \quad \forall x, y \in X, \tag{1.2}$$

$$h(x) \le h(y) + \langle h'(y), x - y \rangle + M\|x - y\|, \quad \forall x, y \in X, \tag{1.3}$$

for some $L > 0$ and $M > 0$, where $h'(x) \in \partial h(x)$. Composite problem of this type appears in many data analysis applications, where either f or h corresponds to a certain data fidelity term, while the other components in Ψ denote regularization terms used to enforce certain structural properties for the obtained solutions.

Throughout this paper, we assume that one can access the first-order information of f and h separately. More specifically, in the deterministic setting, we can compute the exact gradient $\nabla f(x)$ and a subgradient $h'(x) \in \partial h(x)$ for any $x \in X$. We also consider the stochastic situation where only a stochastic subgradient of the nonsmooth component h is available. The main goal of this paper to provide a better theoretical understanding on how many number of gradient evaluations of ∇f and subgradient evaluations of h' are needed in order to find a certain approximate solution of (1.1).

Most existing first-order methods for solving (1.1) require the computation of both ∇f and h' in each iteration. In particular, since the objective function Ψ in (1.1) is nonsmooth, these algorithms would require $\mathcal{O}(1/\epsilon^2)$ first-order iterations, and hence $\mathcal{O}(1/\epsilon^2)$ evaluations for both ∇f and h' to find an ϵ-solution of (1.1), i.e., a point $\bar{x} \in X$ s.t. $\Psi(\bar{x}) - \Psi^* \le \epsilon$. Much recent research effort has been directed to reducing the impact of the Lipschitz constant L on the aforementioned complexity bounds for composite optimization. For example, Juditsky, Nemirovski and Travel showed in [8] that by using a variant of the mirror-prox method, the number of evaluations for ∇f and h' required to find an ϵ-solution of (1.1) can be bounded by

$$\mathcal{O}\left(\frac{L_f}{\epsilon} + \frac{M^2}{\epsilon^2}\right).$$

By developing an enhanced version of Nesterov's accelerated gradient method [15,16], Lan [11] further showed that the above bound can be improved to

$$\mathcal{O}\left(\sqrt{\frac{L_f}{\epsilon}}+\frac{M^2}{\epsilon^2}\right). \tag{1.4}$$

It is also shown in [11] that similar bounds hold for the stochastic case where only unbiased estimators for ∇f and h' are available. It is observed in [11] that such a complexity bound is not improvable if one can only access the first-order information for the summation of f and h all together.

Note, however, that it is unclear whether the complexity bound in (1.4) is optimal if one does have access to the first-order information of f and h separately. In particular, one would expect that the number of evaluations for ∇f can be bounded by $\mathcal{O}(1/\sqrt{\epsilon})$, if the nonsmooth term h in (1.1) does not appear (see [3,18,22]). However, it is unclear whether such a bound still holds for the more general composite problem in (1.1) without significantly increasing the bound in (1.4) on the number of subgradient evaluations for h'. It should be pointed out that in many applications the bottleneck of first-order methods exist in the computation of ∇f rather than that of h'. To motivate our study, let us mention a few such examples.

(a) In many inverse problems, we need to enforce certain block sparsity (e.g., total variation and overlapped group Lasso) by solving the problem of $\min_{x\in\mathbb{R}^n} \|Ax - b\|_2^2 + r(Bx)$. Here $A : \mathbb{R}^n \to \mathbb{R}^m$ is a given linear operator, $b \in \mathbb{R}^m$ denotes the collected observations, $r : \mathbb{R}^p \to \mathbb{R}$ is a relatively simple nonsmooth convex function (e.g., $r = \|\cdot\|_1$), and $B : \mathbb{R}^n \to \mathbb{R}^p$ is a very sparse matrix. In this case, evaluating the gradient of $\|Ax-b\|^2$ requires $\mathcal{O}(mn)$ arithmetic operations, while the computation of $r'(Bx)$ only needs $\mathcal{O}(n+p)$ arithmetic operations.

(b) In many machine learning problems, we need to minimize a regularized loss function given by $\min_{x\in\mathbb{R}^n} \mathbb{E}_\xi[l(x,\xi)] + q(Bx)$. Here $l : \mathbb{R}^n \times \mathbb{R}^d \to \mathbb{R}$ denotes a certain simple loss function, ξ is a random variable with unknown distribution, q is a certain smooth convex function, and $B : \mathbb{R}^n \to \mathbb{R}^p$ is a given linear operator. In this case, the computation of the stochastic subgradient for the loss function $\mathbb{E}_\xi[l(x,\xi)]$ requires only $\mathcal{O}(n+d)$ arithmetic operations, while evaluating the gradient of $q(Bx)$ needs $\mathcal{O}(np)$ arithmetic operations.

(c) In some cases, the computation of ∇f involves a black-box simulation procedure, the solution of an optimization problem, or a partial differential equation, while the computation of h' is given explicitly.

In all these cases mentioned above, it is desirable to reduce the number of gradient evaluations of ∇f to improve the overall efficiency for solving the composite problem (1.1).

Our contribution can be briefly summarized as follows. Firstly, we present a new class of first-order methods, namely the gradient sliding algorithms, and show that the number of gradient evaluations for ∇f required by these algorithms to find an ϵ-solution of (1.1) can be significantly reduced from (1.4) to

$$\mathcal{O}\left(\sqrt{\frac{L}{\epsilon}}\right), \tag{1.5}$$

while the total number of subgradient evaluations for h' is still bounded by (1.4). The basic scheme of these algorithms is to skip the computation of ∇f from time to time so that only $\mathcal{O}(1/\sqrt{\epsilon})$ gradient evaluations are needed in the $\mathcal{O}(1/\epsilon^2)$ iterations required to solve (1.1). Such an algorithmic framework originated from the simple idea of incorporating an iterative procedure to solve the subproblems in the aforementioned accelerated proximal gradient methods, although the analysis of these gradient sliding algorithms appears to be more technical and involved.

Secondly, we consider the stochastic case where the nonsmooth term h is represented by a stochastic oracle (SO), which, for a given search point $u_t \in X$, outputs a vector $H(u_t, \xi_t)$ such that (s.t.)

$$\mathbb{E}[H(u_t, \xi_t)] = h'(u_t) \in \partial h(u_t), \tag{1.6}$$

$$\mathbb{E}\left[\left\| H(u_t, \xi_t) - h'(u_t)\right\|_*^2\right] \le \sigma^2, \tag{1.7}$$

where ξ_t is a random vector independent of the search points u_t. Note that $H(u_t, \xi_t)$ is referred to as a stochastic subgradient of h at u_t and its computation is often much cheaper than the exact subgradient h'. Based on the gradient sliding techniques, we develop a new class of stochastic approximation type algorithms and show that the total number gradient evaluations of ∇f required by these algorithms to find a stochastic ϵ-solution of (1.1), i.e., a point $\bar{x} \in X$ s.t. $\mathbb{E}[\Psi(\bar{x}) - \Psi^*] \le \epsilon$, can still be bounded by (1.5), while the total number of stochastic subgradient evaluations can be bounded by

$$\mathcal{O}\left(\sqrt{\frac{L}{\epsilon}} + \frac{M^2 + \sigma^2}{\epsilon^2}\right).$$

We also establish large-deviation results associated with these complexity bounds under certain "light-tail" assumptions on the stochastic subgradients returned by the SO.

Thirdly, we generalize the gradient sliding algorithms for solving two important classes of composite problems given in the form of (1.1), but with f satisfying additional or alterative assumptions. We first assume that f is not only smooth, but also strongly convex, and show that the number of evaluations for ∇f and h' can be significantly reduced from $\mathcal{O}(1/\sqrt{\epsilon})$ and $\mathcal{O}(1/\epsilon^2)$, respectively, to $\mathcal{O}(\log(1/\epsilon))$ and $\mathcal{O}(1/\epsilon)$. We then consider the case when f is nonsmooth, but can be closely approximated by a class of smooth functions. By incorporating a novel smoothing scheme due to Nesterov [17] into the gradient sliding algorithms, we show that the number of gradient evaluations can be bounded by $\mathcal{O}(1/\epsilon)$, while the optimal $\mathcal{O}(1/\epsilon^2)$ bound on the number of subgradient evaluations of h' is still retained.

This paper is organized as follows. In Sect. 2.1, we provide some preliminaries on the prox-functions and a brief review on existing proximal gradient methods for solving (1.1). In Sect. 3, we present the gradient sliding algorithms and establish their convergence properties for solving problem (1.1). Section 4 is devoted to stochastic gradient sliding algorithms for solving a class of stochastic composite problems. In Sect. 5, we generalize the gradient sliding algorithms for the situation where f is smooth and strongly convex, and for the case when f is nonsmooth but can be closely approximated by a class of smooth functions. Finally, some concluding remarks are made in Sect. 6.

Notation and terminology We use $\|\cdot\|$ to denote an arbitrary norm in $\mathbb{R}^n$, which is not necessarily associated the inner product $\langle\cdot,\cdot\rangle$. We also use $\|\cdot\|_*$ to denote the conjugate norm of $\|\cdot\|$. For any $p \geq 1$, $\|\cdot\|_p$ denotes the standard p-norm in $\mathbb{R}^n$, i.e.,

$$\|x\|_p^p = \sum_{i=1}^{n} |x_i|^p, \qquad \text{for any } x \in \mathbb{R}^n.$$

For any convex function h, $\partial h(x)$ is the set of subdifferential at x. Given any $X \subseteq \mathbb{R}^n$, we say that $h : X \to \mathbb{R}$ is a general Lipschitz convex function if $|h(x) - h(y)| \leq M_h\|x - y\|$ for any $x, y \in X$. In this case, it can be shown that (1.3) holds with $M = 2M_h$ (see Lemma 2 of [11]). We say that a convex function $f : X \to \mathbb{R}$ is smooth if it is Lipschitz continuously differentiable with Lipschitz constant $L > 0$, i.e., $\|\nabla f(y) - \nabla f(x)\|_* \leq L\|y - x\|$ for any $x, y \in X$, which clearly implies (1.2).

For any real number r, $\lceil r \rceil$ and $\lfloor r \rfloor$ denote the nearest integer to r from above and below, respectively. $\mathbb{R}_+$ and $\mathbb{R}_{++}$, respectively, denote the set of nonnegative and positive real numbers. $\mathcal{N}$ denotes the set of natural numbers $\{1, 2, \ldots\}$.

2 Review of the proximal gradient methods

In this section, we provide a brief review on the proximal gradient methods from which the proposed gradient sliding algorithms originate, and point out a few problems associated with these existing algorithms when applied to solve problem (1.1).

2.1 Preliminary: distance generating function and prox-function

In this subsection, we review the concept of prox-function (i.e., proximity control function), which plays an important role in the recent development of first-order methods for convex programming. The goal of using the prox-function in place of the usual Euclidean distance is to allow the developed algorithms to get adapted to the geometry of the feasible sets.

We say that a function $\omega : X \to \mathbb{R}$ is a *distance generating function* with modulus $\nu > 0$ with respect to $\|\cdot\|$, if ω is continuously differentiable and strongly convex with parameter ν with respect to $\|\cdot\|$, i.e.,

$$\langle x - z, \nabla\omega(x) - \nabla\omega(z)\rangle \geq \nu\|x - z\|^2, \quad \forall x, z \in X. \tag{2.1}$$

The *prox-function* associated with ω is given by

$$V(x, z) \equiv V_\omega(x, z) = \omega(z) - [\omega(x) + \langle\nabla\omega(x), z - x\rangle]. \tag{2.2}$$

The prox-function $V(\cdot,\cdot)$ is also called the Bregman's distance, which was initially studied by Bregman [4] and later by many others (see [1,2,9] and references therein).

In this paper, we assume that the prox-function $V(x, z)$ is chosen such that the solution of

$$\arg\min_{u \in X} \{\langle g, u\rangle + V(x, u) + \mathcal{X}(u)\} \tag{2.3}$$

is easily computable for any $g \in \mathcal{E}^*$ and $x \in X$. Some examples of these prox-functions are given in [5].

If there exists a constant $\mathcal{Q}$ such that $V(x, z) \le \mathcal{Q}\|x-z\|^2/2$ for any $x, z \in X$, then we say that the prox-function $V(\cdot, \cdot)$ is growing quadratically. Moreover, the smallest constant $\mathcal{Q}$ satisfying the previous relation is called the *quadratic growth constant* of $V(\cdot, \cdot)$. Without loss of generality, we assume that $\mathcal{Q} = 1$ for the prox-function $V(x, z)$ if it grows quadratically, i.e.,

$$V(x, z) \le \frac{1}{2}\|x - z\|^2, \quad \forall x, z \in X. \tag{2.4}$$

Indeed, if $\mathcal{Q} \neq 1$, we can multiply the corresponding distance generating function ω by $1/\mathcal{Q}$ and the resulting prox-function will satisfy (2.4).

2.2 Proximal gradient methods

In this subsection, we briefly review a few possible first-order methods for solving problem (1.1).

We start with the simplest proximal gradient method which works for the case when the nonsmooth component h does not appear or is relatively simple (e.g., h is affine). For a given $x \in X$, let

$$m_\Psi(x, u) := l_f(x, u) + h(u) + \mathcal{X}(u), \quad \forall u \in X, \tag{2.5}$$

where

$$l_f(x; y) := f(x) + \langle \nabla f(x), y - x\rangle. \tag{2.6}$$

Clearly, by the convexity of f and (1.2), we have

$$m_\Psi(x, u) \le \Psi(u) \le m_\Psi(x, u) + \frac{L}{2}\|u - x\|^2 \le m_\Psi(x, u) + \frac{L}{\nu}V(x, u)$$

for any $u \in X$, where the last inequality follows from the strong convexity of ω. Hence, $m_\Psi(x, u)$ is a good approximation of $\Psi(u)$ when u is "close" enough to x. In view of this observation, we update the search point $x_k \in X$ at the k-th iteration of the proximal gradient method by

$$x_k = \operatorname*{argmin}_{u \in X} \left\{l_f(x_{k-1}, u) + h(u) + \mathcal{X}(u) + \beta_k V(x_{k-1}, u)\right\}, \tag{2.7}$$

Here, $\beta_k > 0$ is a parameter which determines how well we "trust" the proximity between $m_\Psi(x_{k-1}, u)$ and $\Psi(u)$. In particular, a larger value of β_k implies less confidence on $m_\Psi(x_{k-1}, u)$ and results in a smaller step moving from x_{k-1} to x_k. It can be shown that the number of iterations required by the proximal gradient method for finding an ϵ-solution of (1.1) can be bounded by $\mathcal{O}(1/\epsilon)$ (see, e.g., Theorem 2.1.14 of [16]).

The efficiency of the above proximal gradient method can be significantly improved by incorporating a multi-step acceleration scheme. The basic idea of this scheme is to introduce three closely related search sequences, namely, $\{\underline{x}_k\}$, $\{x_k\}$, and $\{\bar{x}_k\}$, which will be used to build the model m_Ψ, control the proximity between m_Ψ and Ψ, and compute the output solution, respectively. More specifically, these three sequences are updated according to

$$\underline{x}_k = (1-\gamma_k)\bar{x}_{k-1} + \gamma_k x_{k-1}, \tag{2.8}$$

$$x_k = \operatorname*{argmin}_{u \in X} \left\{ \Phi_k(u) := l_f(\underline{x}_k, u) + h(u) + \mathcal{X}(u) + \beta_k V(x_{k-1}, u) \right\}, \tag{2.9}$$

$$\bar{x}_k = (1-\gamma_k)\bar{x}_{k-1} + \gamma_k x_k, \tag{2.10}$$

where $\beta_k \geq 0$ and $\gamma_k \in [0, 1]$ are given parameters for the algorithm. Clearly, (2.8)–(2.10) reduces to (2.7), if $\bar{x}_0 = x_0$ and γ_k is set to 1. However, by properly specifying β_k and γ_k, e.g., $\beta_k = 2L/k$ and $\gamma_k = 2/(k+2)$, one can show that the above accelerated proximal gradient method can find an ϵ-solution of (1.1) in at most $\mathcal{O}(1/\sqrt{\epsilon})$ iterations [see [22] for the analysis of the scheme in (2.8)–(2.10)]. Since each iteration of this algorithm requires only one evaluation of ∇f, the total number of gradient evaluations of ∇f can also be bounded by $\mathcal{O}(1/\sqrt{\epsilon})$.

One crucial problem associated with the aforementioned proximal gradient type methods is that the subproblems (2.7) and (2.9) are difficult to solve when h is a general nonsmooth convex function. To address this issue, one can possibly apply an enhanced accelerated gradient method by Lan [11] (see also [5,6]). This algorithm is obtained by replacing $h(u)$ in (2.9) with

$$l_h(\underline{x}_k; u) := h(\underline{x}_k) + \langle h'(\underline{x}_k), u - \underline{x}_k \rangle \tag{2.11}$$

for some $h'(\underline{x}_k) \in \partial h(\underline{x}_k)$. As a result, the subproblems in this algorithm become easier to solve. Moreover, with a proper selection of $\{\beta_k\}$ and $\{\gamma_k\}$, this approach can find an ϵ-solution of (1.1) in at most

$$\mathcal{O}\left\{ \sqrt{\frac{LV(x_0, x^*)}{\epsilon}} + \frac{M^2 V(x_0, x^*)}{\epsilon^2} \right\} \tag{2.12}$$

iterations. Since each iteration requires one computation of ∇f and h', the total number of evaluations for f and h' is bounded by $\mathcal{O}(1/\epsilon^2)$. As pointed out in [11], this bound in (2.12) is not improvable if one can only compute the subgradient of the composite function $f(x) + h(x)$ as a whole. However, as noted in Sect. 1, we do have access to separate first-order information about f and h in many applications. One interesting problem is whether we can further improve the performance of proximal gradient type methods in the latter case.

3 Deterministic gradient sliding

Throughout this section, we consider the deterministic case where exact subgradients of h are available. By presenting a new class of proximal gradient methods, namely the gradient sliding (GS) method, we show that one can significantly reduce the number of gradient evaluations for ∇f required to solve (1.1), while maintaining the optimal bound on the total number of subgradient evaluations for h'.

The basic idea of the GS method is to incorporate an iterative procedure to approximately solve the subproblem (2.9) in the accelerated proximal gradient methods. A critical observation in our development of the GS method is that one needs to compute a pair of closely related approximate solutions of problem (2.9). One of them will be used in place of x_k in (2.8) to construct the model m_Ψ, while the other one will be used in place of x_k in (2.10) to compute the output solution $\bar{x}_k$. Moreover, we show that such a pair of approximation solutions can be obtained by applying a simple subgradient projection type subroutine. We now formally describe this algorithm as follows.

Algorithm 1 The gradient sliding (GS) algorithm

```
**Input:** Initial point $x_0 \in X$ and iteration limit $N$.
Let $\beta_k \in \mathbb{R}_{++}$, $\gamma_k \in \mathbb{R}_+$, and $T_k \in \mathcal{N}$, $k = 1, 2, \ldots$, be given and set $\bar{x}_0 = x_0$.
**for** $k = 1, 2, \ldots, N$ **do**
    1. Set $\underline{x}_k = (1-\gamma_k)\bar{x}_{k-1} + \gamma_k x_{k-1}$, and let $g_k(\cdot) \equiv l_f(\underline{x}_k, \cdot)$ be defined in (2.6).
    2. Set
        $(x_k, \tilde{x}_k) = \mathrm{PS}(g_k, x_{k-1}, \beta_k, T_k);$        (3.1)
    3. Set $\bar{x}_k = (1-\gamma_k)\bar{x}_{k-1} + \gamma_k \tilde{x}_k$.
**end for**
**Output:** $\bar{x}_N$.

    The PS (prox-sliding) procedure called at step 2 is stated as follows.
**procedure** $(x^+, \tilde{x}^+) = \mathrm{PS}(g, x, \beta, T)$
    Let the parameters $p_t \in \mathbb{R}_{++}$ and $\theta_t \in [0,1]$, $t = 1, \ldots$, be given. Set $u_0 = \tilde{u}_0 = x$.
    **for** $t = 1, 2, \ldots, T$ **do**
        $u_t = \mathrm{argmin}_{u \in X} \left\{ g(u) + l_h(u_{t-1}, u) + \beta V(x, u) + \beta p_t V(u_{t-1}, u) + \mathcal{X}(u) \right\},$        (3.2)
        $\tilde{u}_t = (1-\theta_t)\tilde{u}_{t-1} + \theta_t u_t.$        (3.3)
    **end for**
    Set $x^+ = u_T$ and $\tilde{x}^+ = \tilde{u}_T$.
**end procedure**
```

Observe that when supplied with an affine function $g(\cdot)$, prox-center $x \in X$, parameter β, and sliding period T, the PS procedure computes a pair of approximate solutions $(x^+, \tilde{x}^+) \in X \times X$ for the problem of:

$$\underset{u \in X}{\mathrm{argmin}} \left\{ \Phi(u) := g(u) + h(u) + \beta V(x, u) + \mathcal{X}(u) \right\}. \tag{3.4}$$

Clearly, problem (3.4) is equivalent to (2.9) when the input parameters are set to (3.1). Since the same affine function $g(\cdot) = l_f(\underline{x}_{k-1}, \cdot)$ has been used throughout the T iterations of the PS procedure, we skip the computation of the gradients of f when performing the T projection steps in (3.2). This differs from the accelerated gradient method in [11], where one needs to compute $\nabla f + h'$ in each projection step.

It should also be noted that there has been some related work on the accelerated gradient methods with inexact solution of the proximal mapping step (3.4) (see, e.g., [19,23]). The results basically state that the approximation error at each step has to decrease very fast to maintain the accelerated convergence rate. Since (3.4) is strongly convex, one can apply the subgradient method to solve it efficiently. However, one needs to carefully deal with some difficulties in this intuitive approach. Firstly, one has to define an appropriate termination criterion for solving (3.4). It turns out that using the natural functional optimality gap as the termination criterion for this subproblem could not lead to the desirable convergence rates, and we need to use in the GS algorithm a special termination criterion defined by the summation of the functional optimality gap and the distance to the optimal solution [see (3.9) below]. Secondly, even though (3.4) is strongly convex, it is nonsmooth and the strong convexity modulus decreases as the number of iterations increases. Hence, one has to carefully determine the specification of these nested (accelerated) subgradient algorithms. Thirdly, one important modification that we incorporated in the GS method is to use two different approximate solutions in the two interpolation updates in the accelerated gradient methods. Otherwise, one could not obtain the optimal complexity bounds on the computation of both ∇f and h'.

A few more remarks about the above GS algorithm are in order. Firstly, we say that an outer iteration of the GS algorithm occurs whenever k in Algorithm 1 increments by 1. Each outer iteration of the GS algorithm involves the computation of the gradient $\nabla f(\underline{x}_{k-1})$ and a call to the PS procedure to update x_k and $\tilde{x}_k$. Secondly, the PS procedure solves problem (3.4) iteratively. Each iteration of this procedure consists of the computation of subgradient $h'(u_{t-1})$ and the solution of the projection subproblem (3.2), which is assumed to be relatively easy to solve (see Sect. 2.1). For notational convenience, we refer to an iteration of the PS procedure as an inner iteration of the GS algorithm. Thirdly, the GS algorithm described above is conceptual only since we have not yet specified the selection of $\{\beta_k\}$, $\{\gamma_k\}$, $\{T_k\}$, $\{p_t\}$ and $\{\theta_t\}$. We will return to this issue after establishing some convergence properties of the generic GS algorithm described above.

We first present a result which summarizes some important convergence properties of the PS procedure. The following two technical results are needed to establish the convergence of this procedure.

The first technical result below characterizes the solution of the projection step (3.1). The proof of this result can be found in Lemma 2 of [5].

Lemma 1 *Let the convex function $q : X \to \mathbb{R}$, the points $\tilde{x}, \tilde{y} \in X$ and the scalars $\mu_1, \mu_2 \in \mathbb{R}_+$ be given. Let $\omega : X \to \mathbb{R}$ be a differentiable convex function and $V(x, z)$ be defined in (2.2). If*

$$u^* \in \mathrm{Argmin}\{q(u) + \mu_1 V(\tilde{x}, u) + \mu_2 V(\tilde{y}, u) : u \in X\},$$

then for any $u \in X$, we have

$$q(u^*) + \mu_1 V(\tilde{x}, u^*) + \mu_2 V(\tilde{y}, u^*) \le q(u) + \mu_1 V(\tilde{x}, u) + \mu_2 V(\tilde{y}, u) \\ - (\mu_1 + \mu_2) V(u^*, u).$$

The second technical result slightly generalizes Lemma 3 of [12] to provide a convenient way to analyze sequences with sublinear rate of convergence.

Lemma 2 *Let $w_k \in (0, 1]$, $k = 1, 2, \ldots$, and $W_1 > 0$ be given and define*

$$W_k := (1 - w_k) W_{k-1}, \quad k \ge 2. \tag{3.5}$$

Suppose that $W_k > 0$ for all $k \ge 2$ and that the sequence $\{\delta_k\}_{k \ge 0}$ satisfies

$$\delta_k \le (1 - w_k)\delta_{k-1} + B_k, \quad k = 1, 2, \ldots. \tag{3.6}$$

Then for any $k \ge 1$, we have

$$\delta_k \le W_k \left[\frac{1 - w_1}{W_1} \delta_0 + \sum_{i=1}^{k} \frac{B_i}{W_i} \right]. \tag{3.7}$$

Proof The result follows from dividing both sides of (3.6) by W_k and then summing up the resulting inequalities. □

We are now ready to establish the convergence of the PS procedure.

Proposition 1 *If $\{p_t\}$ and $\{\theta_t\}$ in the* PS *procedure satisfy*

$$\theta_t = \frac{P_{t-1} - P_t}{(1 - P_t) P_{t-1}} \quad \text{with} \quad P_t := \begin{cases} 1, & t = 0, \\ p_t (1 + p_t)^{-1} P_{t-1}, & t \ge 1, \end{cases} \tag{3.8}$$

then, for any $t \ge 1$ and $u \in X$,

$$\beta (1 - P_t)^{-1} V(u_t, u) + [\Phi(\tilde{u}_t) - \Phi(u)] \\ \le P_t (1 - P_t)^{-1} \left[\beta V(u_0, u) + \frac{M^2}{2\nu\beta} \sum_{i=1}^{t} (p_i^2 P_{i-1})^{-1} \right], \tag{3.9}$$

where Φ is defined in (3.4).

Proof By (1.3) and the definition of l_h in (2.11), we have $h(u_t) \le l_h(u_{t-1}, u_t) + M\|u_t - u_{t-1}\|$. Adding $g(u_t) + \beta V(x, u_t) + \mathcal{X}(u_t)$ to both sides of this inequality and using the definition of Φ in (3.4), we obtain

$$\Phi(u_t) \le g(u_t) + l_h(u_{t-1}, u_t) + \beta V(x, u_t) + \mathcal{X}(u_t) + M\|u_t - u_{t-1}\|. \tag{3.10}$$

Now applying Lemma 1 to (3.2), we obtain

$$\begin{aligned}
&g(u_t) + l_h(u_{t-1}, u_t) + \beta V(x, u_t) + \mathcal{X}(u_t) + \beta p_t V(u_{t-1}, u_t)\\
&\quad \le g(u) + l_h(u_{t-1}, u) + \beta V(x, u) + \mathcal{X}(u) + \beta p_t V(u_{t-1}, u) - \beta(1+p_t)V(u_t, u)\\
&\quad \le g(u) + h(u) + \beta V(x, u) + \mathcal{X}(u) + \beta p_t V(u_{t-1}, u) - \beta(1+p_t)V(u_t, u)\\
&\quad = \Phi(u) + \beta p_t V(u_{t-1}, u) - \beta(1+p_t)V(u_t, u),
\end{aligned}$$

where the second inequality follows from the convexity of h. Moreover, by the strong convexity of ω,

$$\begin{aligned}
-\beta p_t V(u_{t-1}, u_t) + M\|u_t - u_{t-1}\| &\le -\frac{\nu\beta p_t}{2}\|u_t - u_{t-1}\|^2 + M\|u_t - u_{t-1}\|\\
&\le \frac{M^2}{2\nu\beta p_t},
\end{aligned}$$

where the last inequality follows from the simple fact that $-at^2/2 + bt \le b^2/(2a)$ for any $a > 0$. Combining the previous three inequalities, we conclude that

$$\Phi(u_t) - \Phi(u) \le \beta p_t V(u_{t-1}, u) - \beta(1+p_t)V(u_t, u) + \frac{M^2}{2\nu\beta p_t}.$$

Dividing both sides by $1 + p_t$ and rearranging the terms, we obtain

$$\beta V(u_t, u) + \frac{\Phi(u_t) - \Phi(u)}{1+p_t} \le \frac{\beta p_t}{1+p_t} V(u_{t-1}, u) + \frac{M^2}{2\nu\beta(1+p_t)p_t},$$

which, in view of the definition of P_t in (3.8) and Lemma 2 (with $k = t$, $w_k = 1/(1+p_t)$ and $W_k = P_t$), then implies that

$$\begin{aligned}
\frac{\beta}{P_t} V(u_t, u) + \sum_{i=1}^{t} \frac{\Phi(u_i) - \Phi(u)}{P_i(1+p_i)} &\le \beta V(u_0, u) + \frac{M^2}{2\nu\beta} \sum_{i=1}^{t} \frac{1}{P_i(1+p_i)p_i}\\
&= \beta V(u_0, u) + \frac{M^2}{2\nu\beta} \sum_{i=1}^{t} (p_i^2 P_{i-1})^{-1}, \qquad (3.11)
\end{aligned}$$

where the last identity also follows from the definition of P_t in (3.8). Also note that by the definition of $\tilde{u}_t$ in the PS procedure and (3.8), we have

$$\tilde{u}_t = \frac{P_t}{1 - P_t}\left(\frac{1 - P_{t-1}}{P_{t-1}}\tilde{u}_{t-1} + \frac{1}{P_t(1+p_t)} u_t\right).$$

Applying this relation inductively and using the fact that $P_0 = 1$, we can easily see that

$$\begin{aligned}\tilde{u}_t &= \frac{P_t}{1-P_t}\left[\frac{1-P_{t-2}}{P_{t-2}}\tilde{u}_{t-2} + \frac{1}{P_{t-1}(1+p_{t-1})}u_{t-1} + \frac{1}{P_t(1+p_t)}u_t\right]\\ &= \cdots = \frac{P_t}{1-P_t}\sum_{i=1}^{t}\frac{1}{P_i(1+p_i)}u_i,\end{aligned}$$

which, in view of the convexity of Φ, then implies that

$$\Phi(\tilde{u}_t) - \Phi(u) \le \frac{P_t}{1-P_t}\sum_{i=1}^{t}\frac{\Phi(u_i)-\Phi(u)}{P_i(1+p_i)}. \tag{3.12}$$

Combining the above inequality with (3.11) and rearranging the terms, we obtain (3.9). □

Setting u to be the optimal solution of (3.4), we can see that both x_k and $\tilde{x}_k$ are approximate solutions of (3.4) if the right hand side (RHS) of (3.9) is small enough. With the help of this result, we can establish an important recursion from which the convergence of the GS algorithm easily follows.

Proposition 2 *Suppose that $\{p_t\}$ and $\{\theta_t\}$ in the* PS *procedure satisfy* (3.8). *Also assume that $\{\beta_k\}$ and $\{\gamma_k\}$ in the GS algorithm satisfy*

$$\gamma_1 = 1 \quad \textit{and} \quad \nu\beta_k - L\gamma_k \ge 0, \quad k \ge 1. \tag{3.13}$$

Then for any $u \in X$ and $k \ge 1$,

$$\begin{aligned}\Psi(\bar{x}_k) - \Psi(u) \le& (1-\gamma_k)[\Psi(\bar{x}_{k-1}) - \Psi(u)] + \gamma_k(1-P_{T_k})^{-1}\\ &\left[\beta_k V(x_{k-1},u) - \beta_k V(x_k,u) + \frac{M^2 P_{T_k}}{2\nu\beta_k}\sum_{i=1}^{T_k}(p_i^2 P_{i-1})^{-1}\right].\end{aligned} \tag{3.14}$$

Proof First, notice that by the definition of $\bar{x}_k$ and $\underline{x}_k$, we have $\bar{x}_k - \underline{x}_k = \gamma_k(\tilde{x}_k - x_{k-1})$. Using this observation, (1.2), the definition of l_f in (2.6), and the convexity of f, we obtain

$$\begin{aligned}f(\bar{x}_k) &\le l_f(\underline{x}_k, \bar{x}_k) + \frac{L}{2}\|\bar{x}_k - \underline{x}_k\|^2\\ &= (1-\gamma_k)l_f(\underline{x}_k, \bar{x}_{k-1}) + \gamma_k l_f(\underline{x}_k, \tilde{x}_k) + \frac{L\gamma_k^2}{2}\|\tilde{x}_k - x_{k-1}\|^2\\ &\le (1-\gamma_k)f(\bar{x}_{k-1}) + \gamma_k\left[l_f(\underline{x}_k, \tilde{x}_k) + \beta_k V(x_{k-1}, \tilde{x}_k)\right]\\ &\quad - \gamma_k\beta_k V(x_{k-1}, \tilde{x}_k) + \frac{L\gamma_k^2}{2}\|\tilde{x}_k - x_{k-1}\|^2\end{aligned}$$

$$\begin{aligned}&\le (1-\gamma_k) f(\bar{x}_{k-1}) + \gamma_k \left[l_f(\underline{x}_k, \tilde{x}_k) + \beta_k V(x_{k-1}, \tilde{x}_k) \right] \\ &\quad - \left(\gamma_k \beta_k - \frac{L\gamma_k^2}{\nu} \right) V(x_{k-1}, \tilde{x}_k) \\ &\le (1-\gamma_k) f(\bar{x}_{k-1}) + \gamma_k \left[l_f(\underline{x}_k, \tilde{x}_k) + \beta_k V(x_{k-1}, \tilde{x}_k) \right], \qquad (3.15)\end{aligned}$$

where the third inequality follows from the strong convexity of ω and the last inequality follows from (3.13). By the convexity of h and $\mathcal{X}$, we have

$$h(\bar{x}_k) + \mathcal{X}(\bar{x}_k) \le (1-\gamma_k)[h(\bar{x}_{k-1}) + \mathcal{X}(\bar{x}_{k-1})] + \gamma_k [h(\tilde{x}_k) + \mathcal{X}(\tilde{x}_k)]. \qquad (3.16)$$

Adding up the previous two inequalities, and using the definitions of Ψ in (1.1) and Φ_k in (2.9), we have

$$\Psi(\bar{x}_k) \le (1-\gamma_k)\Psi(\bar{x}_{k-1}) + \gamma_k \Phi_k(\tilde{x}_k).$$

Subtracting $\Psi(u)$ from both sides of the above inequality, we obtain

$$\Psi(\bar{x}_k) - \Psi(u) \le (1-\gamma_k)[\Psi(\bar{x}_{k-1}) - \Psi(u)] + \gamma_k [\Phi_k(\tilde{x}_k) - \Psi(u)]. \qquad (3.17)$$

Also note that by the definition of Φ_k in (2.9) and the convexity of f,

$$\Phi_k(u) \le f(u) + h(u) + \mathcal{X}(u) + \beta_k V(x_{k-1}, u) = \Psi(u) + \beta_k V(x_{k-1}, u), \quad \forall u \in X. \qquad (3.18)$$

Combining these two inequalities (i.e., replacing the third $\Psi(u)$ in (3.17) by $\phi_k(u) - \beta_k V(x_{k-1}, u)$), we obtain

$$\begin{aligned}\Psi(\bar{x}_k) - \Psi(u) &\le (1-\gamma_k)[\Psi(\bar{x}_{k-1}) - \Psi(u)] \\ &\quad + \gamma_k [\Phi_k(\tilde{x}_k) - \Phi_k(u) + \beta_k V(x_{k-1}, u)]. \qquad (3.19)\end{aligned}$$

Now, in view of the definition of Φ_k in (2.9) and the origin of $(x_k, \tilde{x}_k)$ in (3.1), we can apply Proposition 1 with $\phi = \phi_k$, $u_0 = x_{k-1}$, $u_t = x_k$, $\tilde{u}_t = \tilde{x}_k$, and $\beta = \beta_k$, and conclude that for any $u \in X$ and $k \ge 1$,

$$\begin{aligned}&\frac{\beta_k}{1-P_{T_k}} V(x_k, u) + [\Phi_k(\tilde{x}_k) - \Phi_k(u)] \\ &\quad \le \frac{P_{T_k}}{1-P_{T_k}} \left[\beta_k V(x_{k-1}, u) + \frac{M^2}{2\nu\beta_k} \sum_{i=1}^{T_k} (p_i^2 P_{i-1})^{-1} \right].\end{aligned}$$

Plugging the above bound on $\Phi_k(\tilde{x}_k) - \Phi_k(u)$ into (3.19), we obtain (3.14). □

We are now ready to establish the main convergence properties of the GS algorithm. Note that the following quantity will be used in our analysis of this algorithm.

$$\Gamma_k = \begin{cases} 1, & k = 1, \\ (1-\gamma_k)\Gamma_{k-1}, & k \ge 2. \end{cases} \tag{3.20}$$

Theorem 1 *Assume that $\{p_t\}$ and $\{\theta_t\}$ in the* PS *procedure satisfy* (3.8)*, and also that $\{\beta_k\}$ and $\{\gamma_k\}$ in the GS algorithm satisfy* (3.13)*.*

(a) *If for any $k \ge 2$,*

$$\frac{\gamma_k \beta_k}{\Gamma_k(1-P_{T_k})} \le \frac{\gamma_{k-1}\beta_{k-1}}{\Gamma_{k-1}(1-P_{T_{k-1}})}, \tag{3.21}$$

then we have, for any $N \ge 1$,

$$\begin{aligned} \Psi(\bar{x}_N) - \Psi(x^*) \le \mathcal{B}_d(N) := \frac{\Gamma_N \beta_1}{1-P_{T_1}} V(x_0, x^*) \\ + \frac{M^2 \Gamma_N}{2\nu} \sum_{k=1}^{N} \sum_{i=1}^{T_k} \frac{\gamma_k P_{T_k}}{\Gamma_k \beta_k (1-P_{T_k}) p_i^2 P_{i-1}}, \end{aligned} \tag{3.22}$$

where $x^* \in X$ is an arbitrary optimal solution of problem (1.1)*, and P_t and Γ_k are defined in* (3.8) *and* (3.20)*, respectively.*

(b) *If X is compact, and for any $k \ge 2$,*

$$\frac{\gamma_k \beta_k}{\Gamma_k(1-P_{T_k})} \ge \frac{\gamma_{k-1}\beta_{k-1}}{\Gamma_{k-1}(1-P_{T_{k-1}})}, \tag{3.23}$$

then (3.22) *still holds by simply replacing the first term in the definition of $\mathcal{B}_d(N)$ with $\gamma_N \beta_N \bar{V}(x^*)/(1-P_{T_N})$, where $\bar{V}(u) = \max_{x \in X} V(x,u)$.*

Proof We conclude from (3.14) and Lemma 2 that

$$\begin{aligned} \Psi(\bar{x}_N) - \Psi(u) &\le \Gamma_N \frac{1-\gamma_1}{\Gamma_1}[\Psi(\bar{x}_0) - \Psi(u)] \\ &\quad + \Gamma_N \sum_{k=1}^{N} \frac{\beta_k \gamma_k}{\Gamma_k(1-P_{T_k})} \left[V(x_{k-1}, u) - V(x_k, u)\right] \\ &\quad + \frac{M^2 \Gamma_N}{2\nu} \sum_{k=1}^{N} \sum_{i=1}^{T_k} \frac{\gamma_k P_{T_k}}{\Gamma_k \beta_k (1-P_{T_k}) p_i^2 P_{i-1}} \\ &= \Gamma_N \sum_{k=1}^{N} \frac{\beta_k \gamma_k}{\Gamma_k(1-P_{T_k})} \left[V(x_{k-1}, u) - V(x_k, u)\right] \\ &\quad + \frac{M^2 \Gamma_N}{2\nu} \sum_{k=1}^{N} \sum_{i=1}^{T_k} \frac{\gamma_k P_{T_k}}{\Gamma_k \beta_k (1-P_{T_k}) p_i^2 P_{i-1}}, \end{aligned} \tag{3.24}$$

where the last identity follows from the fact that $\gamma_1 = 1$. Now it follows from (3.21) that

$$\begin{aligned}\sum_{k=1}^{N} & \frac{\beta_k \gamma_k}{\Gamma_k(1-P_{T_k})}\left[V(x_{k-1},u) - V(x_k,u)\right] \\ & \le \frac{\beta_1\gamma_1}{\Gamma_1(1-P_{T_1})}V(x_0,u) - \frac{\beta_N\gamma_N}{\Gamma_N(1-P_{T_N})}V(x_N,u) \le \frac{\beta_1}{1-P_{T_1}}V(x_0,u),\end{aligned} \tag{3.25}$$

where the last inequality follows from the facts that $\gamma_1 = \Gamma_1 = 1$, $P_{T_N} \le 1$, and $V(x_N,u) \ge 0$. The result in part a) then clearly follows from the previous two inequalities with $u = x^*$. Moreover, using (3.23) and the fact $V(x_k,u) \le \bar{V}(u)$, we conclude that

$$\begin{aligned}\sum_{k=1}^{N} & \frac{\beta_k \gamma_k}{\Gamma_k(1-P_{T_k})}\left[V(x_{k-1},u) - V(x_k,u)\right] \\ & \le \frac{\beta_1}{1-P_{T_1}}\bar{V}(u) - \sum_{k=2}^{N}\left[\frac{\beta_{k-1}\gamma_{k-1}}{\Gamma_{k-1}(1-P_{T_{k-1}})} - \frac{\beta_k\gamma_k}{\Gamma_k(1-P_{T_k})}\right]\bar{V}(u) \\ & = \frac{\gamma_N\beta_N}{\Gamma_N(1-P_{T_N})}\bar{V}(u).\end{aligned} \tag{3.26}$$

Part b) then follows from the above observation and (3.24) with $u = x^*$. □

Clearly, there are various options for specifying the parameters $\{p_t\}$, $\{\theta_t\}$, $\{\beta_k\}$, $\{\gamma_k\}$, and $\{T_k\}$ to guarantee the convergence of the GS algorithm. Below we provide a few such selections which lead to the best possible rate of convergence for solving problem (1.1). In particular, Corollary 1(a) provides a set of such parameters for the case when the feasible region X is unbounded and the iteration limit N is given a priori, while the one in Corollary 1(b) works only for the case when X is compact, but does not require N to be given in advance.

Corollary 1 *Assume that $\{p_t\}$ and $\{\theta_t\}$ in the* PS *procedure are set to*

$$p_t = \frac{t}{2} \quad and \quad \theta_t = \frac{2(t+1)}{t(t+3)}, \quad \forall t \ge 1. \tag{3.27}$$

(a) *If N is fixed a priori, and $\{\beta_k\}$, $\{\gamma_k\}$, and $\{T_k\}$ are set to*

$$\beta_k = \frac{2L}{vk}, \quad \gamma_k = \frac{2}{k+1}, \quad and \quad T_k = \left\lceil \frac{M^2Nk^2}{\tilde{D}L^2} \right\rceil \tag{3.28}$$

for some $\tilde{D} > 0$, then

$$\Psi(\bar{x}_N) - \Psi(x^*) \le \frac{2L}{N(N+1)}\left[\frac{3V(x_0,x^*)}{\nu} + 2\tilde{D}\right], \quad \forall N \ge 1. \tag{3.29}$$

(b) *If X is compact, and $\{\beta_k\}$, $\{\gamma_k\}$, and $\{T_k\}$ are set to*

$$\beta_k = \frac{9L(1-P_{T_k})}{2\nu(k+1)}, \quad \gamma_k = \frac{3}{k+2}, \quad and \quad T_k = \left\lceil \frac{M^2(k+1)^3}{\tilde{D}L^2} \right\rceil, \tag{3.30}$$

for some $\tilde{D} > 0$, then

$$\Psi(\bar{x}_N) - \Psi(x^*) \le \frac{L}{(N+1)(N+2)} \left(\frac{27\bar{V}(x^*)}{2\nu} + \frac{8\tilde{D}}{3} \right), \quad \forall N \ge 1. \tag{3.31}$$

Proof We first show part a). By the definitions of P_t and p_t in (3.8) and (3.27), we have

$$P_t = \frac{tP_{t-1}}{t+2} = \cdots = \frac{2}{(t+1)(t+2)}. \tag{3.32}$$

Using the above identity and (3.27), we can easily see that the condition in (3.8) holds. It also follows from (3.32) and the definition of T_k in (3.28) that

$$P_{T_k} \le P_{T_{k-1}} \le \cdots \le P_{T_1} \le \frac{1}{3}. \tag{3.33}$$

Now, it can be easily seen from the definition of β_k and γ_k in (3.28) that (3.13) holds. It also follows from (3.20) and (3.28) that

$$\Gamma_k = \frac{2}{k(k+1)}. \tag{3.34}$$

By (3.28), (3.33), and (3.34), we have

$$\frac{\gamma_k\beta_k}{\Gamma_k(1-P_{T_k})} = \frac{2L}{\nu(1-P_{T_k})} \le \frac{2L}{\nu(1-P_{T_{k-1}})} = \frac{\gamma_{k-1}\beta_{k-1}}{\Gamma_{k-1}(1-P_{T_k-1})},$$

from which (3.21) follows. Now, by (3.32) and the fact that $p_t = t/2$, we have

$$\sum_{i=1}^{T_k} \frac{1}{p_i^2 P_{i-1}} = 2\sum_{i=1}^{T_k} \frac{i+1}{i} \le 4T_k, \tag{3.35}$$

which, together with (3.28) and (3.34), then imply that

$$\sum_{i=1}^{T_k} \frac{\gamma_k P_{T_k}}{\Gamma_k\beta_k(1-P_{T_k})p_i^2 P_{i-1}} \le \frac{4\gamma_k P_{T_k} T_k}{\Gamma_k\beta_k(1-P_{T_k})} = \frac{4\nu k^2}{L(T_k+3)}. \tag{3.36}$$

Using this observation, (3.22), (3.33), and (3.34), we have

$$\begin{aligned}\mathcal{B}_d(N) &\le \frac{4LV(x_0,x^*)}{\nu N(N+1)(1-P_{T_1})} + \frac{4M^2}{LN(N+1)}\sum_{k=1}^{N}\frac{k^2}{T_k+3}\\ &\le \frac{6LV(x_0,x^*)}{\nu N(N+1)} + \frac{4M^2}{LN(N+1)}\sum_{k=1}^{N}\frac{k^2}{T_k+3},\end{aligned}$$

which, in view of Theorem 1(a) and the definition of T_k in (3.28), then clearly implies (3.29).

Now let us show that part (b) holds. It follows from (3.33), and the definition of β_k and γ_k in (3.30) that

$$\beta_k \ge \frac{3L}{\nu(k+1)} \ge \frac{L\gamma_k}{\nu} \tag{3.37}$$

and hence that (3.13) holds. It also follows from (3.20) and (3.30) that

$$\Gamma_k = \frac{6}{k(k+1)(k+2)}, \quad k \ge 1, \tag{3.38}$$

and hence that

$$\frac{\gamma_k\beta_k}{\Gamma_k(1-P_{T_k})} = \frac{k(k+1)}{2}\frac{9L}{2\nu(k+1)} = \frac{9Lk}{4\nu}, \tag{3.39}$$

which implies that (3.23) holds. Using (3.30), (3.33), (3.35), and (3.37), we have

$$\begin{aligned}\sum_{i=1}^{T_k}\frac{\gamma_k P_{T_k}}{\Gamma_k\beta_k(1-P_{T_k})p_i^2P_{i-1}} &\le \frac{4\gamma_k P_{T_k}T_k}{\Gamma_k\beta_k(1-P_{T_k})} = \frac{4\nu k(k+1)^2P_{T_k}T_k}{9L(1-P_{T_k})^2}\\ &= \frac{8\nu k(k+1)^2(T_k+1)(T_k+2)}{9LT_k(T_k+3)^2} \le \frac{8\nu k(k+1)^2}{9LT_k}.\end{aligned} \tag{3.40}$$

Using this observation, (3.30), (3.38), and Theorem 1(b), we conclude that

$$\begin{aligned}\Psi(\bar{x}_N) - \Psi(x^*) &\le \frac{\gamma_N\beta_N\bar{V}(x^*)}{(1-P_{T_N})} + \frac{M^2\Gamma_N}{2\nu}\sum_{k=1}^{N}\frac{8\nu k(k+1)^2}{9LT_k}\\ &\le \frac{\gamma_N\beta_N\bar{V}(x^*)}{(1-P_{T_N})} + \frac{8L\tilde{D}}{3(N+1)(N+2)}\\ &\le \frac{L}{(N+1)(N+2)}\left(\frac{27\bar{V}(x^*)}{2\nu} + \frac{8\tilde{D}}{3}\right).\end{aligned}$$

□

Observe that by (3.3) and (3.32), when the selection of $p_t = t/2$, the definition of $\tilde{u}_t$ in the PS procedure can be simplified as

$$\tilde{u}_t = \frac{(t+2)(t-1)}{t(t+3)}\tilde{u}_{t-1} + \frac{2(t+1)}{t(t+3)}u_t.$$

In view of Corollary 1, we can establish the complexity of the GS algorithm for finding an ϵ-solution of problem (1.1).

Corollary 2 *Suppose that $\{p_t\}$ and $\{\theta_t\}$ are set to (3.27). Also assume that there exists an estimate $\mathcal{D}_X > 0$ s.t.*

$$V(x, y) \le \mathcal{D}_X, \quad \forall x, y \in X. \tag{3.41}$$

If $\{\beta_k\}$, $\{\gamma_k\}$, and $\{T_k\}$ are set to (3.28) with $\tilde{D} = 3\mathcal{D}_X/(2\nu)$ for some $N > 0$, then the total number of evaluations for ∇f and h' can be bounded by

$$\mathcal{O}\left(\sqrt{\frac{L\mathcal{D}_X}{\nu\epsilon}}\right) \tag{3.42}$$

and

$$\mathcal{O}\left\{\frac{M^2\mathcal{D}_X}{\nu\epsilon^2} + \sqrt{\frac{L\mathcal{D}_X}{\nu\epsilon}}\right\}, \tag{3.43}$$

respectively. Moreover, the above two complexity bounds also hold if X is bounded, and $\{\beta_k\}$, $\{\gamma_k\}$, and $\{T_k\}$ are set to (3.30) with $\tilde{D} = 81\mathcal{D}_X/(16\nu)$.

Proof In view of Corollary 1(a), if $\{\beta_k\}$, $\{\gamma_k\}$, and $\{T_k\}$ are set to (3.28), the total number of outer iterations (or gradient evaluations) performed by the GS algorithm to find an ϵ-solution of (1.1) can be bounded by

$$N \le \sqrt{\frac{L}{\epsilon}\left[\frac{3V(x_0, x^*)}{\nu} + 2\tilde{D}\right]} \le \sqrt{\frac{6L\mathcal{D}_X}{\nu\epsilon}}. \tag{3.44}$$

Moreover, using the definition of T_k in (3.28), we conclude that the total number of inner iterations (or subgradient evaluations) can be bounded by

$$\sum_{k=1}^{N} T_k \le \sum_{k=1}^{N}\left(\frac{M^2Nk^2}{\tilde{D}L^2} + 1\right) \le \frac{M^2N(N+1)^3}{3\tilde{D}L^2} + N = \frac{2\nu M^2N(N+1)^3}{9\mathcal{D}_X L^2} + N,$$

which, in view of (3.44), then clearly implies the bound in (3.43). Using Corollary 1(b) and similar arguments, we can show that the complexity bounds (3.42) and (3.43) also hold when X is bounded, and $\{\beta_k\}$, $\{\gamma_k\}$, and $\{T_k\}$ are set to (3.30) . □

In view of Corollary 2, the GS algorithm can achieve the optimal complexity bound for solving problem (1.1) in terms of the number of evaluations for both ∇f and h'. To the best of our knowledge, this is the first time that this type of algorithm has been developed in the literature.

It is also worth noting that we can relax the requirement on $\mathcal{D}_X$ in (3.41) to $V(x_0, x^*) \le \mathcal{D}_X$ or $\max_{x\in X} V(x, x^*) \le \mathcal{D}_X$, respectively, when the stepsize policies in (3.28) or in (3.30) is used. Accordingly, we can tighten the complexity bounds in (3.42) and (3.43) by a constant factor.

4 Stochastic gradient sliding

In this section, we consider the situation when the computation of stochastic subgradients of h is much easier than that of exact subgradients. This situation happens, for example, when h is given in the form of an expectation or as the summation of many nonsmooth components. By presenting a stochastic gradient sliding (SGS) method, we show that similar complexity bounds as in Sect. 3 for solving problem (1.1) can still be obtained in expectation or with high probability, but the iteration cost of the SGS method can be substantially smaller than that of the GS method.

More specifically, we assume that the nonsmooth component h is represented by a stochastic oracle (SO) satisfying (1.6) and (1.7). Sometimes, we augment (1.7) by a “light-tail” assumption:

$$\mathbb{E}[\exp(\|H(u,\xi) - h'(u)\|_*^2/\sigma^2)] \le \exp(1). \tag{4.1}$$

It can be easily seen that (4.1) implies (1.7) by Jensen’s inequality.

The stochastic gradient sliding (SGS) algorithm is obtained by simply replacing the exact subgradients in the PS procedure with the stochastic subgradients returned by the SO. This algorithm is formally described as follows.

Algorithm 2 The stochastic gradient sliding (SGS) algorithm

The algorithm is the same as GS except that the identity (3.2) in the PS procedure is replaced by

$$u_t = \operatorname{argmin}_{u\in X}\left\{g(u) + \langle H(u_{t-1}, \xi_{t-1}), u\rangle + \beta V(x,u) + \beta p_t V(u_{t-1}, u) + \mathcal{X}(u)\right\}. \tag{4.2}$$

The above modified PS procedure is called the SPS (stochastic PS) procedure.

We add a few remarks about the above SGS algorithm. Firstly, in this algorithm, we assume that the exact gradient of f will be used throughout the T_k inner iterations. This is different from the accelerated stochastic approximation in [11], where one needs to compute ∇f at each subgradient projection step. Secondly, let us denote

$$\tilde{l}_h(u_{t-1}, u) := h(u_{t-1}) + \langle H(u_{t-1}, \xi_{t-1}), u - u_{t-1}\rangle. \tag{4.3}$$

It can be easily seen that (4.2) is equivalent to

$$u_t = \operatorname*{argmin}_{u\in X} \left\{ g(u) + \tilde{l}_h(u_{t-1}, u) + \beta V(x, u) + \beta p_t V(u_{t-1}, u) + \mathcal{X}(u) \right\}. \tag{4.4}$$

This problem reduces to (3.2) if there is no stochastic noise associated with the SO, i.e., $\sigma = 0$ in (1.7). Thirdly, note that we have not provided the specification of $\{\beta_k\}$, $\{\gamma_k\}$, $\{T_k\}$, $\{p_t\}$ and $\{\theta_t\}$ in the SGS algorithm. Similarly to Sect. 3, we will return to this issue after establishing some convergence properties about the generic SPS procedure and SGS algorithm.

The following result describes some important convergence properties of the SPS procedure.

Proposition 3 *Assume that* $\{p_t\}$ *and* $\{\theta_t\}$ *in the* SPS *procedure satisfy* condspstheta). *Then for any* $t \geq 1$ *and* $u \in X$,

$$\beta(1-P_t)^{-1} V(u_t, u) + [\Phi(\tilde{u}_t) - \Phi(u)] \leq \beta P_t (1-P_t)^{-1} V(u_{t-1}, u) + P_t(1-P_t)^{-1} \sum_{i=1}^{t} (p_i P_{i-1})^{-1} \left[\frac{(M + \|\delta_i\|_*)^2}{2\nu\beta p_i} + \langle \delta_i, u - u_{i-1} \rangle \right], \tag{4.5}$$

where Φ *is defined in* genericspssubproblem)

$$\delta_t := H(u_{t-1}, \xi_{t-1}) - h'(u_{t-1}), \quad \textit{and} \quad h'(u_{t-1}) = \mathbb{E}[H(u_{t-1}, \xi_{t-1})]. \tag{4.6}$$

Proof Let $\tilde{l}_h(u_{t-1}, u)$ be defined in (4.3). Clearly, we have $\tilde{l}_h(u_{t-1}, u) - l_h(u_{t-1}, u) = \langle \delta_t, u - u_{t-1} \rangle$. Using this observation and (3.10), we obtain

$$\begin{aligned}
\Phi(u_t) &\leq g(u) + l_h(u_{t-1}, u_t) + \beta V(x, u_t) + \mathcal{X}(u_t) + M\|u_t - u_{t-1}\| \\
&= g(u) + \tilde{l}_h(u_{t-1}, u_t) - \langle \delta_t, u_t - u_{t-1} \rangle + \beta V(x, u_t) + \mathcal{X}(u_t) + M\|u_t - u_{t-1}\| \\
&\leq g(u) + \tilde{l}_h(u_{t-1}, u_t) + \beta V(x, u_t) + \mathcal{X}(u_t) + (M + \|\delta_t\|_*)\|u_t - u_{t-1}\|,
\end{aligned}$$

where the last inequality follows from the Cauchy–Schwarz inequality. Now applying Lemma 1 to (4.2), we obtain

$$\begin{aligned}
&g(u_t) + \tilde{l}_h(u_{t-1}, u_t) + \beta V(x, u_t) + \beta p_t V(u_{t-1}, u_t) + \mathcal{X}(u_t) \\
&\quad \leq g(u) + \tilde{l}_h(u_{t-1}, u) + \beta V(x, u) + \beta p_t V(u_{t-1}, u) + \mathcal{X}(u) - \beta(1+p_t) V(u_t, u) \\
&\quad = g(u) + l_h(u_{t-1}, u) + \langle \delta_t, u - u_{t-1} \rangle \\
&\qquad + \beta V(x, u) + \beta p_t V(u_{t-1}, u) + \mathcal{X}(u) - \beta(1+p_t) V(u_t, u) \\
&\quad \leq \Phi(u) + \beta p_t V(u_{t-1}, u) - \beta(1+p_t) V(u_t, u) + \langle \delta_t, u - u_{t-1} \rangle,
\end{aligned}$$

where the last inequality follows from the convexity of h and (3.4). Moreover, by the strong convexity of ω,

$$\begin{aligned}&-\beta p_t V(u_{t-1}, u_t) + (M + \|\delta_t\|_*)\|u_t - u_{t-1}\|\\ &\quad\le -\frac{\nu\beta p_t}{2}\|u_t - u_{t-1}\|^2 + (M + \|\delta_t\|_*)\|u_t - u_{t-1}\| \le \frac{(M + \|\delta_t\|_*)^2}{2\nu\beta p_t},\end{aligned}$$

where the last inequality follows from the simple fact that $-at^2/2 + bt \le b^2/(2a)$ for any $a > 0$. Combining the previous three inequalities, we conclude that

$$\begin{aligned}\Phi(u_t) - \Phi(u) \le \beta p_t V(u_{t-1}, u) - \beta(1 + p_t)V(u_t, u)\\ + \frac{(M + \|\delta_t\|_*)^2}{2\nu\beta p_t} + \langle\delta_t, u - u_{t-1}\rangle.\end{aligned}$$

Now dividing both sides of the above inequality by $1 + p_t$ and re-arranging the terms, we obtain

$$\begin{aligned}\beta V(u_t, u) + \frac{\Phi(u_t) - \Phi(u)}{1 + p_t} \le \frac{\beta p_t}{1 + p_t} V(u_{t-1}, u) + \frac{(M + \|\delta_t\|_*)^2}{2\nu\beta(1 + p_t)p_t}\\ + \frac{\langle\delta_t, u - u_{t-1}\rangle}{1 + p_t},\end{aligned}$$

which, in view of Lemma 2, then implies that

$$\begin{aligned}&\frac{\beta}{P_t} V(u_t, u) + \sum_{i=1}^{t} \frac{\Phi(u_i) - \Phi(u)}{P_i(1 + p_i)}\\ &\quad\le \beta V(u_0, u) + \sum_{i=1}^{t}\left[\frac{(M + \|\delta_i\|_*)^2}{2\nu\beta P_i(1 + p_i)p_i} + \frac{\langle\delta_i, u - u_{i-1}\rangle}{P_i(1 + p_i)}\right].\end{aligned} \tag{4.7}$$

The result then immediately follows from the above inequality and (3.12). □

It should be noted that the search points $\{u_t\}$ generated by different calls to the SPS procedure in different outer iterations of the SGS algorithm are distinct from each other. To avoid ambiguity, we use $u_{k,t}$, $k \ge 1$, $t \ge 0$, to denote the search points generated by the SPS procedure in the k-th outer iteration. Accordingly, we use

$$\delta_{k,t-1} := H(u_{k,t-1}, \xi_{t-1}) - h'(u_{k,t-1}), \quad k \ge 1, t \ge 1, \tag{4.8}$$

to denote the stochastic noises associated with the SO. Then, by (4.5), the definition of Φ_k in (2.9), and the origin of $(x_k, \tilde{x}_k)$ in the SGS algorithm, we have

$$\beta_k(1-P_{T_k})^{-1}V(x_k,u)+[\Phi_k(\tilde{x}_k)-\Phi_k(u)]$$
$$\le \beta_k P_{T_k}(1-P_{T_k})^{-1}V(x_{k-1},u)+P_{T_k}(1-P_{T_k})^{-1}\sum_{i=1}^{T_k}\frac{1}{p_iP_{i-1}}$$
$$\times\left[\frac{\left(M+\|\delta_{k,i-1}\|_*\right)^2}{2\nu\beta_k p_i}+\langle\delta_{k,i-1},u-u_{k,i-1}\rangle\right] \tag{4.9}$$

for any $u\in X$ and $k\ge 1$.

With the help of (4.9), we are now ready to establish the main convergence properties of the SGS algorithm.

Theorem 2 *Suppose that $\{p_t\}$, $\{\theta_t\}$, $\{\beta_k\}$, and $\{\gamma_k\}$ in the SGS algorithm satisfy* (3.8) *and* (3.13).

(a) *If relation* (3.21) *holds, then under Assumptions* (1.6) *and* (1.7)*, we have, for any $N\ge 1$,*

$$\mathbb{E}\left[\Psi(\bar{x}_N)-\Psi(x^*)\right]\le\tilde{\mathcal{B}}_d(N):=\frac{\Gamma_N\beta_1}{1-P_{T_1}}V(x_0,u)$$
$$+\frac{\Gamma_N}{\nu}\sum_{k=1}^{N}\sum_{i=1}^{T_k}\frac{(M^2+\sigma^2)\gamma_kP_{T_k}}{\beta_k\Gamma_k(1-P_{T_k})p_i^2P_{i-1}}, \tag{4.10}$$

where x^* is an arbitrary optimal solution of (1.1)*, and P_t and Γ_k are defined in* (3.3) *and* (3.20)*, respectively.*

(b) *If in addition, X is compact and Assumption* (4.1) *holds, then*

$$\mathrm{Prob}\left\{\Psi(\bar{x}_N)-\Psi(x^*)\ge\tilde{\mathcal{B}}_d(N)+\lambda\mathcal{B}_p(N)\right\}\le\exp\left\{-2\lambda^2/3\right\}+\exp\{-\lambda\}, \tag{4.11}$$

for any $\lambda>0$ and $N\ge 1$, where

$$\tilde{\mathcal{B}}_p(N):=\sigma\Gamma_N\left\{\frac{2\bar{V}(x^*)}{\nu}\sum_{k=1}^{N}\sum_{i=1}^{T_k}\left[\frac{\gamma_kP_{T_k}}{\Gamma_k(1-P_{T_k})p_iP_{i-1}}\right]^2\right\}^{\frac{1}{2}}$$
$$+\frac{\Gamma_N}{\nu}\sum_{k=1}^{N}\sum_{i=1}^{T_k}\frac{\sigma^2\gamma_kP_{T_k}}{\beta_k\Gamma_k(1-P_{T_k})p_i^2P_{i-1}}. \tag{4.12}$$

(c) *If X is compact and relation* (3.23) *[instead of* (3.21)*] holds, then both part (a) and part (b) still hold by replacing the first term in the definition of $\tilde{\mathcal{B}}_d(N)$ with $\gamma_N\beta_N\bar{V}(x^*)/(1-P_{T_N})$.*

Proof Using (3.19) and (4.9), we have

$$\begin{aligned}&\Psi(\bar{x}_k) - \Psi(u)\\ &\quad\le (1-\gamma_k)[\Psi(\bar{x}_{k-1}) - \Psi(u)] + \gamma_k \left\{ \frac{\beta_k}{1-P_{T_k}}[V(x_{k-1},u) - V(x_k,u)] \right.\\ &\qquad \left. + \frac{P_{T_k}}{1-P_{T_k}} \sum_{i=1}^{T_k} \frac{1}{p_i P_{i-1}} \left[\frac{\left(M + \|\delta_{k,i-1}\|_*\right)^2}{2\nu\beta_k p_i} + \langle \delta_{k,i-1}, u - u_{k,i-1}\rangle \right] \right\}.\end{aligned}$$

Using the above inequality and Lemma 2, we conclude that

$$\begin{aligned}&\Psi(\bar{x}_N) - \Psi(u)\\ &\quad\le \Gamma_N(1-\gamma_1)[\Psi(\bar{x}_0) - \Psi(u)] + \Gamma_N \sum_{k=1}^{N} \frac{\beta_k\gamma_k}{\Gamma_k(1-P_{T_k})}\left[V(x_{k-1},u) - V(x_k,u)\right]\\ &\qquad + \Gamma_N \sum_{k=1}^{N} \frac{\gamma_k P_{T_k}}{\Gamma_k(1-P_{T_k})} \sum_{i=1}^{T_k} \frac{1}{p_i P_{i-1}} \left[\frac{\left(M + \|\delta_{k,i-1}\|_*\right)^2}{2\nu\beta_k p_i} + \langle \delta_{k,i-1}, u - u_{k,i-1}\rangle \right].\end{aligned}$$

The above relation, in view of (3.25) and the fact that $\gamma_1 = 1$, then implies that

$$\begin{aligned}\Psi(\bar{x}_N) - \Psi(u) \le{}& \frac{\beta_k}{1-P_{T_1}} V(x_0,u) + \Gamma_N \sum_{k=1}^{N} \frac{\gamma_k P_{T_k}}{\Gamma_k(1-P_{T_k})}\\ &\times \sum_{i=1}^{T_k} \frac{1}{p_i P_{i-1}} \left[\frac{M^2 + \|\delta_{k,i-1}\|_*^2}{\nu\beta_k p_i} + \langle \delta_{k,i-1}, u - u_{k,i-1}\rangle \right]. \quad (4.13)\end{aligned}$$

We now provide bounds on the RHS of (4.13) in expectation or with high probability.

We first show part a). Note that by our assumptions on the SO, the random variable $\delta_{k,i-1}$ is independent of the search point $u_{k,i-1}$ and hence $\mathbb{E}[\langle \Delta_{k,i-1}, x^* - u_{k,i}\rangle] = 0$. In addition, Assumption (1.7) implies that $\mathbb{E}[\|\delta_{k,i-1}\|_*^2] \le \sigma^2$. Using the previous two observations and taking expectation on both sides of (4.13) (with $u = x^*$), we obtain (4.10).

We now show that part b) holds. Note that by our assumptions on the SO and the definition of $u_{k,i}$, the sequence $\{\langle \delta_{k,i-1}, x^* - u_{k,i-1}\rangle\}_{k\ge1, 1\le i\le T_k}$ is a martingale-difference sequence. Denoting

$$\alpha_{k,i} := \frac{\gamma_k P_{T_k}}{\Gamma_k(1-P_{T_k}) p_i P_{i-1}},$$

and using the large-deviation theorem for martingale-difference sequence (e.g., Lemma 2 of [13]) and the fact that

$$
\begin{aligned}
&\mathbb{E}\left[\exp\left\{\nu\alpha_{k,i}^2\langle\delta_{k,i-1}, x^* - u_{k,i}\rangle^2/\left(2\alpha_{k,i}^2\bar{V}(x^*)\sigma^2\right)\right\}\right]\\
&\quad\le \mathbb{E}\left[\exp\left\{\nu\alpha_{k,i}^2\|\delta_{k,i-1}\|_*^2\|x^* - u_{k,i}\|^2/\left(2\bar{V}(x^*)\sigma^2\right)\right\}\right]\\
&\quad\le \mathbb{E}\left[\exp\left\{\|\delta_{k,i-1}\|_*^2 V(u_{k,i}, x^*)/\left(\bar{V}(x^*)\sigma^2\right)\right\}\right]\\
&\quad\le \mathbb{E}\left[\exp\left\{\|\delta_{k,i-1}\|_*^2/\sigma^2\right\}\right] \le \exp\{1\},
\end{aligned}
$$

we conclude that

$$
\begin{aligned}
&\mathrm{Prob}\left\{\sum_{k=1}^{N}\sum_{i=1}^{T_k}\alpha_{k,i}\langle\delta_{k,i-1}, x^* - u_{k,i-1}\rangle > \lambda\sigma\sqrt{\frac{2\bar{V}(x^*)}{\nu}\sum_{k=1}^{N}\sum_{i=1}^{T_k}\alpha_{k,i}^2}\right\}\\
&\quad\le \exp\{-\lambda^2/3\}, \quad \forall\lambda > 0. \qquad (4.14)
\end{aligned}
$$

Now let

$$
S_{k,i} := \frac{\gamma_k P_{T_k}}{\beta_k\Gamma_k(1 - P_{T_k})p_i^2 P_{i-1}}
$$

and $S := \sum_{k=1}^{N}\sum_{i=1}^{T_k} S_{k,i}$. By the convexity of exponential function, we have

$$
\begin{aligned}
&\mathbb{E}\left[\exp\left\{\frac{1}{S}\sum_{k=1}^{N}\sum_{i=1}^{T_k} S_{k,i}\|\delta_{k,i}\|_*^2/\sigma^2\right\}\right]\\
&\quad\le \mathbb{E}\left[\frac{1}{S}\sum_{k=1}^{N}\sum_{i=1}^{T_k} S_i \exp\left\{\|\delta_{k,i}\|_*^2/\sigma^2\right\}\right] \le \exp\{1\}.
\end{aligned}
$$

where the last inequality follows from Assumption (4.1). Therefore, by Markov's inequality, for all $\lambda > 0$,

$$
\begin{aligned}
&\mathrm{Prob}\left\{\sum_{k=1}^{N}\sum_{i=1}^{T_k} S_{k,i}\|\delta_{k,i-1}\|_*^2 > (1+\lambda)\sigma^2\sum_{k=1}^{N}\sum_{i=1}^{T_k} S_{k,i}\right\}\\
&= \mathrm{Prob}\left\{\exp\left\{\frac{1}{S}\sum_{k=1}^{N}\sum_{i=1}^{T_k} S_{k,i}\|\delta_{k,i-1}\|_*^2/\sigma^2\right\} \ge \exp\{1+\lambda\}\right\} \le \exp\{-\lambda\}. \quad (4.15)
\end{aligned}
$$

Our result now directly follows from (4.13), (4.14) and (4.15). The proof of part c) is very similar to part (a) and (b) in view of the bound in (3.26), and hence the details are skipped. □

We now provide some specific choices for the parameters $\{\beta_k\}$, $\{\gamma_k\}$, $\{T_k\}$, $\{p_t\}$, and $\{\theta_t\}$ used in the SGS algorithm. In particular, while the stepsize policy in Corollary 3(a) requires the number of iterations N given a priori, such an assumption is not needed in Corollary 3(b) given that X is bounded. However, in order to provide some large-deviation results associated with the rate of convergence for the SGS algorithm [see (4.18) and (4.21) below], we need to assume the boundness of X in both Corollary 3(a) and Corollary 3(b).

Corollary 3 *Assume that* $\{p_t\}$ *and* $\{\theta_t\}$ *in the* SPS *procedure are set to* (3.27).

(a) *If N is given a priori,* $\{\beta_k\}$ *and* $\{\gamma_k\}$ *are set to* (3.28)*, and* $\{T_k\}$ *is given by*

$$T_k = \left\lceil \frac{N(M^2+\sigma^2)k^2}{\tilde{D}L^2} \right\rceil \tag{4.16}$$

for some $\tilde{D} > 0$. *Then under Assumptions* (1.6) *and* (1.7)*, we have*

$$\mathbb{E}\left[\Psi(\bar{x}_N) - \Psi(x^*)\right] \le \frac{2L}{N(N+1)}\left[\frac{3V(x_0,x^*)}{\nu} + 4\tilde{D}\right], \quad \forall N \ge 1. \tag{4.17}$$

If in addition, X is compact and Assumption (4.1) *holds, then*

$$\text{Prob}\left\{\Psi(\bar{x}_N) - \Psi(x^*) \ge \frac{2L}{N(N+1)}\left[\frac{3V(x_0,x^*)}{\nu} + 4(1+\lambda)\tilde{D} + \frac{4\lambda\sqrt{\tilde{D}\bar{V}(x^*)}}{\sqrt{3\nu}}\right]\right\}$$
$$\le \exp\left\{-2\lambda^2/3\right\} + \exp\{-\lambda\}, \ \forall \lambda > 0, \ \forall N \ge 1. \tag{4.18}$$

(b) *If X is compact,* $\{\beta_k\}$ *and* $\{\gamma_k\}$ *are set to* (3.30)*, and* $\{T_k\}$ *is given by*

$$T_k = \left\lceil \frac{(M^2+\sigma^2)(k+1)^3}{\tilde{D}L^2} \right\rceil \tag{4.19}$$

for some $\tilde{D} > 0$. *Then under Assumptions* (1.6) *and* (1.7)*, we have*

$$\mathbb{E}\left[\Psi(\bar{x}_N) - \Psi(x^*)\right] \le \frac{L}{(N+1)(N+2)}\left[\frac{27\bar{V}(x^*)}{2\nu} + \frac{16\tilde{D}}{3}\right], \quad \forall N \ge 1. \tag{4.20}$$

If in addition, Assumption (4.1) *holds, then*

$$\text{Prob}\left\{\Psi(\bar{x}_N) - \Psi(x^*) \ge \frac{L}{N(N+2)}\left[\frac{27\bar{V}(x^*)}{2\nu} + \frac{8}{3}(2+\lambda)\tilde{D} + \frac{12\lambda\sqrt{2\tilde{D}\bar{V}(x^*)}}{\sqrt{3\nu}}\right]\right\}$$
$$\le \exp\left\{-2\lambda^2/3\right\} + \exp\{-\lambda\}, \quad \forall \lambda > 0, \ \forall N \ge 1. \tag{4.21}$$

Proof We first show part (a). It can be easily seen from (3.34) that (3.13) holds. Moreover, Using (3.28), (3.33), and (3.34), we can easily see that (3.21) holds. By (3.33), (3.34), (3.36), (4.10), and (4.16), we have

$$\begin{aligned}\tilde{\mathcal{B}}_d(N) &\le \frac{4LV(x_0,x^*)}{\nu N(N+1)(1-P_{T_1})} + \frac{8\left(M^2+\sigma^2\right)}{LN(N+1)}\sum_{k=1}^{N}\frac{k^2}{T_k+3}\\ &\le \frac{6L}{\nu N(N+1)} + \frac{8\left(M^2+\sigma^2\right)}{LN(N+1)}\sum_{k=1}^{N}\frac{k^2}{T_k+3}\\ &\le \frac{2L}{N(N+1)}\left[\frac{3V(x_0,x^*)}{\nu}+4\tilde{D}\right], \qquad (4.22)\end{aligned}$$

which, in view of Theorem 2(a), then clearly implies (4.17). Now observe that by the definition of γ_k in (3.28) and relation (3.34),

$$\begin{aligned}\sum_{i=1}^{T_k}\left[\frac{\gamma_k P_{T_k}}{\Gamma_k(1-P_{T_k})p_iP_{i-1}}\right]^2 &= \left(\frac{2k}{T_k(T_k+3)}\right)^2\sum_{i=1}^{T_k}(i+1)^2\\ &= \left(\frac{2k}{T_k(T_k+3)}\right)^2\frac{(T_k+1)(T_k+2)(2T_k+3)}{6} \le \frac{8k^2}{3T_k},\end{aligned}$$

which together with (3.34), (3.36), and (4.12) then imply that

$$\begin{aligned}\tilde{\mathcal{B}}_p(N) &\le \frac{2\sigma}{N(N+1)}\left[\frac{2\bar{V}(x^*)}{\nu}\sum_{k=1}^{N}\frac{8k^2}{3T_k}\right]^{\frac{1}{2}} + \frac{8\sigma^2}{LN(N+1)}\sum_{k=1}^{N}\frac{k^2}{T_k+3}\\ &\le \frac{2\sigma}{N(N+1)}\left[\frac{16\tilde{D}L^2\bar{V}(x^*)}{3\nu(M^2+\sigma^2)}\right]^{\frac{1}{2}} + \frac{8\tilde{D}L\sigma^2}{N(N+1)(M^2+\sigma^2)}\\ &\le \frac{8L}{N(N+1)}\left(\frac{\sqrt{\tilde{D}\bar{V}(x^*)}}{\sqrt{3\nu}}+\tilde{D}\right).\end{aligned}$$

Using the above inequality, (4.22), Theorem 2(b), we obtain (4.18).

We now show that part b) holds. Note that P_t and Γ_k are given by (3.32) and (3.38), respectively. It then follows from (3.37) and (3.39) that both (3.13) and (3.23) hold. Using (3.40), the definitions of γ_k and β_k in (3.30), (4.19), and Theorem 2(c), we conclude that

$$\begin{aligned}\mathbb{E}\left[\Psi(\bar{x}_N)-\Psi(x^*)\right] &\le \frac{\gamma_N\beta_N\bar{V}(x^*)}{(1-P_{T_N})} + \frac{\Gamma_N(M^2+\sigma^2)}{\nu}\sum_{k=1}^{N}\sum_{i=1}^{T_k}\frac{\gamma_kP_{T_k}}{\beta_k\Gamma_k(1-P_{T_k})p_i^2P_{i-1}}\\ &\le \frac{\gamma_N\beta_N\bar{V}(x^*)}{(1-P_{T_N})} + \frac{16L\tilde{D}}{3\nu(N+1)(N+2)}\\ &\le \frac{L}{(N+1)(N+2)}\left(\frac{27\bar{V}(x^*)}{2\nu}+\frac{16\tilde{D}}{3}\right). \qquad (4.23)\end{aligned}$$

Now observe that by the definition of γ_k in (3.30), the fact that $p_t = t/2$, (3.32), and (3.38), we have

$$\begin{aligned}\sum_{i=1}^{T_k}\left[\frac{\gamma_k P_{T_k}}{\Gamma_k(1-P_{T_k})p_i P_{i-1}}\right]^2 &= \left(\frac{k(k+1)}{T_k(T_k+3)}\right)^2 \sum_{i=1}^{T_k}(i+1)^2 \\ &= \left(\frac{k(k+1)}{T_k(T_k+3)}\right)^2 \frac{(T_k+1)(T_k+2)(2T_k+3)}{6} \le \frac{8k^4}{3T_k},\end{aligned}$$

which together with (3.38), (3.40), and (4.12) then imply that

$$\begin{aligned}\tilde{\mathcal{B}}_p(N) &\le \frac{6}{N(N+1)(N+2)}\left[\sigma\left(\frac{2\bar{V}(x^*)}{\nu}\sum_{k=1}^{N}\frac{8k^4}{3T_k}\right)^{\frac{1}{2}} + \frac{4\sigma^2}{9L}\sum_{k=1}^{N}\frac{k(k+1)^2}{T_k}\right] \\ &= \frac{6}{N(N+1)(N+2)}\left[\sigma\left(\frac{8\bar{V}(x^*)\tilde{D}L^2N(N+1)}{3\nu(M^2+\sigma^2)}\right)^{\frac{1}{2}} + \frac{4\sigma^2 L\tilde{D}N}{9(M^2+\sigma^2)}\right] \\ &\le \frac{6L}{N(N+2)}\left(\frac{2\sqrt{2\bar{V}(x^*)\tilde{D}}}{\sqrt{3\nu}} + \frac{4\tilde{D}}{9}\right).\end{aligned}$$

The relation in (4.21) then immediately follows from the above inequality, (4.23), and Theorem 2(c). □

Corollary 4 below states the complexity of the SGS algorithm for finding a stochastic ϵ-solution of (1.1), i.e., a point $\bar{x} \in X$ s.t. $\mathbb{E}[\Psi(\bar{x}) - \Psi^*] \le \epsilon$ for some $\epsilon > 0$, as well as a stochastic (ϵ, Λ)-solution of (1.1), i.e., a point $\bar{x} \in X$ s.t. $\mathrm{Prob}\,\{\Psi(\bar{x}) - \Psi^* \le \epsilon\} > 1 - \Lambda$ for some $\epsilon > 0$ and $\Lambda \in (0, 1)$. Since this result follows as an immediate consequence of Corollary 3, we skipped the details of its proof.

Corollary 4 *Suppose that $\{p_t\}$ and $\{\theta_t\}$ are set to (3.27). Also assume that there exists an estimate $\mathcal{D}_X > 0$ s.t. (3.41) holds.*

(a) *If $\{\beta_k\}$ and $\{\gamma_k\}$ are set to (3.28), and $\{T_k\}$ is given by (4.16) with $\tilde{D} = 3\mathcal{D}_X/(4\nu)$ for some $N > 0$, then the number of evaluations for ∇f and h', respectively, required by the SGS algorithm to find a stochastic ϵ-solution of (1.1) can be bounded by*

$$\mathcal{O}\left(\sqrt{\frac{L\mathcal{D}_X}{\nu\epsilon}}\right) \tag{4.24}$$

and

$$\mathcal{O}\left\{\frac{(M^2+\sigma^2)\mathcal{D}_X}{\nu\epsilon^2} + \sqrt{\frac{L\mathcal{D}_X}{\nu\epsilon}}\right\}. \tag{4.25}$$

(b) *If in addition, Assumption* (4.1) *holds, then the number of evaluations for* ∇f *and* h'*, respectively, required by the SGS algorithm to find a stochastic* (ϵ, Λ)*-solution of* (1.1) *can be bounded by*

$$\mathcal{O}\left\{\sqrt{\frac{L\mathcal{D}_X}{\nu\epsilon}}\max\left(1, \log\frac{1}{\Lambda}\right)\right\} \tag{4.26}$$

and

$$\mathcal{O}\left\{\frac{M^2\mathcal{D}_X}{\nu\epsilon^2}\max\left(1, \log^2\frac{1}{\Lambda}\right) + \sqrt{\frac{L\mathcal{D}_X}{\nu\epsilon}}\max\left(1, \log\frac{1}{\Lambda}\right)\right\}. \tag{4.27}$$

(c) *The above bounds in part (a) and (b) still hold if X is bounded,* $\{\beta_k\}$ *and* $\{\gamma_k\}$ *are set to* (3.30)*, and* $\{T_k\}$ *is given by* (4.19) *with* $\tilde{D} = 81\mathcal{D}_X/(32\nu)$.

Observe that both bounds in (4.24) and (4.25) on the number of evaluations for ∇f and h' are essentially not improvable. In fact, to the best of our knowledge, this is the first time that the $\mathcal{O}(1/\sqrt{\epsilon})$ complexity bound on gradient evaluations has been established in the literature for stochastic approximation type algorithms applied to solve the composite problem in (1.1).

5 Generalization to strongly convex and structured nonsmooth optimization

Our goal in this section is to show that the gradient sliding techniques developed in Sects. 3 and 4 can be further generalized to some other important classes of CP problems. More specifically, we first study in Sect. 5.1 the composite CP problems in (1.1) with f being strongly convex, and then consider in Sect. 5.2 the case where f is a special nonsmooth function given in a bi-linear saddle point form. Throughout this section, we assume that the nonsmooth component h is represented by a SO (see Sect. 1). It is clear that our discussion covers also the deterministic composite problems as certain special cases by setting $\sigma = 0$ in (1.7) and (4.1).

5.1 Strongly convex optimization

In this section, we assume that the smooth component f in (1.1) is strongly convex, i.e., $\exists \mu > 0$ such that

$$f(x) \geq f(y) + \langle \nabla f(y), x - y\rangle + \frac{\mu}{2}\|x - y\|^2, \quad \forall x, y \in X. \tag{5.1}$$

In addition, throughout this section, we assume that the prox-function grows quadratically so that (2.4) is satisfied.

One way to solve these strongly convex composite problems is to apply the aforementioned accelerated stochastic approximation algorithm which would require

$\mathcal{O}(1/\epsilon)$ evaluations for ∇f and h' to find an ϵ-solution of (1.1) [5,6]. However, we will show in this subsection that this bound on the number of evaluations for ∇f can be significantly reduced to $\mathcal{O}(\log(1/\epsilon))$, by properly restarting the SGS algorithm in Sect. 4. This multi-phase stochastic gradient sliding (M-SGS) algorithm is formally described as follows.

Algorithm 3 The multi-phase stochastic gradient sliding (M-SGS) algorithm

```
Input: Initial point $y_0 \in X$, iteration limit $N_0$, and an initial estimate $\Delta_0$ s.t. $\Psi(y_0) - \Psi^* \le \Delta_0$.
for $s = 1, 2, \ldots, S$ do
    Run the SGS algorithm with $x_0 = y_{s-1}$, $N = N_0$, $\{p_t\}$ and $\{\theta_t\}$ in (3.27), $\{\beta_k\}$ and $\{\gamma_k\}$ in (3.28),
      and $\{T_k\}$ in (4.16) with $\tilde{D} = \Delta_0/(\nu\mu 2^s)$, and let $y_s$ be its output solution.
end for
Output: $y_S$.
```

We now establish the main convergence properties of the M-SGS algorithm described above.

Theorem 3 *If* $N_0 = \lceil 4\sqrt{2L/(\nu\mu)} \rceil$ *in the MGS algorithm, then*

$$\mathbb{E}[\Psi(y_s) - \Psi^*] \le \frac{\Delta_0}{2^s}, \quad s \ge 0. \tag{5.2}$$

As a consequence, the total number of evaluations for ∇f *and* H*, respectively, required by the M-SGS algorithm to find a stochastic* ϵ*-solution of* (1.1) *can be bounded by*

$$\mathcal{O}\left(\sqrt{\frac{L}{\nu\mu}} \log_2 \max\left\{\frac{\Delta_0}{\epsilon}, 1\right\}\right) \tag{5.3}$$

and

$$\mathcal{O}\left(\frac{M^2 + \sigma^2}{\nu\mu\epsilon} + \sqrt{\frac{L}{\nu\mu}} \log_2 \max\left\{\frac{\Delta_0}{\epsilon}, 1\right\}\right). \tag{5.4}$$

Proof We show (5.2) by induction. Note that (5.2) clearly holds for $s = 0$ by our assumption on Δ_0. Now assume that (5.2) holds at phase $s-1$, i.e., $\Psi(y_{s-1}) - \Psi^* \le \Delta_0/2^{(s-1)}$ for some $s \ge 1$. In view of Corollary 3 and the definition of y_s, we have

$$\begin{aligned}\mathbb{E}[\Psi(y_s) - \Psi^* | y_{s-1}] &\le \frac{2L}{N_0(N_0+1)}\left[\frac{3V(y_{s-1}, x^*)}{\nu} + 4\tilde{D}\right] \\ &\le \frac{2L}{N_0^2}\left[\frac{6}{\nu\mu}(\Psi(y_{s-1}) - \Psi^*) + 4\tilde{D}\right].\end{aligned}$$

where the second inequality follows from the strong convexity of Ψ and (2.4). Now taking expectation on both sides of the above inequality w.r.t. y_{s-1}, and using the induction hypothesis and the definition of $\tilde{D}$ in the M-SGS algorithm, we conclude that

$$\mathbb{E}[\Psi(y_s) - \Psi^*] \le \frac{2L}{N_0^2}\frac{8\Delta_0}{\nu\mu 2^{s-1}} \le \frac{\Delta_0}{2^s},$$

where the last inequality follows from the definition of N_0. Now, by (5.2), the total number of phases performed by the M-SGS algorithm can be bounded by $S = \left\lceil \log_2 \max\left\{\frac{\Delta_0}{\epsilon}, 1\right\}\right\rceil$. Using this observation, we can easily see that the total number of gradient evaluations of ∇f is given by $N_0 S$, which is bounded by (5.3). Now let us provide a bound on total number of stochastic subgradient evaluations of h'. Without loss of generality, let us assume that $\Delta_0 > \epsilon$. Using the previous bound on S and the definition of T_k, the total number of stochastic subgradient evaluations of h' can be bounded by

$$\begin{aligned}\sum_{s=1}^{S}\sum_{k=1}^{N_0} T_k &\le \sum_{s=1}^{S}\sum_{k=1}^{N_0}\left(\frac{\nu\mu N_0(M^2+\sigma^2)k^2}{\Delta_0 L^2}2^s + 1\right)\\ &\le \sum_{s=1}^{S}\left[\frac{\nu\mu N_0(M^2+\sigma^2)}{3\Delta_0 L^2}(N_0+1)^3 2^s + N_0\right]\\ &\le \frac{\nu\mu N_0(N_0+1)^3(M^2+\sigma^2)}{3\Delta_0 L^2}2^{S+1} + N_0 S\\ &\le \frac{4\nu\mu N_0(N_0+1)^3(M^2+\sigma^2)}{3\epsilon L^2} + N_0 S.\end{aligned}$$

This observation, in view of the definition of N_0, then clearly implies the bound in (5.4). □

We now add a few remarks about the results obtained in Theorem 3. Firstly, the M-SGS algorithm possesses optimal complexity bounds in terms of the number of gradient evaluations for ∇f and subgradient evaluations for h', while existing algorithms only exhibit optimal complexity bounds on the number of stochastic subgradient evaluations (see [6]). Secondly, in Theorem 3, we only establish the optimal convergence of the M-SGS algorithm in expectation. It is also possible to establish the optimal convergence of this algorithm with high probability by making use of the light-tail assumption in (4.1) and a domain shrinking procedure similarly to the one studied in Section 3 of [6].

5.2 Structured nonsmooth problems

Our goal in this subsection is to further generalize the gradient sliding algorithms to the situation when f is nonsmooth, but can be closely approximated by a certain smooth convex function.

More specifically, we assume that f is given in the form of

$$f(x) = \max_{y\in Y}\langle Ax, y\rangle - J(y), \tag{5.5}$$

where $A : \mathbb{R}^n \to \mathbb{R}^m$ denotes a linear operator, Y is a closed convex set, and $J : Y \to \Re$ is a relatively simple, proper, convex, and lower semi-continuous (l.s.c.) function (i.e., problem (5.8) below is easy to solve). Observe that if J is the convex conjugate of some convex function F and $Y \equiv \mathcal{Y}$, then problem (1.1) with f given in (5.5) can be written equivalently as

$$\min_{x \in X} h(x) + F(Ax),$$

Similarly to the previous subsection, we focus on the situation when h is represented by a SO. Stochastic composite problems in this form have wide applications in machine learning, for example, to minimize the regularized loss function of

$$\min_{x \in X} \mathbb{E}_{\xi}[l(x, \xi)] + F(Ax),$$

where $l(\cdot, \xi)$ is a convex loss function for any $\xi \in \Xi$ and $F(Kx)$ is a certain regularization (e.g., low rank tensor [10,21], overlapped group lasso [7,14], and graph regularization [7,20]).

Since f in (5.5) is nonsmooth, we cannot directly apply the gradient sliding methods developed in the previous sections. However, as shown by Nesterov [17], the function $f(\cdot)$ in (5.5) can be closely approximated by a class of smooth convex functions. More specifically, for a given strongly convex function $v : Y \to \mathbb{R}$ such that

$$v(y) \geq v(x) + \langle \nabla v(x), y - x \rangle + \frac{\nu'}{2} \|y - x\|^2, \quad \forall x, y \in Y \tag{5.6}$$

for some $\nu' > 0$, let us denote $c_v := \text{argmin}_{y \in Y} v(y)$, $d(y) := v(y) - v(c_v) - \langle \nabla v(c_v), y - c_v \rangle$ and

$$\mathcal{D}_Y := \max_{y \in Y} d(y). \tag{5.7}$$

Then the function $f(\cdot)$ in (5.5) can be closely approximated by

$$f_\eta(x) := \max_y \left\{ \langle Ax, y \rangle - J(y) - \eta\, d(y) : \; y \in Y \right\}. \tag{5.8}$$

Indeed, by definition we have $0 \leq V(y) \leq \mathcal{D}_Y$ and hence, for any $\eta \geq 0$,

$$f(x) - \eta \mathcal{D}_Y \leq f_\eta(x) \leq f(x), \quad \forall x \in X. \tag{5.9}$$

Moreover, Nesterov [17] shows that $f_\eta(\cdot)$ is differentiable and its gradients are Lipschitz continuous with the Lipschitz constant given by

$$\mathcal{L}_\eta := \frac{\|A\|^2}{\eta \nu'}. \tag{5.10}$$

We are now ready to present a smoothing stochastic gradient sliding (S-SGS) method and study its convergence properties.

Theorem 4 *Let $(\bar{x}_k, x_k)$ be the search points generated by a smoothing stochastic gradient sliding (S-SGS) method, which is obtained by replacing f with $f_\eta(\cdot)$ in the definition of g_k in the SGS method. Suppose that $\{p_t\}$ and $\{\theta_t\}$ in the* SPS *procedure are set to (3.27). Also assume that $\{\beta_k\}$ and $\{\gamma_k\}$ are set to (3.28) and that T_k is given by (4.16) with $\tilde{D} = 3\mathcal{D}_X/(4\nu)$ for some $N \geq 1$, where $\mathcal{D}_X$ is given by (3.41). If*

$$\eta = \frac{2\|A\|}{N}\sqrt{\frac{3\mathcal{D}_X}{\nu\nu'\mathcal{D}_Y}},$$

then the total number of outer iterations and inner iterations performed by the S-SGS algorithm to find an ϵ-solution of (1.1) can be bounded by

$$\mathcal{O}\left(\frac{\|A\|\sqrt{\mathcal{D}_X\mathcal{D}_Y}}{\epsilon\sqrt{\nu\nu'}}\right) \tag{5.11}$$

and

$$\mathcal{O}\left\{\frac{(M^2+\sigma^2)\|A\|^2 V(x_0,x^*)}{\nu\epsilon^2} + \frac{\|A\|\sqrt{\mathcal{D}_Y V(x_0,x^*)}}{\sqrt{\nu\nu'}\epsilon}\right\}, \tag{5.12}$$

respectively.

Proof Let us denote $\Psi_\eta(x) = f_\eta(x) + h(x) + \mathcal{X}(x)$. In view of (4.17) and (5.10), we have

$$\begin{aligned}\mathbb{E}[\Psi_\eta(\bar{x}_N) - \Psi_\eta(x)] &\leq \frac{2L_\eta}{N(N+1)}\left[\frac{3V(x_0,x)}{\nu} + 4\tilde{D}\right]\\ &= \frac{2\|A\|^2}{\eta\nu' N(N+1)}\left[\frac{3V(x_0,x)}{\nu} + 4\tilde{D}\right], \quad \forall x \in X,\ N \geq 1.\end{aligned}$$

Moreover, it follows from (5.9) that

$$\Psi_\eta(\bar{x}_N) - \Psi_\eta(x) \geq \Psi(\bar{x}_N) - \Psi(x) - \eta\mathcal{D}_Y.$$

Combining the above two inequalities, we obtain

$$\mathbb{E}[\Psi(\bar{x}_N) - \Psi(x)] \leq \frac{2\|A\|^2}{\eta\nu' N(N+1)}\left[\frac{3V(x_0,x)}{\nu} + 4\tilde{D}\right] + \eta\mathcal{D}_Y, \quad \forall x \in X,$$

which implies that

$$\mathbb{E}[\Psi(\bar{x}_N) - \Psi(x^*)] \leq \frac{2\|A\|^2}{\eta\nu' N(N+1)}\left[\frac{3\mathcal{D}_X}{\nu} + 4\tilde{D}\right] + \eta\mathcal{D}_Y. \tag{5.13}$$

Plugging the value of $\tilde{D}$ and η into the above bound, we can easily see that

$$\mathbb{E}[\Psi(\bar{x}_N) - \Psi(x^*)] \le \frac{4\sqrt{3}\|A\|\sqrt{\mathcal{D}_X\mathcal{D}_Y}}{\sqrt{\nu\nu'}N}, \quad \forall x \in X,\ N \ge 1.$$

It then follows from the above relation that the total number of outer iterations to find an ϵ-solution of problem (5.5) can be bounded by

$$\bar{N}(\epsilon) = \frac{4\sqrt{3}\|A\|\sqrt{\mathcal{D}_X\mathcal{D}_Y}}{\sqrt{\nu\nu'}\epsilon}.$$

Now observe that the total number of inner iterations is bounded by

$$\sum_{k=1}^{\bar{N}(\epsilon)} T_k = \sum_{k=1}^{\bar{N}(\epsilon)} \left[\frac{(M^2+\sigma^2)\bar{N}(\epsilon)k^2}{\tilde{D}L_\eta^2} + 1\right] = \sum_{k=1}^{\bar{N}(\epsilon)} \left[\frac{(M^2+\sigma^2)\bar{N}(\epsilon)k^2}{\tilde{D}L_\eta^2} + 1\right].$$

Combining these two observations, we conclude that the total number of inner iterations is bounded by (4). □

In view of Theorem 4, by using the smoothing SGS algorithm, we can significantly reduce the number of outer iterations, and hence the number of times to access the linear operator A and A^T, from $\mathcal{O}(1/\epsilon^2)$ to $\mathcal{O}(1/\epsilon)$ in order to find an ϵ-solution of (1.1), while still maintaining the optimal bound on the total number of stochastic subgradient evaluations for h'. It should be noted that, by using the result in Theorem 2(b), we can show that the aforementioned savings on the access to the linear operator A and A^T also hold with overwhelming probability under the light-tail assumption in (4.1) associated with the SO.

6 Concluding remarks

In this paper, we present a new class of first-order method which can significantly reduce the number of gradient evaluations for ∇f required to solve the composite problems in (1.1). More specifically, we show that by using these algorithms, the total number of gradient evaluations can be significantly reduced from $\mathcal{O}(1/\epsilon^2)$ to $\mathcal{O}(1/\sqrt{\epsilon})$. As a result, these algorithms have the potential to significantly accelerate first-order methods for solving the composite problem in (1.1), especially when the bottleneck exists in the computation (or communication in the case of distributed computing) of the gradient of the smooth component, as happened in many applications. We also establish similar complexity bounds for solving an important class of stochastic composite optimization problems by developing the stochastic gradient sliding methods. By properly restarting the gradient sliding algorithms, we demonstrate that dramatic saving on gradient evaluations (from $\mathcal{O}(1/\epsilon)$ to $\mathcal{O}(\log(1/\epsilon))$ can be achieved for solving strongly convex problems. Generalization to the case when f is nonsmooth but possessing a bilinear saddle point structure has also been discussed.

It should be pointed out that this paper focuses only on theoretical studies for the convergence properties associated with the gradient sliding algorithms. The practical performance for these algorithms, however, will certainly depend on our estimation for a few problem parameters, e.g., the Lipschitz constants L and M. In addition, the sliding periods $\{T_k\}$ in both GS and SGS have been specified in a conservative way to obtain the optimal complexity bounds for gradient and subgradient evaluations. We expect that the practical performance of these algorithms will be further improved with proper incorporation of certain adaptive search procedures on L, M, and $\{T_k\}$, which will be very interesting research topics in the future.

Moreover, we proposed SGS for composite problems where h' is given by a stochastic oracle. In some applications, instead of having a stochastic h, the smooth part f may be stochastic. In this case, the number of stochastic gradients of f that we need to compute will be bounded by $\mathcal{O}(1/\epsilon^2)$, which is not improvable in general. Therefore, SGS cannot help in this general case. However, it will be interesting to understand whether the SGS algorithm helps in some special cases, e.g., when the variance of the stochastic gradients decreases as the algorithm proceeds.

References

1. Auslender, A., Teboulle, M.: Interior gradient and proximal methods for convex and conic optimization. SIAM J. Optim. **16**, 697–725 (2006)
2. Bauschke, H.H., Borwein, J.M., Combettes, P.L.: Bregman monotone optimization algorithms. SIAM J. Controal Optim. **42**, 596–636 (2003)
3. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**, 183–202 (2009)
4. Bregman, L.M.: The relaxation method of finding the common point convex sets and its application to the solution of problems in convex programming. USSR Comput. Math. Phys. **7**, 200–217 (1967)
5. Ghadimi, S., Lan, G.: Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, I: a generic algorithmic framework. SIAM J. Optim. **22**, 1469–1492 (2012)
6. Ghadimi, S., Lan, G.: Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, II: shrinking procedures and optimal algorithms. SIAM J. Optim. **23**, 2061–2089 (2013)
7. Jacob, L., Obozinski, G., Vert, J.-P.: Group lasso with overlap and graph lasso. In: Proceedings of the 26th International Conference on Machine Learning (2009)
8. Juditsky, A., Nemirovski, A.S., Tauvel, C.: Solving Variational Inequalities with Stochastic Mirror-Prox Algorithm. Georgia Institute of Technology, Atlanta (2011)
9. Kiwiel, K.C.: Proximal minimization methods with generalized Bregman functions. SIAM J. Controal Optim. **35**, 1142–1168 (1997)
10. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Rev. **51**(3), 455–500 (2009)
11. Lan, G.: An optimal method for stochastic composite optimization. Math. Program. **133**(1), 365–397 (2012)
12. Lan, G.: The Complexity of Large-Scale Convex Programming Under a Linear Optimization Oracle. Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL (2013). http://www.optimization-online.org/
13. Lan, G., Nemirovski, A.S., Shapiro, A.: Validation analysis of mirror descent stochastic approximation method. Math. Program. **134**, 425–458 (2012)
14. Mairal, J., Jenatton, R., Obozinski, G., Bach, F.: Convex and network flow optimization for structured sparsity. J. Mach. Learn. Res. **12**, 2681–2720 (2011)
15. Nesterov, Y.E.: A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. Dokl. SSSR **269**, 543–547 (1983)
16. Nesterov, Y.E.: Introductory Lectures on Convex Optimization: A Basic Course. Kluwer Academic Publishers, Massachusetts (2004)

17. Nesterov, Y.E.: Smooth minimization of nonsmooth functions. Math. Program. **103**, 127–152 (2005)
18. Nesterov, Y.E.: Gradient Methods for Minimizing Composite Objective Functions. Technical report, Center for Operations Research and Econometrics (CORE), Catholic University of Louvain (2007)
19. Schmidt, M., Roux, N.L., Bach, F.R.: Convergence rates of inexact proximal-gradient methods for convex optimization. Adv. Neural Inf. Process. Syst. **24**, 1458–1466 (2011)
20. Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., Knight, K.: Sparsity and smoothness via the fused lasso. J. R. Stat. Soc. B **67**(1), 91–108 (2005)
21. Tomioka, R., Suzuki, T., Hayashi, K., Kashima, H.: Statistical performance of convex tensor decomposition. Adv. Neural Inf. Process. Syst. **24** (2011)
22. Tseng, P.: On Accelerated Proximal Gradient Methods for Convex–Concave Optimization. University of Washington, Seattle (2008)
23. Villa, S., Salzo, S., Baldassarre, L., Verri, A.: Accelerated and inexact forward–backward algorithms. SIAM J. Optim. **3**, 1607–1633 (2013)