TOP (2018) 26:110–145
https://doi.org/10.1007/s11750-017-0462-3

ORIGINAL PAPER

Solving structured nonsmooth convex optimization with complexity $\mathcal{O}(\varepsilon^{-1/2})$

Masoud Ahookhosh[1] · **Arnold Neumaier**[1]

Received: 6 July 2016 / Accepted: 7 August 2017 / Published online: 14 August 2017

Abstract This paper describes an algorithm for solving structured nonsmooth convex optimization problems using the optimal subgradient algorithm (OSGA), which is a first-order method with the complexity $\mathcal{O}(\varepsilon^{-2})$ for Lipschitz continuous nonsmooth problems and $\mathcal{O}(\varepsilon^{-1/2})$ for smooth problems with Lipschitz continuous gradient. If the nonsmoothness of the problem is manifested in a structured way, we reformulate the problem so that it can be solved efficiently by a new setup of OSGA (called OSGA-V) with the complexity $\mathcal{O}(\varepsilon^{-1/2})$. Further, to solve the reformulated problem, we equip OSGA-O with an appropriate prox-function for which the OSGA-O subproblem can be solved either in a closed form or by a simple iterative scheme, which decreases the computational cost of applying the algorithm for large-scale problems. We show that applying the new scheme is feasible for many problems arising in applications. Some numerical results are reported confirming the theoretical foundations.

Keywords Structured nonsmooth convex optimization · Subgradient methods · Proximity operator · Optimal complexity · First-order black-box information

Mathematics Subject Classification 90C25 · 90C60 · 49M37 · 65K05 · 68Q25

✉ Masoud Ahookhosh
masoud.ahookhosh@univie.ac.at

Arnold Neumaier
Arnold.Neumaier@univie.ac.at

[1] Faculty of Mathematics, University of Vienna, Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria

1 Introduction

Subgradient methods are a class of first-order methods that have been developed to solve convex nonsmooth optimization problems, dating back to 1960; see, e.g., (Polyak 1987; Shor 1985). In general, they only need function values and subgradients, and not only inherit the basic features of general first-order methods such as low memory requirement and simple structure but also are able to deal with every convex optimization problem. They are suitable for solving convex problems with a large number of variables, say several millions. Although these features make them very attractive for applications involving high-dimensional data, they usually suffer from slow convergence, which finally limits the attainable accuracy. In 1983, Nemirovsky and Yudin (1983) derived the worst-case complexity bound of first-order methods for several classes of problems to achieve an ε-solution, which is $\mathcal{O}(\varepsilon^{-2})$ for Lipschitz continuous nonsmooth problems and $\mathcal{O}(\varepsilon^{-1/2})$ for smooth problems with Lipschitz continuous gradients. The low convergence speed of subgradient methods suggests that they often reach an ε-solution in the number of iterations closing to the worst-case complexity bound on iterations.

In Nemirovsky and Yudin (1983), it was proved that the subgradient, subgradient projection, and mirror descent methods attain the optimal complexity of first-order methods for solving Lipschitz continuous nonsmooth problems; here, the mirror descent method is a generalization of the subgradient projection method, cf. (Beck and Teboulle 2003; Beck et al. 2010). Nesterov (2011), Nesterov (2006) proposed some primal-dual subgradient schemes, which attain the complexity $\mathcal{O}(\varepsilon^{-2})$ for Lipschitz continuous nonsmooth problems. Juditsky and Nesterov (2014) proposed a primal-dual subgradient scheme for uniformly convex functions with an unknown convexity parameter, which attains the complexity close to the optimal bound. Nesterov (1983) and later in Nesterov (2004) proposed some gradient methods for solving smooth problems with Lipschitz continuous gradients attaining the complexity $\mathcal{O}(\varepsilon^{-1/2})$. He also in Nesterov (2005a, b) proposed some smoothing methods for structured nonsmooth problems. Smoothing methods also have been studied by many authors; see, e.g., Beck and Teboulle (2012), Boţ and Hendrich (2013), Boţ and Hendrich (2015), and Devolder et al. (2012).

In many fields of applied sciences and engineering such as signal and image processing, geophysics, economic, machine learning, and statistics, there exist many applications that can be modeled as a convex optimization problem, in which the objective function is a composite function of a smooth function with Lipschitz continuous gradients and a nonsmooth function; see Ahookhosh (2016) and references therein. Studying this class of problems using first-order methods has dominated the convex optimization literature in the recent years. Nesterov (2013, 2015) proposed some gradient methods for solving composite problems obtaining the complexity $\mathcal{O}(\varepsilon^{-1/2})$. For this class of problems, other first-order methods with the complexity $\mathcal{O}(\varepsilon^{-1/2})$ have been developed by Auslender and Teboulle (2006), Beck and Teboulle (2012), Chen et al. (2014, 2017, 2015, 2014), Devolder et al. (2013), Gonzaga and Karas (2013), Gonzaga et al. (2013), Lan (2015), Lan et al. (2011),

and Tseng (2008). In particular, Neumaier (2016) proposed an optimal subgradient algorithm (OSGA) attaining the complexity $\mathcal{O}(\varepsilon^{-2})$ for Lipschitz continuous nonsmooth problems and the complexity $\mathcal{O}(\varepsilon^{-1/2})$ for smooth problems with Lipschitz continuous gradients. OSGA is a black-box method and does not need to know about global information of the objective function such as Lipschitz constants.

1.1 Content

This paper focuses on a class of structured nonsmooth convex constrained optimization problems that is a generalization of the composite problems, which is frequently found in applications. OSGA behaves well for composite problems in applications; see Ahookhosh (2016) and Ahookhosh and Neumaier (2017), Ahookhosh and Neumaier (2017); however, it does not attain the complexity $\mathcal{O}(\varepsilon^{-1/2})$ for this class of problems. Hence, we first reformulate the problem considered in a way that only the smooth part remains in the objective, in the cost of adding a functional constraint to our feasible domain. Afterward, we propose a suitable prox-function, provide a new setup for OSGA called OSGA-O for the reformulated problem, and show that solving the OSGA-O auxiliary problem for the reformulated problem is equivalent to solving a proximal-like problem. It is shown that this proximal-like subproblem can be solved efficiently for many problems appearing in applications either in a closed form or by a simple iterative scheme. Due to this reformulation, the problem can be solved by OSGA-O with the complexity $\mathcal{O}(\varepsilon^{-1/2})$. Finally, some numerical results are reported suggesting a good behavior of OSGA-O.

The underlying function of the subproblem of OSGA is quasi-concave and finding its solution is the most costly part of the algorithm. Hence, efficient solving of this subproblem is crucial but not a trivial task. For unconstrained problems, we found a closed form solution for the subproblem and studied the numerical behavior of OSGA in Ahookhosh (2016) and (Ahookhosh and Neumaier 2013, 2016). In Ahookhosh and Neumaier (2017), we gave one projection version of OSGA and provided a framework to solve the subproblem over simple convex domains or simple functional constraints. In particular, we describe a scheme to compute the global solution of the OSGA subproblem for bound constraints in Ahookhosh and Neumaier (2017). Let us emphasize that the subproblem of OSGA-O is constrained by a simple convex set and simple functional constraints, which is different from that one used in Ahookhosh (2016), Ahookhosh and Neumaier (2013), Ahookhosh and Neumaier (2016), Ahookhosh and Neumaier (2017), Ahookhosh and Neumaier (2017), which leads to solve a proximal-like problem.

The overall structure of this paper takes six sections, including this introductory section. In the next section, we briefly review the main idea of OSGA. In Sect. 3, we give a reformulation for the basic problem considered and show that solving the OSGA-O subproblem is equivalent to solving a proximal-like problem. Section 4 points out how the proximal-like subproblem can be solved in many interesting cases. Some numerical results are reported in Sect. 5, and conclusions are given in Sect. 6.

1.2 Preliminaries and notation

Let $\mathcal{V}$ be a finite-dimensional vector space endowed with the norm $\|\cdot\|$, and let $\mathcal{V}^*$ denotes its dual space, formed by all linear functional on $\mathcal{V}$ where the bilinear pairing $\langle g, x\rangle$ denotes the value of the functional $g \in \mathcal{V}^*$ at $x \in \mathcal{V}$. The associated dual norm of $\|\cdot\|$ is defined by

$$\|g\|_* = \sup_{z\in\mathcal{V}}\{\langle g, z\rangle : \|z\| \le 1\}.$$

If $\mathcal{V} = \mathbb{R}^n$, then, for $1 \le p \le \infty$,

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}, \quad \|x\|_{1,p} = \sum_{i=1}^{m} \|x_{g_i}\|_p,$$

where $x = (x_{g_1}, \ldots, x_{g_m}) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_m}$ in which $n_1 + \cdots + n_m = n$. We set $(x)_+ = \max(x, 0)$. For a function $f : \mathcal{V} \to \overline{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$,

$$\operatorname{dom} f = \{x \in \mathcal{V} \mid f(x) < +\infty\}$$

denotes its effective domain, and f is called proper if $\operatorname{dom} f \neq \emptyset$ and $f(x) > -\infty$ for all $x \in \mathcal{V}$. Let C be a subset of $\mathcal{V}$. In particular, if C is a box, we denote it by $\mathbf{x} = [\underline{x}, \overline{x}]$ in which $\underline{x}$ and $\overline{x}$ are the vectors of lower and upper bounds on the components of x, respectively. The vector $g \in \mathcal{V}^*$ is called a subgradient of f at x if $f(x) \in \mathbb{R}$ and

$$f(y) \ge f(x) + \langle g, y - x\rangle \quad \text{for all } y \in \mathcal{V}.$$

The set of all subgradients is called the subdifferential of f at x and is denoted by $\partial f(x)$. If $f : \mathcal{V} \to \mathbb{R}$ is nonsmooth and convex, then Fermat's optimality condition for the nonsmooth convex optimization problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in C \end{array}$$

is given by

$$0 \in \partial f(x) + N_C(x), \tag{1}$$

where $N_C(x)$ is the normal cone of C at x defined by

$$N_C(x) = \{p \in \mathcal{V} \mid \langle p, x - z\rangle \ge 0 \ \ \forall z \in C\}. \tag{2}$$

The proximal-like operator $\operatorname{prox}_{\lambda f}^{C}(y)$ is the unique optimizer of the optimization problem

$$\operatorname{prox}_{\lambda f}^{C}(y) := \operatorname*{argmin}_{x\in C} \frac{1}{2}\|x - y\|_2^2 + \lambda f(x), \tag{3}$$

where $\lambda > 0$. From (1), the first-order optimality condition of (3) is given by

$$0 \in x - y + \lambda \partial f(x) + N_C(x). \tag{4}$$

If $C = \mathcal{V}$, then (4) is simplified to

$$0 \in x - y + \lambda \partial f(x), \tag{5}$$

giving the classical proximity operator. A function f is called strongly convex with the convexity parameter $\sigma > 0$ if and only if

$$f(z) \geq f(x) + \langle g, z - x \rangle + \frac{\sigma}{2}\|z - x\|_2^2 \quad \text{for all } x, z \in \mathcal{V} \tag{6}$$

where g denotes any subgradient of f at x, i.e., $g \in \partial f(x)$.

The subdifferential of $\phi(x) = \|Wx\|$ is given in the next result, for an arbitrary norm $\|\cdot\|$ in $\mathbb{R}^n$ and a matrix $W \in \mathbb{R}^{m\times n}$. To observe a proof of this result, see Proposition 2.1.17 in Ahookhosh (2015).

Proposition 1 *Let $\phi : \mathbb{R}^n \to \mathbb{R}$, $\phi(x) = \|Wx\|$, where $W \in \mathbb{R}^{m\times n}$ is an invertible matrix and $\|\cdot\|$ is any norm of $\mathbb{R}^n$. Then*

$$\partial\phi(x) = \begin{cases} \{g \in \mathbb{R}^n \mid \|W^{-T}g\| \leq 1\} & \text{if } x = 0, \\ \{g \in \mathbb{R}^n \mid \|W^{-T}g\| = 1,\ \langle g, x\rangle = \|Wx\|\} & \text{if } x \neq 0. \end{cases}$$

In particular, if $\|\cdot\|$ is self-dual ($\|\cdot\| = \|\cdot\|_*$), we have

$$\partial\phi(x) = \begin{cases} \{g \in \mathbb{R}^n \mid \|W^{-T}g\|_* \leq 1\} & \text{if } x = 0, \\ W^T \frac{Wx}{\|Wx\|} & \text{if } x \neq 0. \end{cases}$$

In the next example, we show how Proposition 1 is applied to $\phi = \|\cdot\|_\infty$, which will be needed in Sect. 4. The subdifferential of other norms of $\mathbb{R}^n$ can be computed with Proposition 1 in the same way.

Example 2 We use Proposition 1 to derive the subdifferential of $\phi = \|\cdot\|_\infty$ at an arbitrary point x. We first recall that the dual norm of $\|\cdot\|_\infty$ is $\|\cdot\|_1$. If $x = 0$, Proposition 1 implies

$$\begin{aligned} \partial\phi(0) &= \{g \in \mathbb{R}^n \mid \|g\|_1 \leq 1\} \\ &= \left\{ g \in \mathbb{R}^n \mid g = \sum_{i=1}^n \beta_i e_i,\ \beta \in [-1, 1],\ \sum_{i=1}^n |\beta_i| \leq 1 \right\}, \end{aligned}$$

leading to

$$\begin{aligned} \partial\phi(x) &= \left\{ g \in \mathbb{R}^n \mid \|g\|_1 = 1, \langle g, x\rangle = \|x\|_\infty = \max_{1\leq i\leq n} |x_i| \right\} \\ &= \left\{ g \in \mathbb{R}^n \mid \sum_{j=1}^n |g_j| = 1, \sum_{j=1}^n g_j x_j = \|x\|_\infty \right\}. \end{aligned}$$

If $x \neq 0$, we set

$$\mathcal{I} := \{i \in \{1, \ldots, n\} \mid \|x\|_\infty = |x_i|\}$$

and we have $\|x\|_\infty = \sum_{i\in\mathcal{I}} \beta_i |x_i|$ and $\sum_{i\in\mathcal{I}} \beta_i = 1$ leading to

$$\partial\phi(x) = \left\{ g \in \mathbb{R}^n \mid g = \sum_{i\in\mathcal{I}} \beta_i \ \mathrm{sign}(x_i) e_i, \quad \sum_{i\in\mathcal{I}} \beta_i = 1 \right\}.$$

2 A review of optimal subgradient algorithm (OSGA)

In this section, we briefly review the main idea of the optimal subgradient algorithm (OSGA) proposed by Neumaier (2016). To this end, we first consider the convex constrained minimization problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in C, \end{array} \tag{7}$$

where $f : C \to \overline{\mathbb{R}}$ is a proper and convex function defined on a nonempty, closed, and convex subset C of $\mathcal{V}$. The aim is to derive a solution $\widehat{x} \in C$ using the first-order black-box information, i.e., function values and subgradients. OSGA (see Algorithm 2) is an optimal subgradient algorithm for the problem (7) that constructs a sequence of iterates whose related function values converge to the minimum with the optimal complexity. The primary objective is to monotonically reduce bounds on the error term $f(x_b) - \widehat{f}$ of the function values, where $\widehat{f} := f(\widehat{x})$ and x_b is the best known point.

In details, OSGA considers a linear relaxation of f at x defined by

$$f(x) \geq \gamma + \langle h, x\rangle \ \text{ for all } x \in C, \tag{8}$$

where $\gamma \in \mathbb{R}$ and $h \in \mathcal{V}^*$ and a continuously differentiable prox-function $Q : C \to \mathbb{R}$ satisfying (6) and

$$Q_0 := \inf_{x\in C} Q(x) > 0. \tag{9}$$

Moreover, OSGA requires an efficient routine for finding a maximizer $u := U(\gamma, h)$ and the optimal objective value $\eta := E(\gamma, h)$ of the auxiliary problem

$$\begin{array}{ll} \sup & E_{\gamma,h}(x) \\ \text{s.t.} & x \in C, \end{array} \tag{10}$$

where it is known that the supremum η is positive and the function $E_{\gamma,h} : C \to \mathbb{R}$ is defined by

$$E_{\gamma,h}(x) := -\frac{\gamma + \langle h, x\rangle}{Q(x)}, \tag{11}$$

with $\gamma \in \mathbb{R}$, $h \in \mathcal{V}^*$.

In Neumaier (2016), it is shown that OSGA attains the following bound on function values

$$0 \leq f(x_b) - \widehat{f} \leq \eta Q(\widehat{x}).$$

Hence, by decreasing the error factor η, the convergence to an ε-minimizer x_b is guaranteed by

$$0 \leq f(x_b) - \widehat{f} \leq \varepsilon,$$

for some target tolerance $\varepsilon > 0$. In Neumaier (2016), it is shown that the number of iterations to achieve this optimizer is $\mathcal{O}(\varepsilon^{-1/2})$ for smooth f with Lipschitz continuous gradients and $\mathcal{O}(\varepsilon^{-2})$ for Lipschitz continuous nonsmooth f, which are optimal in both cases, cf. (Nemirovsky and Yudin 1983). The algorithm does not need to know about the global Lipschitz parameters and has a low memory requirement. Hence, if the subproblem (10) can be solved efficiently, it is appropriate for solving large-scale problems. In the next section, we show that OSGA can solve some structured nonsmooth problems with the complexity $\mathcal{O}(\varepsilon^{-1/2})$. Moreover, it is shown that by selecting a suitable prox-function Q, the subproblem (10) can be solved efficiently for this class of problems.

As discussed in Neumaier (2016), to update the given parameters α, h, γ, η and u, OSGA uses the following scheme:

Algorithm 1: PUS (parameters updating scheme)

Input: $\delta,\ \alpha_{\max} \in]0, 1[,\ 0 < \kappa' \leq \kappa, \alpha, \eta, \bar{h}, \bar{\gamma}, \bar{\eta}, \bar{u}$;
Output: $\alpha,\ h,\ \gamma,\ \eta,\ u$;
1 **begin**
2 $R \leftarrow (\eta - \overline{\eta})/(\delta\alpha\eta)$;
3 **if** $R < 1$ **then**
4 $\overline{\alpha} = \alpha e^{-\kappa}$;
5 **else**
6 $\overline{\alpha} \leftarrow \min(\alpha e^{\kappa'(R-1)}, \alpha_{\max})$;
7 **end**
8 $\alpha \leftarrow \overline{\alpha}$;
9 **if** $\overline{\eta} < \eta$ **then**
10 $h \leftarrow \overline{h};\ \gamma \leftarrow \overline{\gamma};\ \eta \leftarrow \overline{\eta};\ u \leftarrow \overline{u}$;
11 **end**
12 **end**

If the best function value f_{x_b} is stored and updated, then each iteration of OSGA only requires the computation of two function values f_x and $f_{x'}$ (Lines 6 and 11) and one subgradient g_x (Line 6).

Algorithm 2: OSGA (optimal subgradient algorithm)

Input: global parameters: $\delta, \alpha_{\max} \in]0, 1[,\ 0 < \kappa' \leq \kappa;$ local parameters: $x_0, \mu \geq 0;$
Output: $x_b,\ f_{x_b};$

begin
  $x_b = x_0;$ compute f_{x_b} and $g_{x_b};$
  $h = g_{x_b} - \mu g_Q(x_b);\ \gamma = f_{x_b} - \mu Q(x_b) - \langle h, x_b \rangle;$
  $\gamma_b = \gamma - f_{x_b};\ u = U(\gamma_b, h);\ \eta = E(\gamma_b, h) - \mu;\ \alpha = \alpha_{\max};$
  while *stopping criteria do not hold* **do**
    $x = x_b + \alpha(u - x_b)$; compute f_x and $g_x;$
    $g = g_x - \mu g_Q(x);\ \overline{h} = h + \alpha(g - h);$
    $\overline{\gamma} = \gamma + \alpha(f_x - \mu Q(x) - \langle g, x \rangle - \gamma);$
    $x'_b = \operatorname{argmin}_{z \in \{x_b, x\}} f(z);\ f_{x'_b} = \min\{f_{x_b}, f_x\};$
    $\gamma'_b = \overline{\gamma} - f_{x'_b};\ u' = U(\gamma'_b, \overline{h});$
    $x' = x_b + \alpha(u' - x_b)$; compute $f_{x'};$
    choose $\overline{x}_b$ in such a way that $f_{\overline{x}_b} \leq \min\{f_{x'_b}, f_{x'}\};$
    $\overline{\gamma}_b = \overline{\gamma} - f_{\overline{x}_b};\ \overline{u} = U(\overline{\gamma}_b, \overline{h});\ \overline{\eta} = E(\overline{\gamma}_b, \overline{h}) - \mu;$
    $x_b = \overline{x}_b;\ \ f_{x_b} = f_{\overline{x}_b};$
    update the parameters α, h, γ, η and u using PUS;
  end
end

3 Structured nonsmooth convex optimization

Let us consider the convex constrained problem

$$\begin{array}{ll} \min & f(\mathcal{A}x, \phi(x)) \\ \text{s.t.} & x \in C, \end{array} \tag{12}$$

where $f : \mathcal{U} \times \mathbb{R} \to \mathbb{R}$ is a proper and convex function that is smooth with Lipschitz continuous gradients with respect to both arguments and monotone increasing with respect to the second argument, $\mathcal{A} : \mathcal{V} \to \mathcal{U}$ is a linear operator, $C \subseteq \mathcal{V}$ is a simple convex domain, and $\phi : \mathcal{V} \to \mathbb{R}$ is a simple nonsmooth, real-valued, and convex loss function. This class of convex problems generalizes the composite problem considered in Nesterov (2013, 2015). As discussed in Sect. 2, OSGA attains the complexity $\mathcal{O}(\varepsilon^{-2})$ for this class of problems. Hence we aim to reformulate the problem (12) in such a way that OSGA attains the complexity $\mathcal{O}(\varepsilon^{-1/2})$. We here reformulate the problem (12) in the form

$$\begin{array}{ll} \min & \widetilde{f}(x, \xi) \\ \text{s.t.} & (x, \xi) \in \widetilde{C}, \end{array} \tag{13}$$

where

$$\widetilde{f} : \mathcal{V} \times \mathbb{R} \to \mathbb{R}, \quad \widetilde{f}(x, \xi) := f(\mathcal{A}x, \xi), \tag{14}$$

$$\widetilde{C} := \{(x, \xi) \in \mathcal{V} \times \mathbb{R} \mid x \in C,\ \ \phi(x) \leq \xi\}. \tag{15}$$

By the assumptions on f, the reformulated function $\widetilde{f}$ is smooth and has Lipschitz continuous gradients. OSGA can handle the problems of the form (13) with the complexity $\mathcal{O}(\varepsilon^{-1/2})$ in the price of adding a functional constraint to the feasible domain

C. In the next subsection, we will show how OSGA can effectively handle (13) with the feasible domain $\widetilde{C}$. A version of OSGA that take advantages of the problem (13) is called OSGA-O.

Problems of the form (12) appears in many applications in the fields of signal and image processing, machine learning, statistics, economic, geophysics, and inverse problems. Let us consider the following example.

Example 3 (composite minimization) We consider the unconstrained minimization problem

$$\begin{array}{ll} \min & f(\mathcal{A}x) + \phi(x) \\ \text{s.t.} & x \in C, \end{array} \tag{16}$$

where $f : \mathcal{U} \to \overline{\mathbb{R}}$ is a smooth, proper, and convex function, $\mathcal{A} : \mathcal{V} \to \mathcal{U}$ is a linear operator, and $\phi : \mathcal{V} \to \mathbb{R}$ is a simple but nonsmooth, real-valued, and convex loss function. In this case, we reformulate (16) in the form (13) by setting $\widetilde{f}(x, \xi) := f(\mathcal{A}x) + \xi$. Let us now consider the linear inverse problem

$$y = \mathcal{A}x + v, \tag{17}$$

where $x \in \mathbb{R}^n$ is the original object, $y \in \mathbb{R}^m$ is an observation, and $v \in \mathbb{R}^m$ is an additive or impulsive noise. The objective is to recover x from y by solving (17). In practice, this problem is typically underdetermined and ill-conditioned, and v is unknown. Hence x typically is recovered by solving one of the minimization problems

$$\begin{array}{ll} \min & \frac{1}{2}\|y - \mathcal{A}x\|_2^2 + \frac{1}{2}\lambda\|x\|_2^2 \\ \text{s.t.} & x \in \mathbb{R}^n, \end{array} \tag{18}$$

$$\begin{array}{ll} \min & \frac{1}{2}\|y - \mathcal{A}x\|_2^2 + \lambda\|x\|_1 \\ \text{s.t.} & x \in \mathbb{R}^n, \end{array} \tag{19}$$

or

$$\begin{array}{ll} \min & \frac{1}{2}\|y - \mathcal{A}x\|_2^2 + \frac{1}{2}\lambda_1\|x\|_2^2 + \lambda_2\|x\|_1 \\ \text{s.t.} & x \in \mathbb{R}^n. \end{array} \tag{20}$$

These problems can be reformulated in the form (13) by setting

$$\widetilde{f}(x, \xi) := \frac{1}{2}\|y - \mathcal{A}x\|_2^2 + \xi, \quad \phi(x) := \frac{1}{2}\lambda\|x\|_2^2, \tag{21}$$

$$\widetilde{f}(x, \xi) := \frac{1}{2}\|y - \mathcal{A}x\|_2^2 + \xi, \quad \phi(x) := \lambda\|x\|_1, \tag{22}$$

or

$$\widetilde{f}(x, \xi) := \frac{1}{2}\|y - \mathcal{A}x\|_2^2 + \xi, \quad \phi(x) := \frac{1}{2}\lambda_1\|x\|_2^2 + \lambda_2\|x\|_1, \tag{23}$$

respectively.

3.1 New setup of optimal subgradient algorithm (OSGA-O)

This section describes the subproblem (10) for a problem of the form (13). To this end, we introduce some prox-function and employ it to derive an inexpensive solution of the subproblem. We generally assume that the domain C is simple enough such that η and $(\widehat{u}, \widetilde{u})$ can be computed cheaply, in $\mathcal{O}(n \log n)$ operations, say.

Let $Q : \mathcal{V} \times \mathbb{R} \to \mathbb{R}$ be a function defined by

$$Q(x, \widetilde{x}) := Q_0 + \frac{1}{2}\left(\|x\|_2^2 + \widetilde{x}^2\right), \tag{24}$$

where $Q_0 > 0$. From $g_Q(x, \widetilde{x}) = (x\ \widetilde{x})^T$, we obtain

$$\begin{aligned}
&Q(z, \widetilde{z}) + \langle g_Q(z, \widetilde{z}), (x - z, \widetilde{x} - \widetilde{z})\rangle + \frac{1}{2}\|(x - z, \widetilde{x} - \widetilde{z})^T\|_2^2 \\
&= Q_0 + \frac{1}{2}\left\langle (z, \widetilde{z})^T, (z, \widetilde{z})^T \right\rangle + \left\langle (z, \widetilde{z})^T, (x - z, \widetilde{x} - \widetilde{z})^T \right\rangle \\
&\quad + \frac{1}{2}\left\langle (x - z, \widetilde{x} - \widetilde{z})^T, (x - z, \widetilde{x} - \widetilde{z})^T \right\rangle \\
&= Q_0 + \frac{1}{2}\left\langle (z, \widetilde{z})^T, (x, \widetilde{x})^T \right\rangle + \frac{1}{2}\left\langle (x, \widetilde{x})^T, (x - z, \widetilde{x} - \widetilde{z})^T \right\rangle \\
&= Q_0 + \frac{1}{2}\left\langle (x, \widetilde{x})^T, (x, \widetilde{x})^T \right\rangle = Q_0 + \frac{1}{2}\left(\|x\|_2^2 + \widetilde{x}^2\right) \\
&= Q(x, \widetilde{x}).
\end{aligned}$$

This means that Q is a strongly convex function with the convexity parameter 1, and since $Q_0 > 0$, we get $Q(x, \widetilde{x}) > 0$. Then Q is strongly convex, and $Q(x, \widetilde{x}) > 0$. This shows that Q is a prox-function. We now replace the linear relaxation (8) by

$$\widetilde{f}(x, \widetilde{x}) \geq \gamma + \langle h, x\rangle + \widetilde{h}\widetilde{x} \ \text{ for all } x \in \widehat{C}. \tag{25}$$

Using this linear relaxation and the prox-function (24), the subproblem (10) is rewritten in the form

$$\begin{aligned} \sup\ & E_{\gamma,h,\widetilde{h}}(x, \widetilde{x}) \\ \text{s.t.}\ & (x, \widetilde{x}) \in \widetilde{C}, \end{aligned} \tag{26}$$

where $E_{\gamma,h,\widetilde{h}} : \mathcal{V} \times \mathbb{R} \to \mathbb{R}$ is differentiable and given by

$$E_{\gamma,h,\widetilde{h}}(x, \widetilde{x}) := -\frac{\gamma + \langle h, x\rangle + \widetilde{h}\widetilde{x}}{Q(x, \widetilde{x})}. \tag{27}$$

Let $(\widehat{u}, \widetilde{u}) \in \mathcal{V} \times \mathbb{R}$ be a maximizer of (26) and $\eta = E_{\gamma,h,\widetilde{h}}(\widehat{u}, \widetilde{u})$. The next result gives a bound on the error $\widetilde{f}(x_b, \widetilde{x}_b) - \widehat{f}$, which is important for providing the complexity analysis of OSGA-O.

Proposition 4 *Let $\gamma_b := \gamma - f(x_b, \widetilde{x}_b)$, $(\widehat{u}, \widetilde{u}) := U(\gamma_b, h, \widetilde{h})$, and $\eta := E(\gamma_b, h, \widetilde{h})$. Then, we have*

$$0 \le f(x_b, \widetilde{x}_b) - \widehat{f} \le \eta Q(\widehat{x}, x^*), \tag{28}$$

where $(\widehat{x}, x^*)$ is the solution of (13). In particular, if $(x_b, \widetilde{x}_b)$ is not yet optimal, then the choice $(\widehat{u}, \widetilde{u})$ implies $\eta = E(\gamma_b, h, \widetilde{h}) > 0$.

Proof Using (25), (26), and (27), this follows similarly to Proposition 2.1 in Neumaier (2016). □

Proposition 5 *The maximizer $(\widehat{u}, \widetilde{u})$ of (26) and the associated η satisfy*

$$\gamma + \langle h, \widehat{u}\rangle + \widetilde{h}\widetilde{u} = -\eta Q(\widehat{u}, \widetilde{u}), \tag{29}$$

$$\langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\widetilde{x} - \widetilde{u}) \ge 0 \quad \textit{for all } (x, \widetilde{x}) \in \widetilde{C}. \tag{30}$$

Proof The problem (26) and the definition (27) imply that the function $\zeta : C \times \mathbb{R} \to \mathbb{R}$ defined by

$$\zeta(x, \widetilde{x}) := \gamma + \langle h, x\rangle + \widetilde{h}\widetilde{x} + \eta Q(x, \widetilde{x})$$

is nonnegative and vanishes at $(x, \widetilde{x}) = (\widehat{u}, \widetilde{u})$, i.e., the identity (29) holds. Since $\zeta(x, \widetilde{x})$ is continuously differentiable with gradient $g_\zeta(x, \widetilde{x}) = (\eta\widehat{u} + h, \eta\widetilde{u} + \widetilde{h})^T$, the first order optimality condition holds, i.e.,

$$\langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\widetilde{x} - \widetilde{u}) \ge 0 \tag{31}$$

for all $(x, \widetilde{x}) \in \widetilde{C}$, giving the results. □

The subsequent result gives a systematic way for solving OSGA subproblem (26) for problems of the form (13).

Theorem 6 *Let $(\widehat{u}, \widetilde{u}) \in \mathcal{V} \times \mathbb{R}$ be a maximizer of (26) and $\eta = E_{\gamma,h,\widetilde{h}}(\widehat{u}, \widetilde{u})$. Then, for $y := -\eta^{-1}h$, $\lambda := \widetilde{u} + \eta^{-1}\widetilde{h}$, we have*

$$\widetilde{u} := \phi(\widehat{u}), \quad \widehat{u} := \operatorname*{argmin}_{x \in C} \frac{1}{2}\|x - y\|_2^2 + \lambda\phi(x). \tag{32}$$

Furthermore, η and λ can be computed by solving the two-dimensional system of equations

$$\begin{cases} \phi(\widehat{u}) + \eta^{-1}\widetilde{h} - \lambda = 0, \\ \eta\left(\frac{1}{2}(\|\widehat{u}\|_2^2 + \phi(\widehat{u})^2) + Q_0\right) + \gamma + \langle h, \widehat{u}\rangle + \widetilde{h}\phi(\widehat{u}) = 0. \end{cases} \tag{33}$$

Proof From Proposition 5, at the minimizer $(\widehat{u}, \widetilde{u})$, we obtain

$$\eta\left(\frac{1}{2}(\|\widehat{u}\|_2^2 + (\widetilde{u})^2) + Q_0\right) = -\gamma - \langle h, \widehat{u}\rangle - \widetilde{h}\widetilde{u} \tag{34}$$

and

$$\langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\widetilde{x} - \widetilde{u}) \geq 0 \quad \text{for all } (x, \widetilde{x}) \in C \times \mathbb{R},\ \phi(x) \leq \widetilde{x}. \tag{35}$$

We conclude the proof in the next two parts:

In the first part, considering $g_Q(\widehat{u}, \widetilde{u}) = (\widehat{u}^T, \widetilde{u})^T$, we show that (35) is equivalent to the following two inequalities

$$\begin{cases} \eta\widetilde{u} + \widetilde{h} \geq 0, \\ \\ \langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\phi(x) - \widetilde{u}) \geq 0 \quad \text{for all } (x, \widetilde{x}) \in C \times \mathbb{R}. \end{cases} \tag{36}$$

Assuming that these two inequalities hold, we prove (35). From $\phi(x) \leq \widetilde{x}$ and $\eta\widetilde{u}+\widetilde{h} \geq 0$, we obtain

$$\langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\widetilde{x} - \widetilde{u}) \geq \langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\phi(x) - \widetilde{u}) \geq 0.$$

We now assume (35) and prove (36). The inequality $\eta\widetilde{u} + \widetilde{h} \geq 0$ holds; otherwise, by selecting $\widetilde{x}$ large enough, we get

$$\langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\widetilde{x} - \widetilde{u}) < 0,$$

which is a contradiction with (35). Since $\phi(x) \leq \widetilde{x}$, the second inequality of (36) holds.

In the second part, by setting $x = \widehat{u}$ and $\widetilde{u} = \phi(\widehat{u})$, we see that $\widehat{u}$ is a solution of the minimization problem

$$\inf_{x \in C} \langle \eta\widehat{u} + h, x - \widehat{u}\rangle + (\eta\widetilde{u} + \widetilde{h})(\phi(x) - \widetilde{u}).$$

The first-order optimality condition (1) of this problem leads to

$$0 \in \widehat{u} + \eta^{-1}h + (\widetilde{u} + \eta^{-1}\widetilde{h})\ \partial\phi(\widehat{u}) + N_C(\widehat{u}). \tag{37}$$

On the other hand, by writing the first-order optimality condition (4) for the problem

$$\begin{array}{ll} \min & \dfrac{1}{2}\|x - y\|_2^2 + \lambda\phi(x) \\ \text{s.t.} & x \in C, \end{array}$$

we get

$$0 \in \widehat{u} - y + \lambda\ \partial\phi(\widehat{u}) + N_C(\widehat{u}). \tag{38}$$

By comparing (37) and (38) and setting $y = -\eta^{-1}h$, $\lambda = \widetilde{u} + \eta^{-1}\widetilde{h}$, we conclude that both problems have the same minimizer $\widehat{u}$. Since $\widetilde{u} = \phi(\widehat{u})$, we obtain

$$\lambda = \widetilde{u} + \eta^{-1}\widetilde{h} = \phi(\widehat{u}) + \eta^{-1}\widetilde{h}.$$

Using this and substituting $\widetilde{u} = \phi(\widehat{u})$ in (34), η and λ are found by solving the system of nonlinear equations (33). This completes the proof. □

In Theorem 6, if $C = \mathcal{V}$, the problem (32) is reduced to the classical proximity operator $\widehat{u} = \text{prox}_{\lambda\phi}(y)$ defined in (3). Hence, the problem (32) is called *proximal-like*. Therefore, the word "simple" in the definition of C means that the problem (32) can be solved efficiently either in a closed form or by an inexpensive iterative scheme. To have a clear view of Theorem 6, we give the following example.

Example 7 Let us consider the ℓ_1-regularized least squares problem (19). Then, the problem can be reformulated as

$$\begin{aligned} \min \ & \frac{1}{2}\|y - Ax\|_2^2 + \xi \\ \text{s.t.} \ & \|x\|_1 \le \xi. \end{aligned}$$

Since $\phi = \|\cdot\|_1$, the solution of (32) is $\widehat{u} = \text{sign}(y_i)(|y_i| - \lambda)_+$ with $y = -\eta^{-1}h$ (see Table 6.1 in Ahookhosh (2015)). Substituting this into (33) gives

$$\begin{cases} f_1(\eta,\lambda) := \displaystyle\sum_{i=1}^{n}(|y_i| - \lambda)_+ + \eta^{-1}\widetilde{h} - \lambda = 0, \\ f_2(\eta,\lambda) := \eta\left(\dfrac{1}{2}\left(\displaystyle\sum_{i=1}^{n}(|y_i| - \lambda)_+^2 + \left(\sum_{i=1}^{n}(|y_i| - \lambda)_+\right)^2\right) + Q_0\right) \\ \qquad\qquad +\gamma + \sum_{i=1}^{n}(h_i + \widetilde{h})(|y_i| - \lambda)_+ = 0. \end{cases}$$

This is a two-dimensional system of nonsmooth equations that can be reformulated as a nonlinear least squares problem; see, e.g., (Pang and Qi 1993).

Theorem 6 leads to the two-dimensional nonlinear system

$$F(\eta,\lambda) := (f_1(\eta,\lambda),\, f_2(\eta,\lambda))^T = 0, \tag{39}$$

where

$$\begin{aligned} f_1(\eta,\lambda) &:= \phi(\widehat{u}) + \eta^{-1}\widetilde{h} - \lambda, \\ f_2(\eta,\lambda) &:= \eta\left(\frac{1}{2}(\|\widehat{u}\|_2^2 + \phi(\widehat{u})^2) + Q_0\right) + \gamma + \langle h, \widehat{u}\rangle + \widetilde{h}\phi(\widehat{u}), \end{aligned}$$

in which $\widehat{u}$ and $\eta, \lambda > 0$. For instance, in Example 7, see the definition of $f_1(\eta,\lambda)$ and $f_2(\eta,\lambda)$. The system of nonsmooth equations (39) can be handled by replacing the vector (η,λ) with $(|\eta|, |\lambda|)$ and solving

$$\begin{array}{ll} \min & \dfrac{1}{2}\|F(|\eta|, |\lambda|)\|_2^2 \\ \text{s.t.} & \eta, \lambda \in \mathbb{R} \end{array} \tag{40}$$

if $f_1(\eta, \lambda)$ and $f_2(\eta, \lambda)$ are nonsmooth. The problems (39) and (40), such as Example 7, can be solved by semismooth Newton methods or smoothing Newton methods (Qi and Sun 1999), quasi-Newton methods (Sun and Han 1997; Li et al. 2001), secant methods (Potra et al. 1998), and trust-region methods (Ahookhosh et al. 2015; Qi 1995).

In view of Theorem 6, we now provide a systematic way for solving OSGA-O subproblem (26), which is summarized in next scheme.

Algorithm 3: SUS (subproblem solver for OSGA-O)

Input: Q_0, γ, h;
Output: u, η;
1 **begin**
2 solve the system of nonlinear equation (39) approximately by a nonlinear solver to find η and λ;
3 set $u = (\widehat{u}, \phi(\widehat{u}))$.
4 **end**

To implement Algorithm 3 (SUS), we need a reliable nonlinear solver to deal with the system of nonlinear equation (39) and a routine giving the solution of the proximal-like problem (32) effectively. In Sect. 4, we investigate solving the proximal-like problem (32) for some practically important loss functions ϕ. Algorithm 2 requires two solutions of the subproblem (26) (u in Line 6 and u' in Line 10) that are provided by Line 3 of SUS (similar notation can be considered for u').

3.2 Convergence analysis and complexity

In this section, we establish the complexity bounds of OSGA-O for Lipschitz continuous nonsmooth problems and smooth problems with Lipschitz continuous gradients. We also show that if $\widetilde{f}$ is strictly convex, the sequence generated by OSGA-O is convergent to $\widehat{x}$.

To guarantee the existence of a minimizer for OSGA-O, we assume the following conditions.

(H1) The objective function $\widetilde{f}$ is proper and convex;
(H2) The upper level set $N_{\widetilde{f}}(x_0, \widetilde{x}_0) := \{x \in \widetilde{C} \mid \widetilde{f}(x, \widetilde{x}) \leq \widetilde{f}(x_0, \widetilde{x}_0)\}$ is bounded, for the starting point $(x_0, \widetilde{x}_0) \in \mathcal{V} \times \mathbb{R}$.

Since $\widetilde{f}$ is convex, the upper level set $N_{\widetilde{f}}(x_0, \widetilde{x}_0)$ is closed, and $\mathcal{V} \times \mathbb{R}$ is a finite-dimensional vector space, (H2) implies that the upper level set $N_{\widetilde{f}}(x_0, \widetilde{x}_0)$ is convex and compact. It follows from the continuity and properness of the objective function $\widetilde{f}$ that it attains its global minimizer on the upper level set $N_{\widetilde{f}}(x_0, \widetilde{x}_0)$. Therefore, there is at least one minimizer $(\widehat{x}, x^*)$.

Since the underlying problem (13) is a special case of the problem (7) considered by Neumaier (2016), the complexity results of OSGA-O is the same as OSGA.

Theorem 8 *Suppose that $\widetilde{f} - \mu Q$ is convex and $\mu \geq 0$. Then we have*

(i) *(Nonsmooth complexity bound) If the points generated by Algorithm 2 stay in a bounded region of the interior of $\widetilde{C}$, or if $\widetilde{f}$ is Lipschitz continuous in $\widetilde{C}$, the total number of iterations needed to reach a point with $\widetilde{f}(x,\widetilde{x}) \leq \widetilde{f}(\widehat{x},x^*) + \varepsilon$ is at most $\mathcal{O}((\varepsilon^2+\mu\varepsilon)^{-1})$. Thus the asymptotic worst case complexity is $\mathcal{O}(\varepsilon^{-2})$ when $\mu = 0$ and $\mathcal{O}(\varepsilon^{-1})$ when $\mu > 0$.*

(ii) *(Smooth complexity bound) If $\widetilde{f}$ has Lipschitz continuous gradients with Lipschitz constant L, the total number of iterations needed by Algorithm 2 to reach a point with $\widetilde{f}(x,\widetilde{x}) \leq \widetilde{f}(\widehat{x},x^*) + \varepsilon$ is at most $\mathcal{O}(\varepsilon^{-1/2})$ if $\mu = 0$, and at most $\mathcal{O}(|\log\varepsilon|\sqrt{L/\mu})$ if $\mu > 0$.*

Proof Since all assumptions of Theorems 4.1 and 4.2, Propositions 5.2 and 5.3, and Theorem 5.1 of Neumaier (2016) are satisfied, the results remain valid. □

Indeed, if a nonsmooth problem can be reformulated as (13) with a nonsmooth loss function ϕ, then OSGA-O can solve the reformulated problem with the complexity $\mathcal{O}(\varepsilon^{-1/2})$ for an arbitrary accuracy parameter ε. The next result shows that the sequence $\{(x_k,\widetilde{x}_k)\}$ generated by OSGA-O is convergent to $(\widehat{x},x^*)$ if the objective $\widetilde{f}$ is strictly convex and $(\widehat{x},x^*) \in \operatorname{int}\widetilde{C}$, where $\operatorname{int}\widetilde{C}$ denotes the interior of $\widetilde{C}$.

Proposition 9 *Suppose that $\widetilde{f}$ is strictly convex, then the sequence $\{(x_k,\widetilde{x}_k)\}$ generated by OSGA-O is convergent to $(\widehat{x},x^*)$ if $(\widehat{x},x^*) \in \operatorname{int}\widetilde{C}$.*

Proof Since $\widetilde{f}$ is strictly convex, the minimizer $(\widehat{x},x^*)$ is unique. By $(\widehat{x},x^*) \in \operatorname{int}\widetilde{C}$, there exists a small $\delta > 0$ such that the neighborhood

$$N(\widehat{x},x^*) := \{(x,\widetilde{x}) \in \widetilde{C} \mid \|(x,\widetilde{x}) - (\widehat{x},x^*)\| \leq \delta\}$$

is contained in $\widetilde{C}$ and it is a convex and compact set. Let $(x_\delta,\widetilde{x}_\delta)$ be a minimizer of the problem

$$\begin{aligned} &\min \quad \widetilde{f}(x,\widetilde{x}) \\ &\text{s.t.} \quad (x,\widetilde{x}) \in \partial N(\widehat{x},x^*), \end{aligned} \tag{41}$$

where $\partial N(\widehat{x},x^*)$ denotes the boundary of $N(\widehat{x},x^*)$. Set $\varepsilon_\delta := \widetilde{f}(x_\delta,\widetilde{x}_\delta) - \widehat{f}$ and consider the upper level set

$$N_{\widetilde{f}}(x_\delta,\widetilde{x}_\delta) := \{(x,\widetilde{x}) \in \widetilde{C} \mid \widetilde{f}(x,\widetilde{x}) \leq \widetilde{f}(x,\widetilde{x}) = \widehat{f} + \varepsilon_\delta\}.$$

Now, Theorem 8 implies that the algorithm attains an ε_δ-solution of (13) in a finite number κ of iterations. Hence, after κ_1 iterations, the best point $(x_b,\widetilde{x}_b)$ attained by OSGA-O satisfies $\widetilde{f}(x_b,\widetilde{x}_b) \leq \widehat{f} + \varepsilon_\delta$, i.e., $(x_b,\widetilde{x}_b) \in N_{\widetilde{f}}(x_\delta,\widetilde{x}_\delta)$. We now show that $N_{\widetilde{f}}(x_\delta,\widetilde{x}_\delta) \subseteq N(\widehat{x},x^*)$. To prove this statement by contradiction, we suppose that there exists $(x,\widetilde{x}) \in N_{\widetilde{f}}(x_\delta,\widetilde{x}_\delta) \setminus N(\widehat{x},x^*)$. Since $(x,\widetilde{x}) \notin N(\widehat{x},x^*)$, we have $\|(x,\widetilde{x}) - (\widehat{x},x^*)\| > \delta$. Therefore, there exists λ_0 such that

$$\|\lambda_0(x,\widetilde{x}) + (1-\lambda_0)(\widehat{x},x^*)\| = \delta.$$

It follows from (41), the strictly convex property of $\widetilde{f}$, and $\widetilde{f}(x,\widetilde{x}) \leq \widetilde{f}(x_\delta,\widetilde{x}_\delta)$ that

$$\begin{aligned}\widetilde{f}(x_\delta,\widetilde{x}_\delta) &\leq \widetilde{f}(\lambda_0(x,\widetilde{x})+(1-\lambda_0)(\widehat{x},x^*)) < \lambda_0\widetilde{f}(x,\widetilde{x})+(1-\lambda_0)\widetilde{f}(\widehat{x},x^*)\\ &\leq \lambda_0\widetilde{f}(x_\delta,\widetilde{x}_\delta)+(1-\lambda_0)\widetilde{f}(x_\delta,\widetilde{x}_\delta)=\widetilde{f}(x_\delta,\widetilde{x}_\delta),\end{aligned}$$

which is a contradiction, i.e., $N_{\widetilde{f}}(x_\delta,\widetilde{x}_\delta) \subseteq N(\widehat{x},x^*)$ implying $(x,\widetilde{x}) \in N(\widehat{x},x^*)$, giving the result. □

4 Solving proximal-like subproblem

In this section, we show that the proximal-like problem (32) can be solved in a closed form for many special cases appearing in applications. To this end, we first consider unconstrained problems ($C = \mathcal{V}$) and study some problems with simple constrained domains ($C \neq \mathcal{V}$). Although finding proximal points is a mature area in convex nonsmooth optimization (cf. (Combettes and Pesquet 2011; Parikh and Boyd 2013)), we here address the solution of several proximal-like problems of the form (32) appearing in applications that to the best of our knowledge have not been studied in literature.

4.1 Unconstrained examples ($C = \mathcal{V}$)

We here consider several interesting unconstrained proximal problems appearing in applications and explain how the associated OSGA-O auxiliary problem (32) can be solved.

In recent years, the interest of applying regularizations with weighted norms is increased by emerging many applications; see, e.g., (Daubechies et al. 2010; Rauhut and Ward 2016). Let d be a vector in $\mathbb{R}^n$ such that $d_i \neq 0$ for $i = 1,\ldots,n$. Then, we define the weight matrix $D := \mathrm{diag}(d)$, which is a diagonal matrix with $D_{i,i} = d_i$ for $i = 1,\ldots,n$. It is clear that D is an invertible matrix. The next two results show how to compute a solution of the problem (32) for special cases of ϕ arising frequently in applications.

Proposition 10 *Let $D := diag(d)$, where $d \in \mathbb{R}^n$ with $d_i \neq 0$, for $i = 1,\ldots,n$. If $\phi(x) = \|Dx\|_1$, the proximity operator (32) is given by*

$$\big(prox_{\lambda\phi}(y)\big)_i = sign(y_i)(|y_i| - \lambda|d_i|)_+, \tag{42}$$

for $i = 1,\ldots,n$.

Proof See Proposition 6.2.1 in Ahookhosh (2015). □

Proposition 11 *Let $D := diag(d)$, where $d \in \mathbb{R}^n$ and $d_i \neq 0$, for $i = 1,\ldots,n$. If $\phi(x) = \|Dx\|_2$, the proximity operator (32) is given by $prox_{\lambda\phi}(y) = 0$ if $\|D^{-1}y\|_2 \leq \lambda$ and otherwise, for $i = 1,\ldots,n$,*

$$\big(prox_{\lambda\phi}(y)\big)_i = \frac{\tau y_i}{\tau + \lambda d_i^2},$$

where τ is the unique solution of the one-dimensional nonlinear equation

$$\sum_{i=1}^{n} \frac{d_i^2 y_i^2}{(\tau + \lambda d_i^2)^2} - 1 = 0.$$

Proof The optimality condition (5) shows that $u = \text{prox}_{\lambda\phi}(y)$ if and only if

$$0 \in u - y + \lambda\, \partial \|Du\|_2. \tag{43}$$

We consider two cases:

(i) $\|D^{-1}y\|_2 \leq \lambda$; (ii) $\|D^{-1}y\|_2 > \lambda$.

Case (i). Let $\|D^{-1}y\|_2 \leq \lambda$. Then, we show that $u = 0$ satisfies (43). If $u = 0$, Proposition 1 implies $\partial\phi(0) = \{g \in \mathcal{V}^* \mid \|D^{-1}g\|_2 \leq 1\}$. Using this, we get that $u = 0$ is satisfied in (43) if $y \in \{g \in \mathcal{V}^* \mid \|D^{-1}g\|_2 \leq \lambda\}$ leading to $\text{prox}_{\lambda\phi}(y) = 0$.

Case (ii). Let $\|D^{-1}y\|_2 > \lambda$. Case (i) implies $u \neq 0$. Proposition 1 implies $\partial\phi(u) = D^T Du/\|Du\|_2$, and the optimality condition (5) yields

$$u - y + \lambda\, D^T \frac{Du}{\|Du\|_2} = 0.$$

By this and setting $\tau = \|Du\|_2$, we get

$$\left(1 + \frac{\lambda d_i^2}{\tau}\right) u_i - y_i = 0,$$

leading to

$$u_i = \frac{\tau y_i}{\tau + \lambda d_i^2},$$

for $i = 1, \ldots, n$. Substituting this into $\tau = \|Du\|_2$ implies

$$\sum_{i=1}^{n} \frac{d_i^2 y_i^2}{(\tau + \lambda d_i^2)^2} = 1.$$

We define the function $\psi :]0, +\infty[\to \mathbb{R}$ by

$$\psi(\tau) := \sum_{i=1}^{n} \frac{d_i^2 y_i^2}{(\tau + \lambda d_i^2)^2} - 1,$$

where it is clear that ψ is decreasing and

$$\lim_{\tau \to 0} \psi(\tau) = \frac{1}{\lambda^2} \sum_{i=1}^{n} \frac{y_i^2}{d_i^2} - 1 = \frac{1}{\lambda^2} \left(\|D^{-1}y\|_2^2 - \lambda^2 \right), \quad \lim_{\tau \to +\infty} \psi(\tau) = -1.$$

It can be deduced by $\|D^{-1}y\|_2 > \lambda$ and the mean value theorem that there exists $\widehat{\tau} \in]0, +\infty[$ such that $\psi(\widehat{\tau}) = 0$, giving the result. □

We here emphasize that if $D = I$ (I denotes the identity matrix) then the proximity operator for $\phi(\cdot) = \|\cdot\|_2$ is given by

$$\text{prox}_{\lambda\phi}(y) = (1 - \lambda/\|y\|_2)_+ y,$$

cf. (Parikh and Boyd 2013). If one solves the equation $\psi(\tau) = 0$ approximately, and an initial interval $[a, b]$ is available such that $\psi(a)\psi(b) < 0$, then a solution can be computed to an ε-accuracy using the bisection scheme in $\mathcal{O}(\log_2((b-a)/\varepsilon))$ iterations; see, e.g., (Neumaier 2001). However, it is preferable to use a more sophisticated zero finder like the secant bisection scheme (Algorithm 5.2.6, (Neumaier 2001)). If an interval $[a, b]$ with sign change is available, one can also use MATLAB `fzero` function combining the bisection scheme, the inverse quadratic interpolation, and the secant method.

Grouped variables typically appear in high-dimensional statistical learning problems. For example, in data mining applications, categorical features are encoded by a set of dummy variables forming a group. Another interesting example is learning sparse additive models in statistical inference, where each component function can be represented using basis expansions and thus can be treated as a group. For such problems (see (Liu et al. 2010) and references therein), it is more natural to select groups of variables instead of individual ones when a sparse model is preferred.

In the following two results, we show how the proximity operator $\text{prox}_{\lambda\phi}(\cdot)$ can be computed for the mixed-norms $\phi(\cdot) = \|\cdot\|_{1,2}$ and $\phi(\cdot) = \|\cdot\|_{1,\infty}$, which are especially important in the context of sparse optimization and sparse recovery with grouped variables.

Proposition 12 *Let $\phi(\cdot) = \|\cdot\|_{1,2}$. Then, the proximity operator (32) is given by*

$$(prox_{\lambda\phi}(y))_{g_i} = \left(1 - \frac{\lambda}{\|y_{g_i}\|_2}\right)_+ y_{g_i}. \tag{44}$$

for $i = 1, \ldots, m$.

Proof Since $u = (u_{g_1}, \ldots, u_{g_m}) \in \mathbb{R}^{n_1} \times \ldots \times \mathbb{R}^{n_m}$ and ϕ is separable with respect to the grouped variables, we fix the index $i \in \{1, \ldots, m\}$. The optimality condition (5) shows that $u_{g_i} = \text{prox}_{\lambda\phi}(y_{g_i})$ if and only if

$$0 \in u_{g_i} - y_{g_i} + \lambda\, \partial\|u_{g_i}\|_2, \tag{45}$$

for $i = 1, \ldots, m$. We now consider two cases: (i) $\|y_{g_i}\|_2 \leq \lambda$; (ii) $\|y_{g_i}\|_2 > \lambda$.

Case (i). Let $\|y_{g_i}\|_2 \leq \lambda$. Then, we show that $u_{g_i} = 0$ satisfies (45). If $u_{g_i} = 0$, Proposition 1 implies $\partial\phi(0_{g_i}) = \{g \in \mathbb{R}^{n_i} \mid \|g_{g_i}\|_2 \leq 1\}$. By substituting this into (45), we get that $u_{g_i} = 0$ is satisfied in (45) if $y_{g_i} \in \{g \in \mathbb{R}^{n_i} \mid \|g_{g_i}\|_2 \leq \lambda\}$, which leads to $\text{prox}_{\lambda\phi}(y_{g_i}) = 0_{g_i}$. Since the right hand side of (44) is also zero, (44) holds.

Case (ii). Let $\|y_{g_i}\|_2 > \lambda$. Then, Case (i) implies that $u_{g_i} \neq 0$. From Proposition 1, we obtain

$$\partial_\phi(u_{g_i}) = \left\{ \frac{u_{g_i}}{\|u_{g_i}\|_2} \right\}, \tag{46}$$

where $i = 1, \ldots, m$ and $\|y_{g_i}\|_2 > \lambda$. Then (45) and (46) imply

$$u_{g_i} - y_{g_i} + \lambda \frac{u_{g_i}}{\|u_{g_i}\|_2} = 0,$$

leading to

$$\left(1 + \frac{\lambda}{\|u_{g_i}\|_2}\right) u_{g_i} = y_{g_i}$$

giving $u_{g_i} = \mu_i y_{g_i}$. Substituting this into the previous identity and solving it with respect to μ_i yield

$$\mu_i = \left(1 - \frac{\lambda}{\|y_{g_i}\|_2}\right)_+ y_{g_i}, \quad u_{g_i} = \mu_i y_{g_i},$$

completing the proof. □

Proposition 13 *Let $\phi(\cdot) = \|\cdot\|_{1,\infty}$. Then, the proximity operator (32) is given by*

$$(prox_{\lambda\phi}(y_{g_i}))^j_{g_i} = \begin{cases} 0 & \text{if } \|y_{g_i}\|_1 \leq \lambda, \\ sign(y^j_{g_i}) u^i_\infty & \text{if } \|y_{g_i}\|_1 > \lambda, \ j \in \mathcal{I}_{g_i}, \\ y^j_{g_i} & \text{if } \|y_{g_i}\|_1 > \lambda, \ j \notin \mathcal{I}_{g_i}, \end{cases} \tag{47}$$

for $i = 1, \ldots, m$, where

$$u^i_\infty := \frac{1}{\widehat{k}_i} \left(\sum_{j \in \mathcal{I}_{g_i}} |y^j_{g_i}| - \lambda \right) \tag{48}$$

with

$$\mathcal{I}_{g_i} := \{l^1_{g_i}, \ldots, l^{\widehat{k}_i}_{g_i}\} \tag{49}$$

in which $\widehat{k}_i$ is the smallest $k \in \{1, \ldots, n_i - 1\}$ such that

$$\frac{1}{\widehat{k}_i} \left(\sum_{j=1}^{\widehat{k}_i} v^j_{g_i} - \lambda \right) \geq v^{\widehat{k}_i+1}_{g_i}, \tag{50}$$

where $v^j_{g_i} := |y_{l^j_{g_i}}|$ and $l^1_{g_i}, \ldots, l^{n_i}_{g_i}$ is a permutation of $1, \ldots, n_i$ such that $v^1_{g_i} \geq v^2_{g_i} \geq \ldots \geq v^{n_i}_{g_i}$. If (59) is not satisfied for $k \in \{1, \ldots, n_i - 1\}$, then $\widehat{k}_i = n_i$, for $i = 1, \ldots, m$.

Proof Since $u = (u_{g_1}, \ldots, u_{g_m}) \in \mathbb{R}^{n_1} \times \ldots \times \mathbb{R}^{n_m}$ and ϕ is separable with respect to the grouped variables, we fix the index $i \in \{1, \ldots, m\}$. The optimality condition (5) shows that $u_{g_i} = \text{prox}_{\lambda\phi}(y_{g_i})$ if and only if

$$0 \in u_{g_i} - y_{g_i} + \lambda\, \partial \|u_{g_i}\|_\infty. \tag{51}$$

We now consider two cases: (i) $\|y_{g_i}\|_1 \leq \lambda$; (ii) $\|y_{g_i}\|_1 > \lambda$.

Case (i). Let $\|y_{g_i}\|_1 \leq \lambda$. Then, we show that $u_{g_i} = 0$ satisfies (51). If $u_{g_i} = 0$, the subdifferential of ϕ derived in Example 2 is $\partial\phi(0_{g_i}) = \{g \in \mathbb{R}^{n_i} \mid \|g\|_1 \leq 1\}$. By substituting this into (51), we get that $u_{g_i} = 0$ satisfies (51) if $y_{g_i} \in \{g \in \mathbb{R}^{n_i} \mid \|g\|_1 \leq 1\}$, i.e., $\text{prox}_{\lambda\phi}(y_{g_i}) = 0_{g_i}$.

Case (ii). Let $\|y_{g_i}\|_1 > \lambda$. From Case (i), we have $u_{g_i} \neq 0$. We show that

$$u^j_{g_i} = \begin{cases} \text{sign}(y^j_{g_i})u^i_\infty & \text{if } i \in \mathcal{I}_{g_i}, \\ y^j_{g_i} & \text{otherwise,} \end{cases} \tag{52}$$

with $\mathcal{I}_{g_i}$ defined in (49), satisfies (51). Hence, using the subdifferential of ϕ derived in Example 2, there exist coefficients $\beta^j_{g_i}$, for $j \in \mathcal{I}_{g_i}$, such that

$$u_{g_i} - y_{g_i} + \lambda \sum_{j \in \mathcal{I}_{g_i}} \beta^j_{g_i}\, \text{sign}(u^j_{g_i}) e_j = 0, \tag{53}$$

where

$$\beta^j_{g_i} \geq 0 \;\; j \in \mathcal{I}_{g_i}, \quad \sum_{j \in \mathcal{I}_{g_i}} \beta^j_{g_i} = 1. \tag{54}$$

Let u_{g_i} be the vector defined in (52). We define

$$\beta^j_{g_i} := \frac{|y^j_{g_i}| - u^i_\infty}{\lambda}, \tag{55}$$

for $j \in \mathcal{I}_{g_i} = \{l^i_1, \ldots, l^i_{\widehat{k}_i}\}$ with u^i_∞ defined in (48). We show that the choice (55) satisfies (53). We first show $u^i_\infty > 0$. It follows from (48) and (50) if $\widehat{k}_i < n$ and from $\|y_{g_i}\|_1 > \lambda$ if $\widehat{k}_i = n$. Using (52) and (55), we come to

$$\begin{aligned} u^j_{g_i} - y^j_{g_i} + \lambda\beta^j_{g_i}\, \text{sign}(u^j_{g_i}) &= \text{sign}(y^j_{g_i})u^i_\infty - y^j_{g_i} + \left(|y^j_{g_i}| - u^i_\infty\right) \text{sign}(\text{sign}(y^j_{g_i})u^i_\infty) \\ &= \text{sign}(y^j_{g_i})u^i_\infty - y^j_{g_i} + \left(|y^j_{g_i}| - u^i_\infty\right) \text{sign}(y^j_{g_i}) = 0, \end{aligned}$$

for $j \in \mathcal{I}_{g_i}$. For $j \notin \mathcal{I}^j_{g_i}$, we have $u^j_{g_i} - y^j_{g_i} = 0$. Hence, (53) is satisfied componentwise. It remains to show that (54) holds. From (50), we have

that $|y_{g_i}^j| \geq u_\infty^i$, for $j \in \mathcal{I}_{g_i}$. This and (55) yield $\beta_{g_i}^j \geq 0$ for $j \in \mathcal{I}_{g_i}$. It can be deduced from (48) that

$$\sum_{j=1}^{\widehat{k}_i} \beta_{g_i}^j = \frac{1}{\lambda} \sum_{j=1}^{\widehat{k}_i} |y_{g_i}^j| - \frac{\widehat{k}^i}{\lambda} u_\infty^i = \frac{1}{\lambda} \sum_{j=1}^{\widehat{k}_i} |y_{g_i}^j| - \frac{1}{\lambda} \left(\sum_{j=1}^{\widehat{k}_i} |y_{g_i}^j| - \lambda \right) = 1,$$

giving the results. □

Corollary 14 *Let $\phi(\cdot) = \|\cdot\|_\infty$. Then, the proximity operator (32) is given by*

$$(prox_{\lambda\phi}(y))_i = \begin{cases} 0 & \text{if } \|y\|_1 \leq \lambda, \\ sign(y_i) u_\infty & \text{if } \|y\|_1 > \lambda,\ i \in \mathcal{I}, \\ y_i & \text{if } \|y\|_1 > \lambda,\ i \notin \mathcal{I}, \end{cases} \tag{56}$$

for $i = 1, \ldots, n$, where

$$u_\infty := \frac{1}{\widehat{k}} \left(\sum_{i \in \mathcal{I}} |y_i| - \lambda \right) \tag{57}$$

with

$$\mathcal{I} := \{l_1, \ldots, l_{\widehat{k}}\} \tag{58}$$

in which $\widehat{k}$ is the smallest $k \in \{1, \ldots, n-1\}$ such that

$$\frac{1}{\widehat{k}} \left(\sum_{i=1}^{\widehat{k}} v_i - \lambda \right) \geq v_{\widehat{k}+1}, \tag{59}$$

where $v_i := |y_{l_i}|$ and $l_1, \ldots, l_n$ is a permutation of $1, \ldots, n$ such that $v_1 \geq v_2 \geq \cdots \geq v_n$. If (59) is not satisfied for $k \in \{1, \ldots, n-1\}$, then $\widehat{k} = n$.

Proof The proof is straightforward from Proposition 13 by setting $m = 1$, $n_1 = n$, $y_{g_1} = y$, and $\mathcal{I}_{g_1} = \mathcal{I}$.

4.2 Constrained examples ($C \neq \mathcal{V}$)

In this section, we consider the subproblem (32) and show how it can be solved for some ϕ and C. More precisely, we solve the minimization problem

$$\begin{aligned} \min\ & \frac{1}{2}\|x - y\|_2^2 + \lambda\phi(x) \\ \text{s.t.}\ & x \in C, \end{aligned}$$

where $\phi(x)$ is a simple convex function and C is a simple domain. We consider a few examples of this form.

Proposition 15 *Let $\phi(x) = \|Dx\|_1$ and $C = [\underline{x}, \overline{x}]$, where D is a diagonal matrix. Then, the global minimizer of the subproblem (32) is given by*

$$(prox^C_{\lambda\phi}(y))_i = \begin{cases} \underline{x}_i & \text{if } \omega(y,\lambda) > 0,\ \underline{x}_i - y_i + \lambda|d_i|\, sign(\underline{x}_i) \geq 0, \\ \overline{x}_i & \text{if } \omega(y,\lambda) > 0,\ \overline{x}_i - y_i + \lambda|d_i|\, sign(\overline{x}_i) \leq 0, \\ y_i - \lambda|d_i| & \text{if } \omega(y,\lambda) > 0,\ y_i > \lambda|d_i|, \\ y_i + \lambda|d_i| & \text{if } \omega(y,\lambda) > 0,\ y_i < -\lambda|d_i|, \\ 0 & \text{otherwise}, \end{cases} \tag{60}$$

for $i = 1, \ldots, n$, where

$$\omega(y,\lambda) := \sum_{y_i + \lambda|d_i| < 0} (y_i + \lambda|d_i|)\underline{x} + \sum_{y_i + \lambda|d_i| > 0} (y_i + \lambda|d_i|)\overline{x}. \tag{61}$$

Proof The optimality condition (4) shows that $u = \text{prox}^C_{\lambda\phi}(y)$ if and only if

$$0 \in u - y + \lambda\, \partial\|Du\|_1 + N_C(u), \tag{62}$$

where $N_C(u)$ is the normal cone of C at u defined in (2). We show that $u = 0$ if and only if $\omega(y,\lambda) \leq 0$. We first consider that

$$N_C(0) = \left\{ p \in \mathcal{V} \;\middle|\; \forall z \in [\underline{x}, \overline{x}],\ \langle p, z\rangle \leq 0 \right\} = \left\{ p \in \mathcal{V} \mid \sum_{p_i < 0} p_i \underline{x} + \sum_{p_i > 0} p_i \overline{x} \leq 0 \right\}. \tag{63}$$

(62) suggests $u = 0$ if and only if there exists $p \in N_C(0) \cap (y - \lambda\partial\phi(x))$. By Proposition 1, this is possible if and only if

$$\min \left\{ \sum_{p_i < 0} p_i \underline{x} + \sum_{p_i > 0} p_i \overline{x} \;\middle|\; p = \lambda g,\ \|D^{-1} g\|_\infty \leq 1 \right\} \leq 0.$$

The solution of this problem is $p = y - \lambda|D\mathbf{1}|$, where $\mathbf{1}$ is the vector of all ones. Hence, the minimum of this problem is given by (61). This implies $u = 0$ if and only if $\omega(y,\lambda) \leq 0$. We, therefore, consider two cases:

Case (i). $u = 0$. Then, we have $\omega(y,\lambda) \leq 0$.
Case (ii). $u \neq 0$. Then, $\omega(y,\lambda) > \lambda$. Proposition 1 yields

$$\partial\phi(u) = \{g \in \mathcal{V}^* \mid \|D^{-1} g\|_\infty = 1,\ \ \langle g, u\rangle = \|Du\|_1\},$$

leading to

$$\sum_{i=1}^{n} (g_i u_i - |d_i u_i|) = 0.$$

By induction on nonzero elements of u, we get $g_i u_i = |d_i u_i|$, for $i = 1, \ldots, n$. This implies that $g_i = |d_i| \operatorname{sign}(u_i)$ if $u_i \neq 0$. This and the definition of $N_C(u)$ yield

$$u_i - y_i + \lambda(\partial \|Du\|_1)_i \begin{cases} \geq 0 & \text{if } u_i = \underline{x}_i, \\ \leq 0 & \text{if } u_i = \overline{x}_i, \\ = 0 & \text{if } \underline{x}_i < u_i < \overline{x}_i, \end{cases}$$

for $i = 1, \ldots, n$, and equivalently for $u \neq 0$, we get

$$u_i - y_i + \lambda |d_i| \operatorname{sign}(u_i) \begin{cases} \geq 0 & \text{if } u_i = \underline{x}_i, \\ \leq 0 & \text{if } u_i = \overline{x}_i, \\ = 0 & \text{if } \underline{x}_i < u_i < \overline{x}_i, \end{cases} \tag{64}$$

for $i = 1, \ldots, n$. If $u_i = \underline{x}_i$, substituting $u_i = \underline{x}_i$ in (64) implies $\underline{x}_i - y_i + \lambda |d_i| \operatorname{sign}(\underline{x}_i) \geq 0$. If $u_i = \overline{x}_i$, substituting $u_i = \overline{x}_i$ in (64) gives $\overline{x}_i - y_i + \lambda |d_i| \operatorname{sign}(\overline{x}_i) \leq 0$. If $\underline{x}_i < u_i < \overline{x}_i$, there are three possibilities: (a) $u_i > 0$; (b) $u_i < 0$; (c) $u_i = 0$. In Case (a), $\operatorname{sign}(u_i) = 1$ and (64) lead to $u_i = y_i - \lambda |d_i| > 0$. In Case (b), $\operatorname{sign}(u_i) = -1$ and (64) imply $u_i = y_i + \lambda |d_i| < 0$. In Case (c), we end up to $u_i = 0$, completing the proof. □

Proposition 16 *Let $\phi(x) = \frac{1}{2}\|x\|_2^2$ and $C = [\underline{x}, \overline{x}]$. Then, the global minimizer of the subproblem (32) is given by*

$$(prox_{\lambda\phi}^{C}(y))_i = \begin{cases} \underline{x}_i & \text{if } (1+\lambda)\underline{x}_i \geq y_i, \\ \overline{x}_i & \text{if } (1+\lambda)\overline{x}_i \leq y_i, \\ y_i/(1+\lambda) & \text{if } \underline{x}_i < y_i/(1+\lambda) < \overline{x}_i, \end{cases} \tag{65}$$

for $i = 1, \ldots, n$.

Proof The function $\phi(x) = \frac{1}{2}\|x\|_2^2$ is differentiable, i.e.,

$$\partial\phi(x) = x.$$

This and the definition of $N_C(u)$ imply

$$u_i - y_i + \lambda u_i \begin{cases} \geq 0 & \text{if } u_i = \underline{x}_i, \\ \leq 0 & \text{if } u_i = \overline{x}_i, \\ = 0 & \text{if } \underline{x}_i < u_i < \overline{x}_i, \end{cases} \tag{66}$$

for $i = 1, \ldots, n$. If $u_i = \underline{x}_i$, substituting $u_i = \underline{x}_i$ in (66) implies $(1+\lambda)\underline{x}_i \geq y_i$. If $u_i = \overline{x}_i$, substituting $u_i = \overline{x}_i$ in (66) yields $(1+\lambda)\overline{x}_i \leq y_i$. If $\underline{x}_i < u_i < \overline{x}_i$, then $u_i = y_i/(1+\lambda)$, giving the result. □

Proposition 17 *Let* $\phi(x) = \frac{1}{2}\lambda_1\|x\|_2^2 + \lambda_2\|Dx\|_1$ *and* $C = [\underline{x}, \overline{x}]$. *Then the global minimizer of the subproblem* (32) *is determined by*

$$(prox_{\lambda\phi}^{C}(y))_i = \begin{cases} \underline{x}_i & \text{if } \omega(y,\lambda) > 0,\ (1+\lambda_1)\underline{x}_i - y_i + \lambda_2|d_i|\, sign(\underline{x}_i) \geq 0, \\ \overline{x}_i & \text{if } \omega(y,\lambda) > 0,\ (1+\lambda_1)\overline{x}_i - y_i + \lambda_2|d_i|\, sign(\overline{x}_i) \leq 0, \\ 1/(1+\lambda_1)(y_i - \lambda_2|d_i|) & \text{if } \omega(y,\lambda) > 0,\ y_i > \lambda_2|d_i|, \\ 1/(1+\lambda_1)(y_i + \lambda_2|d_i|) & \text{if } \omega(y,\lambda) > 0,\ y_i < -\lambda_2|d_i|, \\ 0 & \text{otherwise}, \end{cases} \tag{67}$$

for $i = 1, \ldots, n$, *where* $\omega(y,\lambda)$ *is defined by* (61).

Proof Since $\mathcal{V}$ is finite-dimensional and $\text{dom}\left(\frac{1}{2}\lambda_1\|x\|_2^2\right) \cap \text{dom}\lambda_2\|Dx\|_1 \neq \emptyset$, we get

$$\partial\left(\frac{1}{2}\lambda_1\|x\|_2^2 + \lambda_2\|Dx\|_1\right) = \lambda_1\partial\left(\frac{1}{2}\|x\|_2^2\right) + \lambda_2\partial\left(\|Dx\|_1\right). \tag{68}$$

The optimality condition (4) shows that $u = \text{prox}_{\lambda\phi}^{C}(y)$ if and only if

$$0 \in u - y + \lambda_1 u + \lambda_2\, \partial\|Du\|_1 + N_C(u). \tag{69}$$

By (69), we have $u = 0$ if and only if there exists $p \in N_C(0) \cap (y - \lambda_2\partial\phi(x))$, where $N_C(0)$ is defined by (63). By Proposition 1, this is possible if and only if

$$\min\left\{\sum_{p_i<0} p_i\underline{x} + \sum_{p_i>0} p_i\overline{x} \;\middle|\; p = \lambda_2 g,\ \|D^{-1}g\|_\infty \leq 1\right\} \leq 0.$$

The solution of this problem is $p = y - \lambda_2|D\mathbf{1}|$, where $\mathbf{1}$ is the vector of all ones. Hence the minimum of this problem is given by (61). This implies $u = 0$ if and only if $\omega(y,\lambda_2) \leq 0$. We, therefore, consider two cases:

Case (i). $u = 0$. Then, we have $\omega(y,\lambda_2) \leq 0$.

Case (ii). $u \neq 0$. Then, $\omega(y,\lambda_2) > 0$. From (68) and the definition of $N_C(u)$, we obtain

$$u_i - y_i + \lambda_1 u_i + \lambda_2\partial|d_i u_i| \begin{cases} \geq 0 & \text{if } u_i = \underline{x}_i, \\ \leq 0 & \text{if } u_i = \overline{x}_i, \\ = 0 & \text{if } \underline{x}_i < u_i < \overline{x}_i, \end{cases}$$

for $i = 1, \ldots, n$. This leads to

$$(1+\lambda_1)u_i - y_i + \lambda_2|d_i|\,\text{sign}(u_i) \begin{cases} \geq 0 & \text{if } u_i = \underline{x}_i, \\ \leq 0 & \text{if } u_i = \overline{x}_i, \\ = 0 & \text{if } \underline{x}_i < u_i < \overline{x}_i, \end{cases} \tag{70}$$

for $i = 1, \ldots, n$. If $u_i = \underline{x}_i$, substituting $u_i = \underline{x}_i$ in (64) gives $(1 + \lambda_1)\underline{x}_i - y_i + \lambda_2|d_i|\,\text{sign}(\underline{x}_i) \geq 0$. If $u_i = \overline{x}_i$, substituting $u_i = \overline{x}_i$ in (64) implies $(1+\lambda_1)\overline{x}_i - y_i + \lambda_2|d_i|\,\text{sign}(\overline{x}_i) \leq 0$. If $\overline{x}_i < u_i < \overline{x}_i$, there are three possibilities: (i) $u_i > 0$; (ii) $u_i < 0$; (iii) $u_i = 0$. In Case (i), $\text{sign}(u_i) = 1$ and (64) imply $u_i = 1/(1+\lambda_1)(y_i - \lambda_2|d_i|) > 0$. In Case

(ii), $\text{sign}(u_i) = -1$ and (64) imply $u_i = 1/(1+\lambda_1)(y_i + \lambda_2|d_i|) < 0$. In Case (iii), we get $u_i = 0$, giving the result. □

Let $x \geq 0$ be nonnegativity constraints. These constraints are important in many applications, especially if x describes physical quantities; see, e.g., (Esser et al. 2013; Kaufman and Neumaier 1996, 1997). Since nonnegativity constraints can be regarded as an especial case of bound-constrained domain, Propositions 15, 16, and 17 can be used to derive the results for nonnegativity constraints.

5 Numerical experiments and application

We here report some numerical results to compare the performance of OSGA-O with OSGA and some state-of-the-art methods. In our comparison, we consider PGA (proximal gradient algorithm (Parikh and Boyd 2013)), NSDSG (nonsummable diminishing subgradient algorithm (Boyd et al. 2003)), FISTA (Beck and Teboulle's fast proximal gradient algorithm (Beck and Teboulle 2012)), NESCO (Nesterov's composite optimal algorithm (Nesterov 2013)), NESUN (Nesterov's universal gradient algorithm (Nesterov 2015)), NES83 (Nesterov's 1983 optimal algorithm (Nesterov 1983)), NESCS (Nesterov's constant step optimal algorithm (Nesterov 2004)), and NES05 (Nesterov's 2005 optimal algorithm (Nesterov 2005a)). We adapt NES83, NESCS, and NES05 by passing a subgradient in the place of the gradient to be able to apply them to nonsmooth problems (see Ahookhosh (2016)). The codes of these algorithms are written in MATLAB, where we use the parameters proposed in the associated papers.

5.1 Experiment with random data

We consider solving an underdetermined system

$$Ax = y, \tag{71}$$

where $A \in \mathbb{R}^{m\times n}$ $(m < n)$ and $y \in \mathbb{R}^m$. Underdetermined systems of linear equations have frequently appeared in many applications of linear inverse problem such as those in the fields of signal and image processing, geophysics, economics, machine learning, and statistics. The objective is to recover x from the observed vector y, and matrix A by some optimization models. Due to the ill-conditioned feature of the problem, the most popular optimization models are (18), (19), and (20), where (18) is smooth and (19) and (20) are nonsmooth. In Sect. 5.1.1, we report numerical results with the ℓ_1 minimization (19), and in Sect. 5.1.2, we give results regarding the elastic net minimization problem (20). We set $m = 5000$ and $n = 10000$, and the data A, y, and x_0 for problem (19) is randomly generated by

$$\texttt{A} = \texttt{rand(m, n)}, \quad \texttt{y} = \texttt{rand(1, m)}, \quad \texttt{x}_0 = \texttt{rand(1, n)},$$

where `rand` generates uniformly distributed random numbers between 0 and 1 and x_0 is a starting point for algorithms.

We divide the solvers into two classes: (i) proximal-based methods (PGA, FISTA, NESCO, and NESUN) that can be directly applied to nonsmooth problems; (ii) Subgradient-based methods (NSDSG, NES83, NESCS, and NES05) in which the nonsmooth first-order oracle is required, where NES83, NESCS, and NES05 are adapted to take a subgradient in the place of the gradient. We set

$$\widehat{L} := \max_{1\le i\le n} \|a_i\|^2,$$

where a_i $(i = 1, 2, \ldots, n)$ is the i-th column of A. In the implementation, NESCS, NES05, PGA, and FISTA use $L = 10^4\widehat{L}$, and NSDSG employs $\alpha_0 = 10^{-7}$. Algorithm 1, for both OSGA and OSGA-O, uses the parameters

$$\delta = 0.9, \quad \alpha_{max} = 0.7, \quad \kappa = \kappa' = 0.5,$$

and the prox-function (24) with $Q_0 = \frac{1}{2}\|x_0\|_2 + \epsilon$, where ϵ is the machine precision. All numerical experiments were executed on a PC Intel Core i7-3770 CPU 3.40GHz 8 GB RAM. To solve the nonlinear system of equations (33), we first consider the nonlinear least-squares problem (40) and solve it by the MATLAB internal function `fminsearch`,[1] which is a derivative-free solver handling both smooth and nonsmooth problems. In our implementation, we apply OSGA-O to the problem, stop it after 100 iterations and save the best function value attained (f_s), and run the others until either the same function value is achieved or the number of iterations reaches 5000. In our comparison, N_i and T denote the total number of iterations and the running time, respectively.

To display the results, we used the Dolan and Moré performance profile (Dolan and Moré 2002) with the performance measures N_i and T. In this procedure, the performance of each algorithm is measured by the ratio of its computational outcome versus the best numerical outcome of all algorithms. This performance profile offers a tool to statistically compare the performance of algorithms. Let $\mathcal{S}$ be a set of all algorithms and $\mathcal{P}$ be a set of test problems. For each problem p and algorithm s, $t_{p,s}$ denotes the computational outcome with respect to the performance index, which is used in the definition of the performance ratio

$$r_{p,s} := \frac{t_{p,s}}{\min\{t_{p,s} : s \in \mathcal{S}\}}. \tag{72}$$

If an algorithm s fails to solve a problem p, the procedure sets $r_{p,s} := r_{\text{failed}}$, where r_{failed} should be strictly larger than any performance ratio (72). Let n_p be the number of problems in the experiment. For any factor $\tau \in \mathbb{R}$, the overall performance of an algorithm s is given by

$$\rho_s(\tau) := \frac{1}{n_p}\text{size}\{p \in \mathcal{P} : r_{p,s} \le \tau\}.$$

[1] The function `fminsearch` is a derivative-free solver for unconstrained optimization problems based on Nelder–Mead simplex direct search method performing well for two-dimensional problems; see, e.g., (Hansen et al. 2010; Lagarias et al. 1998).

Here, $\rho_s(\tau)$ is the probability that a performance ratio $r_{p,s}$ of an algorithm $s \in \mathcal{S}$ is within a factor τ of the best possible ratio. The function $\rho_s(\tau)$ is a distribution function for the performance ratio. In particular, $\rho_s(1)$ gives the probability that an algorithm s wins over all other considered algorithms, and $\lim_{\tau \to r_{\text{failed}}} \rho_s(\tau)$ gives the probability that algorithm s solves all considered problems. Therefore, this performance profile can be considered as a measure of efficiency among all considered algorithms. In the following figures of this section, the number τ is represented in the x-axis, while $P(r_{p,s} \leq \tau : 1 \leq s \leq n_s)$ is shown in the y-axis.

5.1.1 ℓ_1 minimization

Here, we consider the ℓ_1 minimization problem (19), reformulate it as a minimization problem of the form (13) with the objective and the constraint given in (22), and solve the reformulated problem by OSGA-O. We then report some numerical results and a comparison among OSGA-O, OSGA and some state-of-the-art methods. For OSGA-O and OSGA, we here set $\mu = 0$.

Let us consider 6 different regularization parameters, apply PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O to (19) with 10 generated random data for each regularization parameter, and report numerical results in Table 1. Then, we use NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O for solving (19) with 10 generated random data corresponding to each regularization parameter and report numerical results in Table 2. We give a comparison among these algorithms in Fig. 1 for all 60 problems with the performance profile of N_i and T. We illustrate function values versus iterations for both classes of solvers with the regularization parameters $\lambda = 1,\ 10^{-1},\ 10^{-2},\ 10^{-3},\ 10^{-4},\ 10^{-5}$ in Fig. 2.

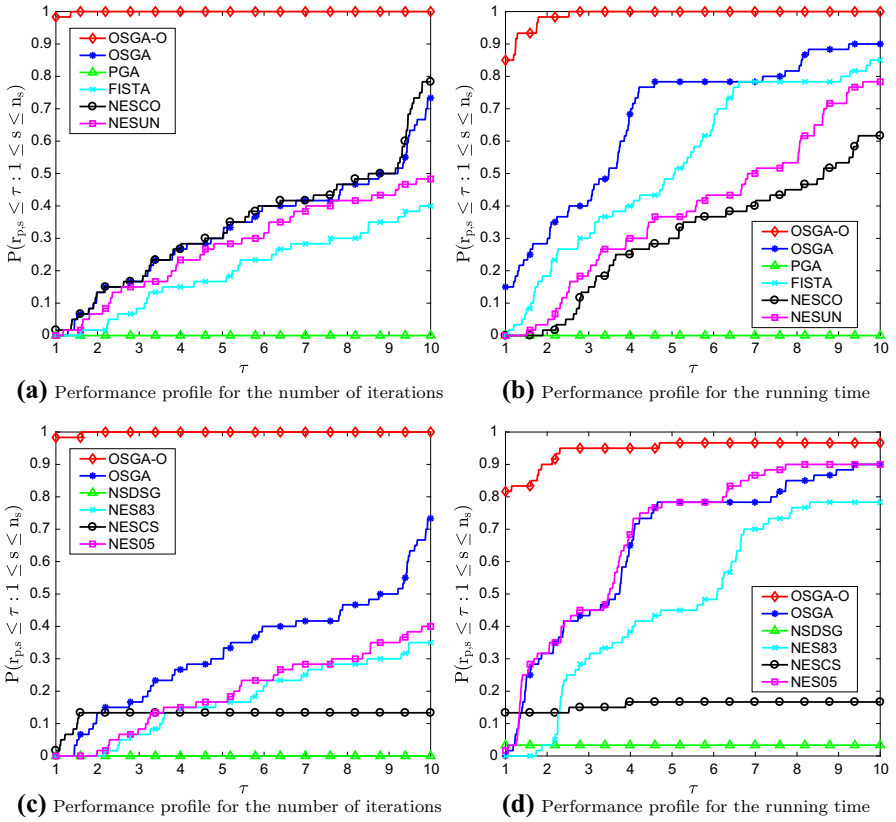
The results of Tables 1 and 2 show that OSGA-O attains the best number of iterations and running time for the ℓ_1 minimization problem, where the average of 10 implementations associated to each regularization parameter is given in these tables. In Fig. 1, subfigures (a) and (b) stand for performance profiles with measures N_i and T comparing proximal-based methods, where OSGA-O outperforms the others substantially. In this figure, subfigures (c) and (d) display performance profiles for measures

Table 1 Averages (only integer part) of N_i and T for PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O for solving ℓ_1 minimization problem with several regularization parameters

Reg. Par.	OSGA-O		OSGA		PGA		FISTA		NESCO		NESUN	
	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T
$\lambda = 1$	100	12	2277	103	5000	138	3316	138	3189	385	2614	223
$\lambda = 10^{-1}$	100	12	1497	72	4680	141	1940	87	1162	153	1376	125
$\lambda = 10^{-2}$	100	12	638	31	5000	156	1024	48	617	85	735	69
$\lambda = 10^{-3}$	100	12	773	38	5000	154	1241	60	749	102	890	85
$\lambda = 10^{-4}$	100	12	783	36	5000	138	1287	51	775	87	922	73
$\lambda = 10^{-5}$	100	12	462	22	5000	148	744	34	450	59	536	49

Table 2 Averages (only integer part) of N_i and T for NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O for solving ℓ_1 minimization problem with several regularization parameters

Reg. Par.	OSGA-O		OSGA		NSDSG		NES83		NESCS		NES05	
	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T
$\lambda = 1$	100	12	2277	103	5000	162	3352	169	4508	224	3318	106
$\lambda = 10^{-1}$	100	12	1497	72	5000	152	2167	105	4021	193	1947	60
$\lambda = 10^{-2}$	100	12	638	31	5000	138	1142	46	3956	167	1029	26
$\lambda = 10^{-3}$	100	12	773	38	5000	148	1386	62	4482	200	1248	35
$\lambda = 10^{-4}$	100	12	783	36	5000	142	1434	64	4949	216	1229	37
$\lambda = 10^{-5}$	100	12	462	22	5000	150	831	37	3572	161	749	21



(a) Performance profile for the number of iterations

(b) Performance profile for the running time

(c) Performance profile for the number of iterations

(d) Performance profile for the running time

Fig. 1 Performance profiles for the number of iterations N_i and the running time T for the ℓ_1 minimization problem: **a**, **b** display the results for N_i and T of PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O; **c**, **d**, respectively, illustrate the results for N_i and T of NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O. In all of these subfigures OSGA-O attains the best results with respect to both measures N_i and T

N_i and T to compare subgradient-based methods, where OSGA-O performs much better than the others with respect to both measures. Further, from Fig. 2, it can be seen that the worst results are obtained by NSDSG and PGA, while FISTA, NESCO, NESUN, NES83, NESCS, NES05, and OSGA are comparable to some extent; however, OSGA-O is significantly superior to the other methods.

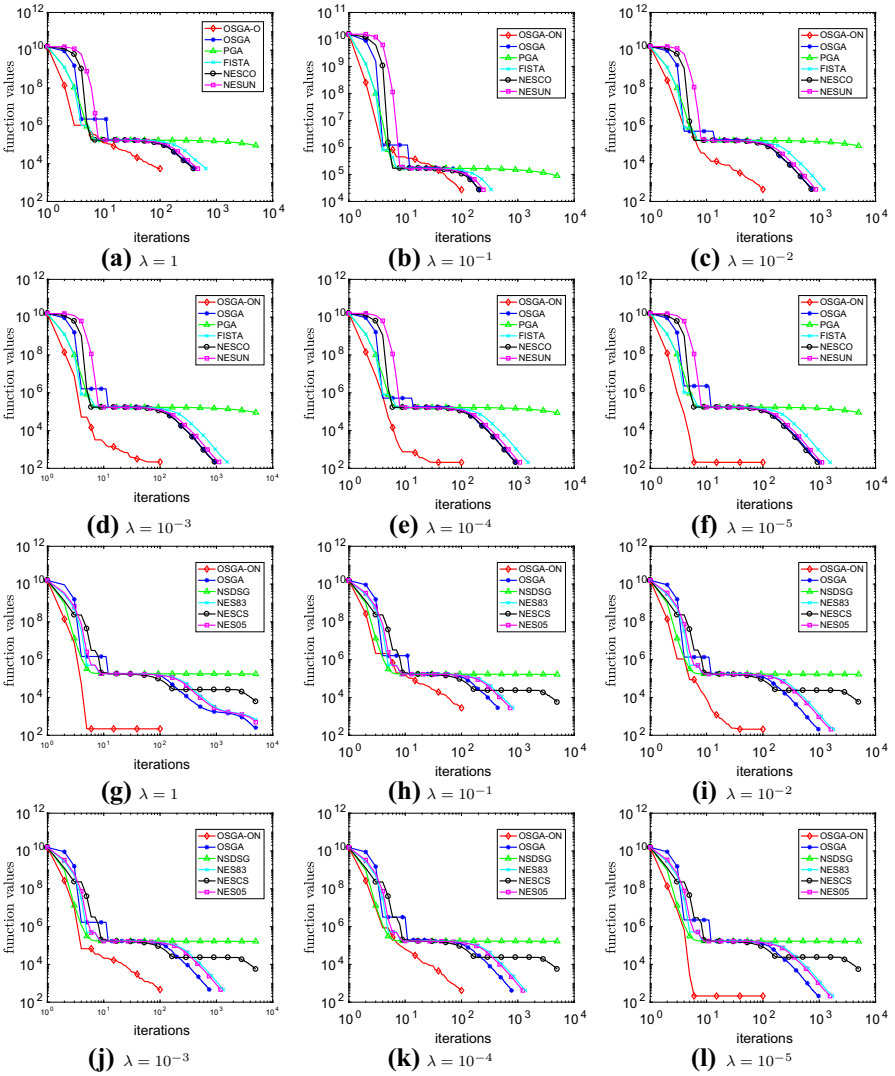


Fig. 2 A comparison among first-order methods for solving ℓ_1 minimization problem: **a–f** illustrate a comparison of function values versus iterations among PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O for $\lambda = 1,\ 10^{-1},\ 10^{-2},\ 10^{-3}, 10^{-4},\ 10^{-5}$, respectively; **g–l** display a comparison of function values versus iterations among NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O for $\lambda = 1,\ 10^{-1},\ 10^{-2},\ 10^{-3}, 10^{-4},\ 10^{-5}$, respectively

5.1.2 Elastic net minimization

We now consider the elastic net minimization problem (20), reformulate it as a minimization problem of the form (13) with the objective and the constraint given in (23), and solve the reformulated problem by OSGA-O. We then give some numerical results and a comparison among OSGA-O,OSGA and some state-of-the-art solvers. For OSGA-O and OSGA, we here set $\mu = \lambda_1/2$.

Let us consider six different regularization parameters $\lambda_1 = \lambda_2 = 1,\ 10^{-1},\ 10^{-2},\ 10^{-3},\ 10^{-4},\ 10^{-5}$. For each of these parameters, we generate the random data 10 times and report numerical results of PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O in Table 3 and numerical results of NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O in Table 4. For these 60 problems, we illustrate the performance profile for the measures N_i and T in Fig. 3. We then display function values versus iterations for both classes of solvers with $\lambda_1 = \lambda_2 = 1,\ 10^{-1},\ 10^{-2},\ 10^{-3},\ 10^{-4},\ 10^{-5}$ in Fig. 4.

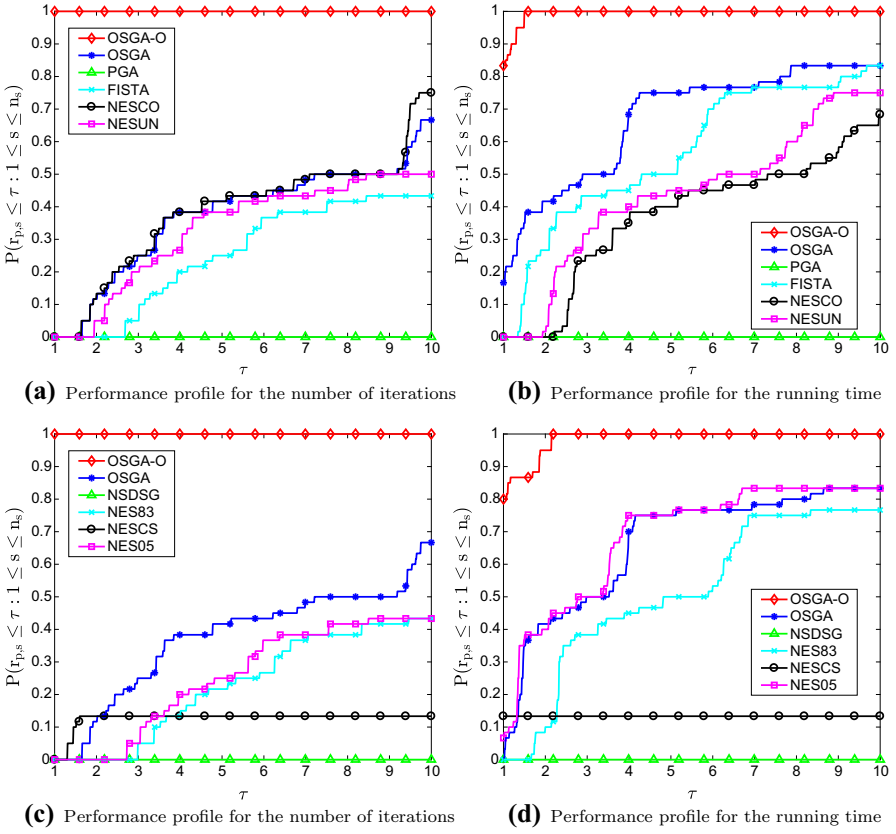
The results of Tables 3 and 4 show that the best number of iterations (N_i) and running time (T) are obtained by OSGA-O. From the results of Fig. 3, it can be seen that OSGA-O outperforms the others considerably with respect to N_i and T for both proximal-type and subgradient-type methods. It is also clear that the second best algorithm is OSGA.

Table 3 Averages (only integer part) of N_i and T for PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O for solving the elastic net problem (19) with several regularization parameters

Reg. Par.	OSGA-O		OSGA		PGA		FISTA		NESCO		NESUN	
	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T
$\lambda = 1$	100	12	4781	222	5000	143	4904	215	4756	609	4071	371
$\lambda = 10^{-1}$	100	12	1128	52	5000	143	1517	66	908	110	1078	94
$\lambda = 10^{-2}$	100	12	652	31	5000	148	1038	45	626	78	744	63
$\lambda = 10^{-3}$	100	12	474	23	5000	151	762	33	460	55	549	48
$\lambda = 10^{-4}$	100	12	513	25	5000	147	839	37	506	62	602	54
$\lambda = 10^{-5}$	100	12	661	32	5000	148	1076	47	649	79	772	68

Table 4 Averages (only integer part) of N_i and T for NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O for solving the elastic net problem with several regularization parameters

Reg. Par.	OSGA-O		OSGA		NSDSG		NES83		NESCS		NES05	
	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T	N_i	T
$\lambda = 1$	100	12	4781	222	5000	147	4949	221	5000	226	4904	145
$\lambda = 10^{-1}$	100	12	1128	52	5000	156	1692	80	4677	218	1523	47
$\lambda = 10^{-2}$	100	12	652	31	5000	146	1158	50	4225	182	1044	28
$\lambda = 10^{-3}$	100	12	474	23	5000	144	8516	36	3058	130	766	20
$\lambda = 10^{-4}$	100	12	513	25	5000	144	935	40	3120	135	844	23
$\lambda = 10^{-5}$	100	12	661	32	5000	148	1203	52	4589	201	1083	29



(a) Performance profile for the number of iterations **(b)** Performance profile for the running time

(c) Performance profile for the number of iterations **(d)** Performance profile for the running time

Fig. 3 Performance profiles for the number of iterations N_i and the running time T for the elastic net problem: **a**, **b** display the results for N_i and T of PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O; **c**, **d**, respectively, illustrate the results for N_i and T of NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O. In all of these subfigures OSGA-O attains the best results with respect to both measures N_i and T

In Fig. 4, we can see that the worst results are obtained by NSDSG and PGA, while FISTA, NESCO, NESUN, NES83, NESCS, NES05 and OSGA behave competitively. Further, OSGA-O performs better than the others significantly.

5.2 Sparse recovery (compressed sensing)

In recent years, there has been an increasing interest in finding sparse solutions of many problems using the structured models in various areas of applied mathematics. In most cases, the problem involves high-dimensional data with a small number of available measurements, where the core of these problems involves an optimization problem of the form (19) or (20). Thanks to the sparsity of solutions and the structure of problems, these optimization problems can be solved in reasonable time even for the extremely high-dimensional data sets. Sparse recovery, basis pursuit, lasso, wavelet-based deconvolution, and compressed sensing are some examples, where the latter case receives lots of attentions during the recent years, cf. (Candés 2006; Donoho 2006).

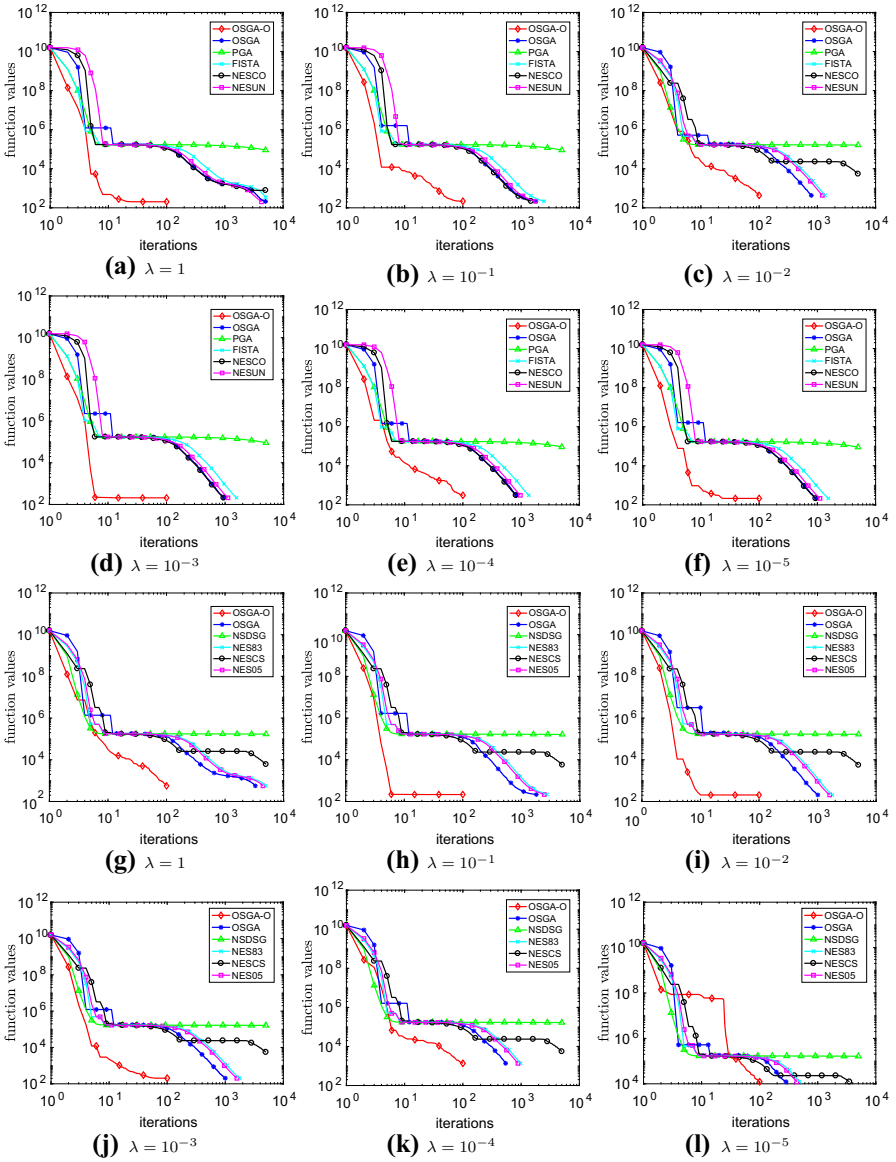


Fig. 4 A comparison among first-order methods for solving the elastic net problem: **a–f** illustrate a comparison of function values versus iterations among PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O for $\lambda = 1,\ 10^{-1},\ 10^{-2},\ 10^{-3}, 10^{-4},\ 10^{-5}$, respectively; **g–l** display a comparison of function values versus iterations among NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O for $\lambda = 1,\ 10^{-1},\ 10^{-2},\ 10^{-3}, 10^{-4},\ 10^{-5}$, respectively

Let us consider a linear inverse problem of the form (71) that we solve it with minimization problems (19) and (20). We set $n = 4096$ and $m = 1024$. The problem is generated by the same procedure given in GPSR (Figueiredo et al. 2007) package available at

```
http://www.lx.it.pt/mtf/GPSR/
```

which is

```
n_spikes = floor(0.01*n);p = zeros(n,1); q = randperm(n);
p(q(1:n_spikes)) = sign(randn(n_spikes,1));B = randn(m,n);
B = orth(B')';bf = B*p; b = bf + sigma*randn(m,1);
```

with $\lambda = \lambda_1 = \lambda_2 = \frac{1}{2}\max(|A^T b|)$. We conclude this section by solving this sparse recovery problem with OSGA-O, OSGA, and the other methods described in the previous section. We show the results in Fig. 5. In this implementation, we apply OSGA-O to the problem, stop it after 10 iterations and save the best function value attained (f_s), and run the others until either the same function value is achieved or the number of iterations reaches 5000. From Fig. 5, it is clear that that OSGA-O attains

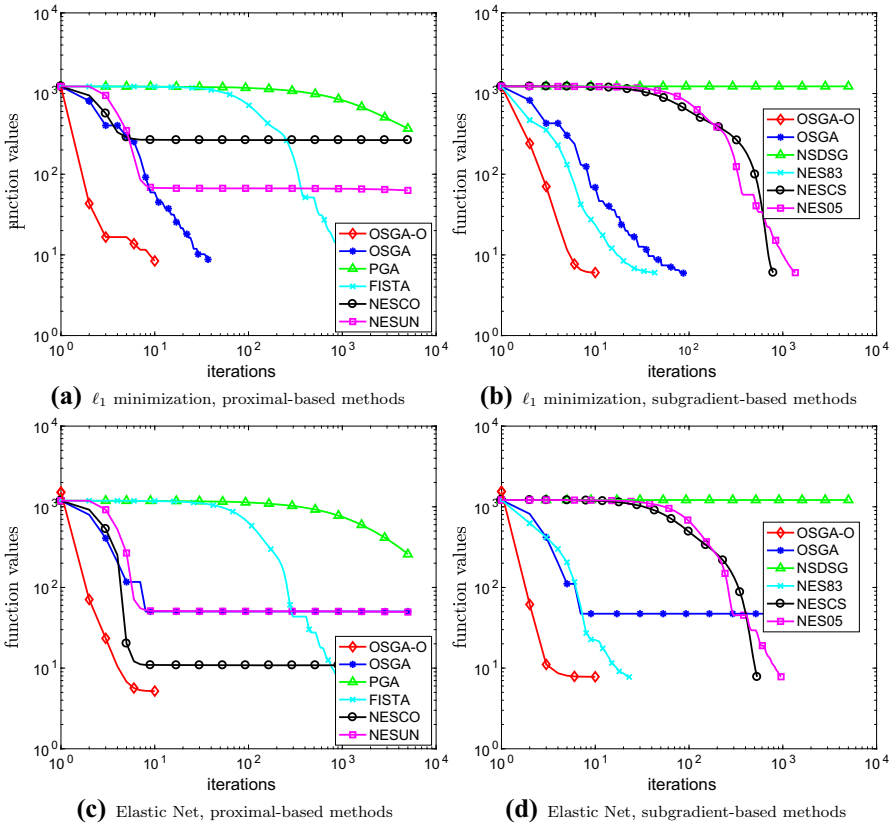


Fig. 5 Function values versus iterations for the ℓ_1 minimization and elastic net problems: **a**, **b** display the results for PGA, FISTA, NESCO, NESUN, OSGA, and OSGA-O for the ℓ_1 minimization; **c**, **d** illustrate the results for NSDSG, NES83, NESCS, NES05, OSGA, and OSGA-O for the elastic net problem

the best performance compared with the others for both ℓ_1 minimization and elastic net problems.

6 Conclusions

This paper discusses the solution of structured nonsmooth convex optimization problems with the complexity $\mathcal{O}(\varepsilon^{-1/2})$, which is optimal for smooth problems with Lipschitz continuous gradients. First, if the nonsmoothness of the problem is manifested in a structured way, the problem is reformulated so that the objective is smooth with Lipschitz continuous gradients in the price of adding a functional constraint to the feasible domain. Then, a new setup of the optimal subgradient algorithm (OSGA-O) is developed to solve the reformulated problem with the complexity $\mathcal{O}(\varepsilon^{-1/2})$.

Next, it is proved that the OSGA-O auxiliary problem is equivalent to a proximal-like problem, which is well-studied due to its appearance in Nesterov-type optimal methods for composite minimization. For several problems appearing in applications, either an explicit formula or a simple iterative scheme for solving the corresponding proximal-like problems is provided.

Finally, some numerical results with random data and a sparse recovery problem are given indicating a good behavior of OSGA-O compared to some state-of-the-art first-order methods, which confirm the theoretical foundations.

Acknowledgements Open access funding provided by University of Vienna. Thanks to Stephen M. Robinson and Defeng Sun for their comments about solving nonsmooth equations. We are very thankful to anonymous referees for a careful reading and many useful suggestions, which improved the paper.

References

Ahookhosh M (2015) High-dimensional nonsmooth convex optimization via optimal subgradient methods, PhD Thesis, University of Vienna

Ahookhosh M (2016) Optimal subgradient algorithms with application to large-scale linear inverse problems (Submitted). http://arxiv.org/abs/1402.7291

Ahookhosh M, Amini K, Kimiaei M (2015) A globally convergent trust-region method for large-scale symmetric nonlinear systems. Numer Funct Anal Optim 36:830–855

Ahookhosh M, Neumaier A (2013) High-dimensional convex optimization via optimal affine subgradient algorithms. In: ROKS workshop, pp 83–84

Ahookhosh M, Neumaier A (2016) An optimal subgradient algorithm with subspace search for costly convex optimization problems (submitted). http://www.optimization-online.org/DB_FILE/2015/04/4852.pdf

Ahookhosh M, Neumaier A (2017) An optimal subgradient algorithms for large-scale bound-constrained convex optimization. Math Methods Oper Res. doi:10.1007/s00186-017-0585-1

Ahookhosh M, Neumaier A (2017) Optimal subgradient algorithms for large-scale convex optimization in simple domains. Numer Algorithm. doi:10.1007/s11075-017-0297-x

Auslender A, Teboulle M (2006) Interior gradient and proximal methods for convex and conic optimization. SIAM J Optim 16:697–725

Beck A, Teboulle M (2003) Mirror descent and nonlinear projected subgradient methods for convex optimization. Oper Res Lett 31(3):167–175
Beck A, Teboulle M (2012) Smoothing and first order methods: a unified framework. SIAM J Optim 22:557–580
Beck A, Ben-Tal A, Guttmann-Beck N, Tetruashvili L (2010) The CoMirror algorithm for solving nonsmooth constrained convex problems. Oper Res Lett 38(6):493–498
Boţ RI, Hendrich C (2013) A double smoothing technique for solving unconstrained nondifferentiable convex optimization problems. Comput Optim Appl 54(2):239–262
Boţ RI, Hendrich C (2015) On the acceleration of the double smoothing technique for unconstrained convex optimization problems. Optimization 64(2):265–288
Boyd S, Xiao L, Mutapcic A (2003) Subgradient methods, Notes for EE392o, Stanford University. http://www.stanford.edu/class/ee392o/subgrad_method.pdf
Candés E (2006) Compressive sampling. In: Proceedings of International Congress of Mathematics, vol 3, Madrid, Spain, pp 1433–1452
Chen Y, Lan G, Ouyang Y (2014) Optimal primal-dual methods for a class of saddle point problems. SIAM J Optim 24(4):1779–1814
Chen Y, Lan G, Ouyang Y (2017) Accelerated scheme for a class of variational inequalities. doi:10.1007/s10107-017-1161-4
Chen Y, Lan G, Ouyang Y (2015) An accelerated linearized alternating direction method of multipliers. SIAM J Imaging Sci 8(1):644–681
Chen Y, Lan G, Ouyang Y, Zhang W (2014) Fast bundle-level type methods for unconstrained and ball-constrained convex optimization. http://arxiv.org/pdf/1412.2128v1.pdf
Combettes P, Pesquet JC (2011) Proximal splitting methods in signal processing. In: Bauschke H, Burachik R, Combettes P, Elser V, Luke D, Wolkowicz H (eds) Fixed-point algorithms for inverse problems in science and engineering. Springer, New York, pp 185–212
Daubechies I, DeVore R, Fornasier M, Güntürk CS (2010) Iteratively reweighted least squares minimization for sparse recovery. Commun Pure Appl Math 63(1):1–38
Devolder O, Glineur F, Nesterov Y (2013) First-order methods of smooth convex optimization with inexact oracle. Math Program 146:37–75
Devolder O, Glineur F, Nesterov Y (2012) Double smoothing technique for large-scale linearly constrained convex optimization. SIAM J Optim 22(2):702–727
Dolan ED, Moré JJ (2002) Benchmarking optimization software with performance profiles. Math Program 91(2):201–213
Donoho DL (2006) Compressed sensing. IEEE Tran Inf Theory 52(4):1289–1306
Esser E, Lou Y, Xin J (2013) A method for finding structured sparse solutions to nonnegative least squares problems with applications. SIAM J Imaging Sci 6(4):2010–2046
Figueiredo MAT, Nowak RD, Wright SJ (2007) Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. IEEE J Sel Top Signal Process 1(4):586–597
Hansen N, Auger A, Ros R, Finck S, Posik P (2010) Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009. In: Proc. Workshop GECCO, pp 1689–1696
Gonzaga CC, Karas EW (2013) Fine tuning Nesterov's steepest descent algorithm for differentiable convex programming. Math Program 138:141–166
Gonzaga CC, Karas EW, Rossetto DR (2013) An optimal algorithm for constrained differentiable convex optimization. SIAM J Optim 23(4):1939–1955
Juditsky A, Nesterov Y (2014) Deterministic and stochastic primal-dual subgradient algorithms for uniformly convex minimization. Stoch Syst 4(1):44–80
Kaufman L, Neumaier A (1996) PET regularization by envelope guided conjugate gradients. IEEE Trans Med Imaging 15:385–389
Kaufman L, Neumaier A (1997) Regularization of ill-posed problems by envelope guided conjugate gradients. J Comput Graph Stat 6(4):451–463
Lagarias JC, Reeds JA, Wright MH, Wright PE (1998) Convergence properties of the Nelder–Mead simplex method in low dimensions. SIAM J Optim 9:112–147
Lan G (2015) Bundle-level type methods uniformly optimal for smooth and non-smooth convex optimization. Math Program 149(1):1–45
Lan G, Lu Z, Monteiro RDC (2011) Primal-dual first-order methods with $O(1/\varepsilon)$ iteration-complexity for cone programming. Math Program 126:1–29

Li DH, Yamashita N, Fukushima M (2001) Nonsmooth equation based bfgs method for solving KKT systems in mathematical programming. J Optim Theory Appl 109(1):123–167
Liu H, Zhang J, Jiang X, Liu J (2010) The group Dantzig selector. J Mach Learn Res Proc Track 9:461–468
Nemirovsky AS, Yudin DB (1983) Problem complexity and method efficiency in optimization. Wiley, New York
Nesterov Y (2004) Introductory lectures on convex optimization: a basic course. Kluwer, Dordrecht
Nesterov Y (1983) A method of solving a convex programming problem with convergence rate $O(1/k^2)$, Doklady AN SSSR (In Russian), 269:543–547. English translation: Soviet Math. Dokl., 27: 372–376 (1983)
Nesterov Y (2005) Smooth minimization of non-smooth functions. Math Prog 103:127–152
Nesterov Y (2005) Excessive gap technique in nonsmooth convex minimization. SIAM J Optim 16:235–249
Nesterov Y (2011) Barrier subgradient method. Math Program 127:31–56
Nesterov Y (2006) Primal-dual subgradient methods for convex problems. Math Program 120:221–259
Nesterov Y (2013) Gradient methods for minimizing composite objective function. Math Program 140:125–161
Nesterov Y (2015) Universal gradient methods for convex optimization problems. Math Program 152(1):381–404
Neumaier A (2016) OSGA: a fast subgradient algorithm with optimal complexity. Math Program 158(1):1–21
Neumaier A (2001) Introduction to numerical analysis. Cambridge University Press, Cambridge
Pang JS, Qi L (1993) Nonsmooth equations: motivation and algorithms. SIAM J Optim 3:443–465
Parikh N, Boyd S (2013) Proximal algorithms. Found Trends Optim 1(3):123–231
Polyak B (1987) Introduction to optimization. Optimization Software Inc., Publications Division, NewYork
Potra FA, Qi L, Sun D (1998) Secant methods for semismooth equations. Numerische Mathematik 80(2):305–324
Qi L (1995) Trust region algorithms for solving nonsmooth equations. SIAM J Optim 5:219–230
Qi L, Sun D (1999) A survey of some nonsmooth equations and smoothing Newton methods. Prog Optim 30:121–146
Rauhut H, Ward R (2016) Interpolation via weighted l_1-minimization. Appl Comput Harmon Anal 40(2):321–351
Shor NZ (1985) Minimization methods for non-differentiable functions. Springer, Berlin (Springer series in computational mathematics)
Sun D, Han J (1997) Newton and quasi-Newton methods for a class of nonsmooth equations and related problems. SIAM J Optim 7(2):463–480
Tseng P (2008) On accelerated proximal gradient methods for convex-concave optimization, Technical report, Mathematics Department, University of Washington. http://pages.cs.wisc.edu/~brecht/cs726docs/Tseng.APG.pdf