FULL LENGTH PAPER

Series A

The multiproximal linearization method for convex composite problems

Jérôme Bolte[1] · Zheng Chen[3] · Edouard Pauwels[2]

Abstract

Composite minimization involves a collection of smooth functions which are aggregated in a nonsmooth manner. In the convex setting, we design an algorithm by linearizing each smooth component in accordance with its main curvature. The resulting method, called the Multiprox method, consists in solving successively simple problems (e.g., constrained quadratic problems) which can also feature some proximal operators. To study the complexity and the convergence of this method, we are led to prove a new type of qualification condition and to understand the impact of multipliers on the complexity bounds. We obtain explicit complexity results of the form $O(\frac{1}{k})$ involving new types of constant terms. A distinctive feature of our approach is to be able to cope with oracles involving moving constraints. Our method is flexible enough to include the moving balls method, the proximal Gauss–Newton's method, or the forward–backward splitting, for which we recover known complexity results or establish new ones. We show through several numerical experiments how the use of multiple proximal terms can be decisive for problems with complex geometries.

Keyword Composite optimization · Convex optimization · Complexity · First order methods · Proximal Gauss–Newton's method · Prox-linear method

✉ Zheng Chen
  z_chen@zju.edu.cn

  Jérôme Bolte
  jbolte@ut-capitole.fr

  Edouard Pauwels
  edouard.pauwels@irit.fr

[1] Toulouse School of Economics, Université Toulouse Capitole,
   Manufacture des Tabacs, 31015 Toulouse, France

[2] IRIT, Université Paul Sabatier, 118 route de Narbonne, 31062 Toulouse, France

[3] School of Aeronautics and Astronautics, Zhejiang University, No. 38, Zheda Road, Xihu District,
   310027 Hangzhou, Zhejiang, China

Mathematics Subject Classification 90C25 · 90C30 · 90C47

1 Introduction

Proximal methods are at the heart of optimization. The idea has its roots within the infimal convolution of Moreau [29] with early algorithmic applications to variational inequalities [28], constrained minimization [40], and mechanics [30]. The principle is elementary but far reaching: it simply consists in generating algorithms by considering successive strongly convex approximations of a given objective function. Many methods can be seen through these lenses, like for instance, the classical gradient method, the gradient projection method, or mirror descent methods [1,22,34,42,43]. At this day, the most famous example is probably the forward–backward splitting algorithm [11,26,37,48] and its accelerated variant FISTA [5]. Many generalizations in many settings followed, see for instance [4,12–14,17,21,36,44,45,49].

In order to deal with problems with more complex structure, we are led to consider models of the form

$$(\mathcal{P}): \quad \min \left\{ g\left(F(x)\right) : x \in \mathbb{R}^n \right\},$$

where $F = (f_1, \ldots, f_m)$ is a collection of convex differentiable functions with L_i Lipschitz continuous gradient and $g : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a proper convex lower semicontinuous function.

The function g is allowed to take infinite values offering a great flexibility in the modeling of constraints while it is often assumed to be finite in the literature. When g is restricted to be Lipschitz, a natural proximal approach to this problem consists in linearizing the smooth part, leaving the nonsmooth term unchanged and adding an adequate quadratic form. Given x in $\mathbb{R}^n$, one obtains the proximal Gauss Newton method or the prox-linear method

$$\text{(PGNM)} \quad x^+ = \operatorname*{argmin}_{y \in \mathbb{R}^n} \left(g\left(F(x) + \nabla F(x)(y - x)\right) + \frac{\lambda}{2} \|y - x\|^2 \right),$$

where

$$\lambda \geq L_g \max_{1, \ldots, m} L_i \tag{1}$$

with L_g being the Lipschitz constant of g. The method[1] progressively emerged from quadratic programming, see [39] and references therein, but also from ideas à la Gauss–Newton [7,8,18]. It was eventually formulated under a proximal form in [23]. It allows to deal with general nonlinear programming problems and unifies within a simple framework many different classes of methods met in practice [6,9,15,23–25,38]. It is one of the rare primal methods for composite problems without linesearch[2] (see

---

[1] Also known as the proximal Gauss–Newton's method

[2] Indeed, PGNM is somehow a "constant step size" method

also [2]), and as such assessing its complexity is a natural question. Even though the complexity analysis of convex first order methods has now become classical (see e.g., [33]), considerable difficulties remain for composite problems with such generality. One of the reasons is that constraints, embodied in g, generate multipliers whose role is not yet understood. To our knowledge, there are very few works in this line. In [15] the authors study this method under error bounds conditions and establish linear convergence results, see also [33, Section 2.3] for related results. In a recent article [16], the authors propose an acceleration of the same method and they obtain faster convergence guaranties under mild assumptions. The global complexity with general assumptions on a convex g seems to be an open question.

We work here along a different line and we propose a new flexible method with *inner quadratic/proximal approximations*. Given a starting point x, we consider

$$\text{(Multiprox)} \quad x^+ = \operatorname*{argmin}_{y \in \mathbb{R}^n} g \left(F(x) + \nabla F(x)(y - x) + \|y - x\|^2 \begin{pmatrix} \frac{\lambda_1}{2} \\ \vdots \\ \frac{\lambda_m}{2} \end{pmatrix} \right),$$

or more explicitly

$$x^+ = \operatorname*{argmin}_{y \in \mathbb{R}^n} g \begin{pmatrix} f_1(x) + \nabla f_1(x)^T (y - x) + \lambda_1 \frac{\|y - x\|^2}{2} \\ \vdots \\ f_m(x) + \nabla f_m(x)^T (y - x) + \lambda_m \frac{\|y - x\|^2}{2} \end{pmatrix},$$

where

$$\lambda_i \geq L_i \text{ for } i = 1, \ldots, m, \tag{2}$$

and g is allowed to be extended valued. In order to preserve the convexity properties of the local model, the function g is assumed to be componentwise nondecreasing. In spite of the monotonicity restriction on g, this model is quite versatile and includes as special cases general inequality constrained convex programs, important min–max problems or additive composite models. Observe that the local approximation used in Multiprox is sharper than the one in the proximal Gauss–Newton's method since it relies on the vector $(L_1, \ldots, L_m)$ rather than on the mere constant $\max\{L_i : i = 1, \ldots, m\}$. Due to the presence of multiple proximal/gradient terms we called our method Multiprox. The key idea behind Multiprox, already present in [2], is to design local approximations through upper quadratic models specifically taylored for each of the components. This makes the method well adapted to the geometry of the original problem and allows in general to take much larger and clever steps as illustrated in numerical experiments in the last section.

Studying this method presents several serious difficulties. First g may not have full domain (contrary to what is assumed in [8,15,16,25]), which reflects the fact that subproblems may feature "moving constraint sets". Even though moving constraints are very common in sequential convex programming, we did not find any genuine

results on complexity in the literature. Secondly the nature of our algorithm rises new issues concerning qualification conditions (and subsequently on the role of Lagrange multipliers in the complexity analysis). The qualification condition we consider is surprisingly simple to state, yet non trivial to study:

$$F^{-1}(\text{int dom } g) \neq \emptyset.$$

This Slater's like condition is specific to situations when g is monotone and was already used in [20, Theorem 2] to provide formulas for computing the Legendre transform of composite functions. Under this condition, we establish a complexity result of the form $O(\frac{1}{k})$ whose constant term depends on the geometry of $(\mathcal{P})$ through a quantity combining various curvature constants of the components of F with the multipliers attached to the subproblems. When g is finite and Lipschitz continuous, the complexity boils down to

$$\frac{L_g \sqrt{\sum_{i=1}^m L_i^2}}{2k} \|x_0 - x_*\|^2$$

where L_g is the Lipschitz constant of g. The exact same analysis leads to improved bounds if the outer function g has a favorable structure, such as the coordinatewise maximum, in which case, the numerator reduces to $\max_{i=1,\ldots,m}\{L_i\}$. To our knowledge, these results were missing from the literature.

We study further the boundedness properties of the sequences generated by Multiprox. We derive in particular quantitative bounds on the multipliers for hard constrained[3] problems. This allows us in turn to derive complexity estimates for sequential convex methods in convex nonlinear programming such as the moving balls method [2] and its nonsmooth objective variant [46]. To put this into perspective, the only feasible methods for nonlinear programming which come with such explicit estimates are, to the best of our knowledge, interior point methods (see [32,50] and references therein).

We also analyze into depth the important cases when Lipschitz constants are not known[4] or when they only exist locally (e.g., the C^2 case). In this setting, the "step sizes" (the various λ_i) cannot be tuned a priori and thus complexity results are much more difficult to establish due to the use of linesearch routines. Yet we obtain some useful rates and we are able to establish convergence of the sequence. Once more we insist on the fact that convergence in this setting is not an easy matter and very few results are known [2,3,9,10,47].

Finally, we illustrate the efficiency of Multiprox on synthetic data. We consider a composite function consisting of the maximum of convex quadratic functions with different smoothness moduli. We compare our method with the proximal Gauss–Newton algorithm and its accelerated variant described in [16]. These experiments illustrate that, although the complexity estimates of the Multiprox algorithm are not

---

[3] Here, hard constraints means that only feasible point can be considered, contrasting with infeasible methods (e.g., [3,10]).

[4] We refer to constants relative to the gradients.

better than existing estimates for the concurrent methods, its adaptivity to different smoothness moduli gives it a crucial advantage in practice.

Outline In Sect. 2 we describe the composite optimization problem and study qualification conditions. Section 3 provides first general complexity and convergence results and presents consequences for specific models. Section 4 on linesearch describes cases when Lipschitz constants are unknown or merely locally bounded. In the Sect. 5 we provide numerical experiments illustrating the efficiency of our method.

Notations $\mathbb{R}^n$ is the n-dimensional Euclidean space equipped with the Euclidean norm $\|\cdot\|$. For $x \in \mathbb{R}^n$ and $r \geq 0$, $B(x, r)$ denotes the closed Euclidean ball of radius r centered at x. By $\mathbb{R}^n_+$, we denote the n-dimensional nonnegative orthant (n-dimensional vectors with nonnegative entries). The notations of $<, \leq, >$, and $\geq$ between vectors indicate that the corresponding inequalities are met coordinatewise.

Our notations for convex analysis are taken from [41]. We recall the most important ones. Given a convex extended-valued convex function $h : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$, we set

$$\operatorname{dom} h := \{z \in \mathbb{R}^m : h(z) < +\infty\}.$$

The subdifferential of h at any $\bar{z} \in \operatorname{dom} h$ is defined as usual by

$$\partial h(\bar{z}) := \{\lambda \in \mathbb{R}^m : (z - \bar{z})^T \lambda + h(\bar{z}) \leq h(z), \forall z \in \mathbb{R}^m\},$$

and is the empty set if $z \notin \operatorname{dom} h$.

For any convex subset $D \subset \mathbb{R}^m$, we denote by i_D its indicator function—recall that $i_D(x) = 0$ if x is in D, $i_D(x) = +\infty$ otherwise.

2 Minimization problem and algorithm

2.1 Composite model and assumptions

We consider a composite minimization problem of the type:

$$(\mathcal{P}) : \quad \min \left\{ g(F(x)) : x \in \mathbb{R}^n \right\},$$

where $g : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ and $F : \mathbb{R}^n \to \mathbb{R}^m$. We set $F = (f_1, \dots, f_m)$ and we make the standing assumptions:

Assumption 1 (a) Each f_i is continuously differentiable, convex, with L_i Lipschitz continuous gradient ($L_i \geq 0$).
(b) The function $g : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is convex, proper, lower semicontinuous and L_g Lipschitz continuous on its domain. That is

$$|g(x) - g(y)| \leq L_g \|x - y\| \text{ for all } x, y \text{ in } \operatorname{dom} g.$$

For each $i = 1, \ldots, m$ with $L_i > 0$, then g is nondecreasing in its i-th argument.[5] In other words g is nondecreasing in its i-th argument whenever f_i is not affine.

Remark 1 (a) Note that the monotonicity restriction on g implies some restrictions. For example, ignoring the affine components of F, for any $z \in \mathrm{dom}\, g$, we also have $z - \mathbb{R}_+^m \subset \mathrm{dom}\, g$, so that $\mathrm{dom}\, g$ is not compact. Prominent examples for g includes the max function, the indicator of $\mathbb{R}_-^m$ (which allows to handle non-linearity of the type $f_i \leq 0$), support functions of a subset of positive numbers. Depending on the structure of F other examples are possible (see further sections).
(b) Contrary to $L_1, \ldots, L_m$, the value of L_g is never required to design/run the algorithm (see in particular Theorems 1 and 3).

Assumption 2 The function $g \circ F$ is proper and has a minimizer.

Observe that the monotonicity of g and the convexity of F in Assumption 1 ensure that the problem $(\mathcal{P})$ is a convex optimization problem, in other words:

$$g \circ F \text{ is convex.} \tag{3}$$

2.2 The multiproximal linearization algorithm

Let us introduce the last fundamental ingredient necessary to the description of our method:

$$\boldsymbol{L} := (L_1, \ldots, L_m)^T \in \mathbb{R}_+^m.$$

Observe that the monotonicity of g in Assumption 1 implies that for any $z \in \mathrm{dom}\, g$, one has

$$\boldsymbol{L}^T \lambda \geq 0, \text{ for any } \lambda \in \partial g(z). \tag{4}$$

The central idea is to use quadratic upper approximations componentwise on the smooth term F. We thus introduce the following mapping

$$H(x, y) := F(x) + \nabla F(x)(y - x) + \frac{\boldsymbol{L}}{2} \|y - x\|^2, \quad (x, y) \in \mathbb{R}^n \times \mathbb{R}^n,$$

where ∇F denotes the Jacobian matrix of F. This leads to the following family of subproblems

$$(\mathcal{P}_x): \quad \min\{ g(H(x, y)) : y \in \mathbb{R}^n \}.$$

where x ranges in $F^{-1}(\mathrm{dom}\, g)$. As shall be discussed in further sections this problem is well-posed for broad classes of examples. We make the following additional standing assumption:

---

[5] For any such i and any m real numbers $z_1, \ldots z_m$, the function $z \mapsto g(z_1, \ldots, z_{i-1}, z, z_{i+1}, \ldots, z_m)$ is nondecreasing. In particular, its domain is either the whole of $\mathbb{R}$ or a closed half line $(-\infty, a]$ for some $a \in \mathbb{R}$, or empty.

Assumption 3 For any x in $F^{-1}(\operatorname{dom} g)$, the function $g \circ H(x, \cdot)$ has a minimizer.

Elementary but important properties of problem $(\mathcal{P}_x)$ are given in the following lemma.

Lemma 1 *For any $x \in F^{-1}(\operatorname{dom} g)$, the following statements hold:*

(1) $\operatorname{dom} g(H(x, \cdot)) \subset \operatorname{dom} g \circ F$ *and*

$$g(F(y)) \le g(H(x, y)), \quad \forall y \in \operatorname{dom} g(H(x, \cdot)).$$

(2) $g(F(x)) = g(H(x, x))$.
(3) $g \circ H(x, \cdot)$ *is proper and convex.*

Proof According to Assumption 1 and the descent Lemma, [33, Lemma 1.2.3], for every i in $\{1, \ldots, m\}$ one has

$$f_i(y) \begin{cases} \le f_i(x) + \nabla f_i(x)^T (y - x) + \frac{L_i}{2} \|y - x\|^2, \ L_i > 0, \\ = f_i(x) + \nabla f_i(x)^T (y - x) + \frac{L_i}{2} \|y - x\|^2, \ L_i = 0, \end{cases} \quad \forall (x, y) \in \mathbb{R}^n \times \mathbb{R}^n.$$

By the monotonicity properties of g we obtain that $g(F(y)) \le g(H(x, y))$ for all $y \in \mathbb{R}^n$ and (1) is proved. Items (2) and (3) follow from simple verifications. $\square$

Multiproximal method (Multiprox)

Choose $x_0 \in F^{-1}(\operatorname{dom} g)$ and iterate for $k \in \mathbb{N}$:

$$x_{k+1} \in \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \ g\left(F(x_k) + \nabla F(x_k)(y - x_k) + \frac{\boldsymbol{L}}{2} \|y - x_k\|^2 \right) \tag{5}$$

with the choice $x_{k+1} = x_k$ whenever x_k is a minimizer of $g(F(x_k, \cdot))$.

If we set

$$p(x) := \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \ g(H(x, y))$$

for any x in $F^{-1}(\operatorname{dom} g)$ the algorithm simply reads as $x_{k+1} \in p(x_k)$ with $x_{k+1} = x_k$ whenever $x_k \in p(x_k)$.

Remark 2 (a) Item (1) of Lemma 1, along with Assumption 3, implies that the algorithm is well defined. Observe that Lemma 1 actually shows that our algorithm is based on the classical idea of minimizing successively majorant functions coinciding at order 1 with the original function.
(b) As already mentioned, the algorithm does not require the knowledge of the Lipschitz constant of g on its domain.

2.3 Examples and implementation issues

We give here two important examples for which the subproblems are simple quadratic problems.

2.3.1 Convex nonlinear programming

Consider the classical convex nonlinear programming problem

$$\min \left\{ f(x) : f_i(x) \leq 0, \ i = 1, \ldots, m, \ x \in \mathbb{R}^n \right\}, \tag{6}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a convex function with L_f Lipschitz continuous gradient and each f_i is defined as in Assumption 1. Using the reformulation: $\min\{f(x) + i_{\mathbb{R}_-^m}(f_1(x), \ldots, f_m(x)) : x \in \mathbb{R}^n\}$ and setting $g(s, y_1, \ldots, y_m) = s + i_{\mathbb{R}_-}(y_1, \ldots, y_m)$ and $F(x) = (f(x), f_1(x), \ldots, f_m(x))$, the problem[6] (6) can be seen as an instance of $(\mathcal{P})$. Multiprox writes

$$x_{k+1} \in \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \quad f(x_k) + \nabla f(x_k)^T(y - x_k) + \frac{L_f}{2}\|y - x_k\|^2$$

$$\text{s.t.} \quad f_i(x_k) + \nabla f_i(x_k)^T(y - x_k) + \frac{L_i}{2}\|y - x_k\|^2 \leq 0, \ i = 1, \ldots, m, \tag{7}$$

which is a generalization of the moving balls method [2,46] in the sense that our algorithm offers the additional flexibility that affine constraints can be left unchanged in the subproblem (by setting the corresponding L_i to 0). Assume for simplicity that $L_f > 0$.

Computing x_{k+1} leads to solve very specific quadratic problems. Indeed, if q is a quadratic form appearing within the above subproblem, its Hessian is given by $\nabla^2 q = cI_n$ (with $c > 0$) or $\nabla^2 q = 0_n$, where I_n (resp. 0_n) denotes the identity (resp. null) matrix in $\mathbb{R}^{n \times n}$. Computing x_{k+1} amounts to computing the Euclidean projection of a point to an intersection of Euclidean balls/hyperplanes. Both types of sets have extremely simple projection operators and one can thus apply Dykstra's projection algorithm (see e.g., [4]) or a fast quadratic solver (see e.g., [35]). Let us also mention that this type of problems can be treated very efficiently by specific methods based on activity detection described in [2,46].

2.3.2 Min–max problems

We consider the problem

$$\min \left\{ \max_{i=1,\ldots,m} f_i(x) : x \in \mathbb{R}^n \right\}.$$

---

[6] There is a slight shift in the indices of F

This type of problems is very classical in optimization but also in game theory (see e.g., [33]). Observing that $g = \max_{1,...,m} f_i$ satisfies our assumptions, we see that the problem is already under the form $(\mathcal{P})$. The substeps assume thus the form

$$x_{k+1} \in \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \left(\max_{i=1,...,m} \left(f_i(x_k) + \nabla f_i(x_k)^T (y - x_k) + \frac{L_i}{2} \|y - x_k\|^2 \right) \right).$$

As previously explained, this subproblem can be rewritten as a simple quadratic problem and it can thus be solved through the same means. In the last section, we illustrate the numerical efficiency of Multiprox on this type of problems.

Remark 3 Other cases can be treated by Multiprox. Consider for example, the following problem

$$\min_{x \in \mathbb{R}^n} \quad \max\{f_1(x), f_2(x)\} + \|x\|_1,$$
$$\text{s.t.} \quad \max\{f_3(x), f_4(x)\} + \|x\|_\infty \leq 0.$$

where f_i are smooth convex functions for $i = 1, \ldots, 4$. Then, for any $x \in \mathbb{R}^n$, the solution of $(\mathcal{P}_x)$ can be computed as follows:

$$\min_{y \in \mathbb{R}^n, s \in \mathbb{R}, t \in \mathbb{R}^n, u \in \mathbb{R}, v \in \mathbb{R}} \quad s + \sum_{i=1}^n t_i$$

$$\text{s.t.} \quad f_1(x) + \nabla f_1(x)^T (y - x) + \frac{L_1}{2} \|y - x\|^2 \leq s$$
$$f_2(x) + \nabla f_2(x)^T (y - x) + \frac{L_2}{2} \|y - x\|^2 \leq s$$
$$x_i \leq t_i, \; i = 1, \ldots, n$$
$$-x_i \leq t_i, \; i = 1, \ldots, n$$
$$t_i \leq v, \; i = 1, \ldots, n$$
$$f_3(x) + \nabla f_3(x)^T (y - x) + \frac{L_3}{2} \|y - x\|^2 \leq u$$
$$f_4(x) + \nabla f_4(x)^T (y - x) + \frac{L_4}{2} \|y - x\|^2 \leq u$$
$$u + v \leq 0$$

which is a quadratically constrained linear program.

2.4 Qualification, optimality conditions and a condition number

The first issue met in the study of Multiprox is the one of qualification conditions both for $(\mathcal{P})$ and $(\mathcal{P}_x)$. Classical qualification conditions take the form

$$N_{\text{dom } g}(F(x)) \cap \ker(\nabla F(x)^T) = \{0\}, \quad \forall x \in F^{-1}(\text{dom } g), \tag{8}$$

where $N_{\mathrm{dom}\,g}(F(x))$ denotes the normal cone to dom g (see e.g., [41, Example 10.8]). In this section, we first describe a different qualification condition which takes advantage of the specific monotonicity properties of g as described in Assumption 1. This condition was already used in [20] to provide a formula for the Legendre conjugate of the composite function $g(F(\cdot))$. In this setting, we show that this condition allows to use the chain rule and provides optimality conditions which will be crucial to study Multiprox algorithm. We emphasize that this qualification condition is much more practical than conditions of the form (8). Besides it is also naturally amenable to quantitative estimation which appears to be fundamental for the computation of complexity estimates.

Qualification condition and chain rule The following Slater's like qualification condition is specific to the "monotone composite model" we consider here (see [20, Theorem 2] and references therein for pioneering works on the subject).

We make the following standing assumption:

Assumption 4 *(Qualification)* There exists $\bar{x}$ in $F^{-1}(\mathrm{int}\,\mathrm{dom}\,g)$.

To our knowledge, the following result is new. It illustrates the main interest of Assumption 4:

Proposition 1 (Chain rule) *For all x in $F^{-1}(\mathrm{dom}\,g)$:*

$$\partial\,(g \circ F)\,(x) = \nabla F(x)^T \partial g(F(x)).$$

The proof of Proposition 1 is postponed to "Appendix A". Another interesting and useful consequence of Assumption 4 is that it automatically ensures a similar qualification condition for all subproblems $(\mathcal{P}_x)$.

Proposition 2 (Qualification for subproblems) *For all x in $F^{-1}(\mathrm{dom}\,g)$ there exists $w(x) \in \mathbb{R}^n$ such that*

$$H(x, w(x)) \in \mathrm{int}\,\mathrm{dom}\,g.$$

Proof An explicit construction of $w(x)$ is provided in Lemma 6 ("Appendix B"). □

These results provide necessary and sufficient optimality conditions for $(\mathcal{P})$ and for $(\mathcal{P}_x)$.

Corollary 1 (Fermat's rule) *A point $x_* \in F^{-1}(\mathrm{dom}\,g)$ is a minimizer of $g \circ F$ if and only if*

$$\exists \lambda_* \in \partial g(F(x_*)) \text{ s.t. } \nabla F(x_*)^T \lambda_* = 0.$$

For any x in $F^{-1}(\mathrm{dom}\,g)$, and for any y in $\mathbb{R}^n$, we have $y \in p(x)$ if and only if

$$\exists v \in \partial g(H(x, y)) \text{ s.t. } \nabla_y H(x, y)^T v = 0.$$

The first part of the corollary is of course an immediate consequence of the chain rule in Proposition 1. The second part holds true for the same reasons, replacing Assumption 4 by Proposition 2.

Lagrange multipliers and a condition number

Definition 1 *(Lagrange multipliers for* $(\mathcal{P}_x)$*)* For any $x \in F^{-1}(\text{dom } g)$ and any $y \in p(x)$, we set

$$\mathcal{V}(x, y) := \{v \in \partial g(H(x, y)) : \nabla_y H(x, y)^T v = 0\}.$$

The following quantity, which can be seen as a kind of condition number captures the boundedness properties of the multipliers for the subproblems. It will play a crucial role in our complexity studies.

Lemma 2 (A condition number) *Given any nonempty compact set $K \subseteq F^{-1}(\text{dom } g)$ and any $\gamma \geq \min g \circ F$, the following quantity is finite*

$$\mathscr{C}_\gamma(K) := \sup \left\{ \inf\{\boldsymbol{L}^T v : v \in \mathcal{V}(x, y)\} : x \in K, \ g(F(x)) \leq \gamma, \ y \in p(x) \right\}.$$

Proof We shall see that Lemma 4 provides an explicit bound on this condition number and as a consequence $\mathscr{C}_\gamma(K)$ is finite.   □

Remark 4 ($\mathscr{C}_\gamma(K)$ as a condition number) In numerical analysis, the term condition number usually refers to a measure of the magnitude of variation of the output as a function of the variation of the input. For instance, when one studies the usual gradient method for minimizing a convex function $f : \mathbb{R}^n \to \mathbb{R}$ with Lipschitz gradient L_f, one is led to algorithms of the form $x_{k+1} = x_k - \nabla f(x_k)/L_f$ and the complexity takes the form $L_f \|x_0 - x^*\|^2/(2k)$ where x^* is a minimizer of f. The bigger L_f is, the worse the estimate is. It captures in particular the compositional structure of the model by combining the smoothness modulus of F with some regularity for g captured through KKT multipliers.

3 Complexity and convergence

This section is devoted to the exposition of the complexity results obtained for Multiprox. In the first subsection, we describe our main results, an abstract convergence result and we provide explicit complexity estimate. We then describe the consequences for known algorithms.

3.1 Complexity results for Multiprox

General complexity and convergence results We begin by establishing that Multiprox is a descent method:

Lemma 3 (Descent property) *For any $x \in F^{-1}(\text{dom } g)$, if $y \in p(x)$, one has*

$$g(F(y)) - g(F(x)) \leq -\frac{\|y - x\|^2}{2} \boldsymbol{L}^T v,$$

for all v in $\mathcal{V}(x, y)$.

Proof By Definition 1, for every $x \in F^{-1}(\text{dom } g)$ and $y \in p(x)$ we have $\nabla_y H(x, y)^T v = 0$, for any $v \in \mathcal{V}(x, y)$. In other words

$$\nabla F(x)^T v + (y - x)(\boldsymbol{L}^T v) = 0, \ \forall v \in \mathcal{V}(x, y). \tag{9}$$

By convexity of g, one has

$$g(H(x, y)) - g(F(x)) \leq [H(x, y) - F(x)]^T v$$
$$= [\nabla F(x)^T v]^T (y - x) + \frac{\|y - x\|^2}{2} \boldsymbol{L}^T v, \ \forall v \in \mathcal{V}(x, y).$$

Substituting this inequality into Eq. (9) yields

$$g(H(x, y)) - g(F(x)) \leq -\frac{\|y - x\|^2}{2} \boldsymbol{L}^T v, \ \forall v \in \mathcal{V}(x, y).$$

Combining Lemma 1 with this inequality completes the proof.   □

Remark 5 (Multiprox is a descent method) For any sequence $(x_k)_{k \in \mathbb{N}}$ generated by Multiprox, the corresponding sequence of objective values $(g \circ F(x_k))_{k \in \mathbb{N}}$ is nonincreasing.

Let us set

$$S := \text{argmin } g \circ F$$
$$= \{x \in F^{-1}(\text{dom } g) : \exists \lambda \in \partial g(F(x)) \text{ s.t. } \nabla F(x)^T \lambda = 0\}.$$

The following theorem is our first main result under the assumption that the smoothness moduli of the components of F are known and available to the user. The first item is a complexity result while the second one is a convergence result. Discussion regarding the impact of our results on other existing algorithms is held in Sect. 3.2.

Theorem 1 (Complexity and convergence for Multiprox) *Let* $(x_k)_{k \in \mathbb{N}}$ *be a sequence generated by* Multiprox. *Then, the following statements hold:*

(i) *For any* $x_* \in S$ *set* $B_{x_0, x_*} = B(x_*, \|x_0 - x_*\|)$. *Then, for all* $k \geq 1$,

$$g(F(x_k)) - g(F(x_*)) \leq \frac{\mathscr{C}_{g(F(x_0))}\left(B_{x_0, x_*}\right)}{2k} \|x_0 - x_*\|^2; \tag{10}$$

(ii) *The sequence* $(x_k)_{k \in \mathbb{N}}$ *converges to a point in the solution set* S.

Proof (i) Let n be a positive integer. The following elementary observation appears to be very useful: given any function of the form

$$f : \mathbb{R}^n \rightarrow \mathbb{R}, \; x \mapsto a\|x\|^2 + b^T x + c,$$

where $a \in \mathbb{R}_+$, $b \in \mathbb{R}^n$, and $c \in \mathbb{R}$, if there exists $\hat{x}$ in $\mathbb{R}^n$ such that $\nabla f(\hat{x}) = 0$, one has

$$f(x) = f(\hat{x}) + a\|x - \hat{x}\|^2 \; \forall x \in \mathbb{R}^n. \tag{11}$$

By Definition 1, for any integer $k > 0$ and for every $v_k \in \mathcal{V}(x_{k-1}, x_k)$ the gradient of $H(x_{k-1}, \cdot)^T v_k$ at x_k is zero, i.e., $\nabla_y [H(x_{k-1}, x_k)^T v_k] = 0$. Combining the explicit expression of $H(x_{k-1}, \cdot)^T v_k$ with Eq. (11) and considering that $L^T v_k \geq 0$ [see (4)], one has, for any v_k in $\mathcal{V}(x_{k-1}, x_k)$ and any x_* in S,

$$H(x_{k-1}, x_k)^T v_k = F(x_{k-1})^T v_k + [\nabla F(x_{k-1})(x_k - x_{k-1})]^T v_k$$
$$+ \frac{\|x_k - x_{k-1}\|^2}{2} L^T v_k \tag{12}$$

$$= H(x_{k-1}, x_*)^T v_k - \frac{L^T v_k}{2} \|x_k - x_*\|^2. \tag{13}$$

According to the convexity of g, for any v_k in $\mathcal{V}(x_{k-1}, x_k)$ and any x_* in S,

$$g \circ H(x_{k-1}, x_k) - g \circ F(x_*) \leq [H(x_{k-1}, x_k) - F(x_*)]^T v_k. \tag{14}$$

As a consequence, for any v_k in $\mathcal{V}(x_{k-1}, x_k)$ and any x_* in S, one has

$$g \circ F(x_k) - g \circ F(x_*)$$
$$\overset{(a)}{\leq} [H(x_{k-1}, x_k) - F(x_*)]^T v_k$$
$$\overset{(b)}{=} H(x_{k-1}, x_*)^T v_k - \frac{L^T v_k}{2} \|x_k - x_*\|^2 - F(x_*)^T v_k$$
$$\overset{(c)}{=} [F(x_{k-1}) + \nabla F(x_{k-1})(x_* - x_{k-1})]^T v_k + \frac{L^T v_k}{2} (\|x_{k-1} - x_*\|^2 - \|x_k - x_*\|^2)$$
$$- F(x_*)^T v_k$$
$$\overset{(d)}{\leq} F(x_*)^T v_k + \frac{L^T v_k}{2} (\|x_{k-1} - x_*\|^2 - \|x_k - x_*\|^2) - F(x_*)^T v_k$$
$$= \frac{L^T v_k}{2} (\|x_{k-1} - x_*\|^2 - \|x_k - x_*\|^2), \tag{15}$$

where (a) is obtained by combining Lemma 1 with Eq. (14), for (b) we use Eq. (12), for (c) we expand $H(x_{k-1}, x_*)^T v_k$ explicitly, and for (d) we use the property that the i-th coordinate of v_k is nonnegative if $L_i > 0$ (c.f. Assumption 1) and the coordinatewise convexity of F.

Let us consider beforehand the stationary case. If there exists a positive integer k_0 and a subgradient $v \in \mathcal{V}(x_{k_0-1}, x_{k_0})$ such that $L^T v = 0$, one deduces from Eq. (15) that $g(F(x_{k_0})) = \inf\{g \circ F\}$. Recalling that the sequence $\big(g(F(x_k))\big)_{k \in \mathbb{N}}$ is nonincreasing

(cf. Remark 5), one thus has $x_{k_0+j} \in S$ for any $j \in \mathbb{N}$. Using Lemma 1, it follows that for any $j \in \mathbb{N}$, $x_{k_0+j} \in p(x_{k_0+j})$. Hence, for all $j \in \mathbb{N}$, $x_{k_0+j} = x_{k_0} \in S$ and the algorithm actually stops at a global minimizer.

This ensures that if there exists k_0 such that $1 \le k_0 \le k$ and a subgradient $v \in \mathcal{V}(x_{k_0-1}, x_{k_0})$ with $\boldsymbol{L}^T v = 0$, then Eq. (10) holds since in this case, $x_k = x_{k_0} \in S$.

To proceed, we now suppose that $\boldsymbol{L}^T v > 0$ for every $v \in \mathcal{V}(x_{j-1}, x_j)$ and for every j in $\{1, \dots, k\}$. Observe first that by (15) the sequence $\|x_k - x_*\|$ is nonincreasing and, since x_k is a descent sequence for $g \circ F$, it evolves within $B(x_*, \|x_0 - x_*\|)$ and satisfies $g(F(x_k)) \le g(F(x_0))$ for all $k \in \mathbb{N}$. Recalling Lemma 2, we have the following boundedness result

$$\mathscr{C}_{g(F(x_0))}(B_{x_0,x_*}) \ge \max_{i=1,\dots,k} \left\{ \min_{v \in \mathcal{V}(x_{i-1},x_i)} \{\boldsymbol{L}^T v\} \right\}, \forall k \ge 1. \tag{16}$$

The rest of the proof is quite standard. Combining inequalities of the form (15) with the above inequality (16) one obtains

$$g(F(x_j)) - g(F(x_*)) \le \frac{\mathscr{C}_{g(F(x_0))}(B_{x_0,x_*})}{2} (\|x_{j-1} - x_*\|^2 - \|x_j - x_*\|^2), \tag{17}$$

for all $j \in \{1, \dots, k\}$ and for any $x_* \in S$.

Fix $k \ge 1$. Summing up inequality (17) for $j \in \{1, \dots, k\}$ ensures that for any $x_* \in S$, we have

$$\sum_{j=1}^{k} [g \circ F(x_j) - g \circ F(x_*)] \le \frac{\mathscr{C}_{g(F(x_0))}(B_{x_0,x_*})}{2} (\|x_0 - x_*\|^2 - \|x_k - x_*\|^2)$$

$$\le \frac{\mathscr{C}_{g(F(x_0))}(B_{x_0,x_*})}{2} \|x_0 - x_*\|^2. \tag{18}$$

Since the sequence $(g \circ F(x_j))_{j \in \mathbb{N}}$ is nonincreasing, it follows that, for any $x_* \in S$,

$$k[g \circ F(x_k) - g \circ F(x_*)] \le \sum_{j=1}^{k} [g \circ F(x_j) - g \circ F(x_*)]. \tag{19}$$

Substituting inequality (19) into inequality (18), we obtain

$$k[g \circ F(x_k) - g \circ F(x_*)] \le \frac{\mathscr{C}_{g(F(x_0))}(B_{x_0,x_*})}{2} \|x_0 - x_*\|^2, \ \forall x_* \in S.$$

Dividing both sides of this inequality by k indicates that (10) holds. Since k was an arbitrary positive integer, this completes the proof of (i).

(ii) The proof relies on Opial's lemma/monotonicity techniques à la Féjer (see [4]). Using (10) and the lower semicontinuity of $g \circ F$ one immediately proves that cluster points of $(x_k)_{k \in \mathbb{N}}$ are in S. Combining this with the fact that $\|x_k - x_*\|$ is nonincreasing for all $x_* \in S$ concludes the proof.   □

Remark 6 (a) If the function g is globally L_g Lipschitz continuous, one has $\mathscr{C}_\gamma(K) \leq L_g\|L\|$ for any compact set $K \subset \mathbb{R}^n$ and any $\gamma \geq \min g \circ F$, see Sect. 3.2.1.

(b) Consider a minimization problem $(\mathcal{P})$ which has several formulations in the sense that there exist g_1, F_1 and g_2, F_2 such that the objective is given by $g_1 \circ F_1 = g_2 \circ F_2$. Then the complexity results for the two formulations may differ considerably, see Sect. 3.2.4.

(c) Note that the above proof actually yields a more subtle "online" estimate:

$$g(F(x_k)) - g(F(x_*)) \leq \frac{\max_{i=1,\ldots,k}\left\{\min\left\{L^T v : v \in \mathcal{V}(x_{i-1}, x_i)\right\}\right\}}{2k} \|x_0 - x_*\|^2. \tag{20}$$

This shows that the specific history of a sequence plays an important role in its actual complexity. This is of course not captured by global constants of the form $\mathscr{C}_{g(F(x_0))}(B_{x_0,x_*})$ which are worst case estimates.

(d) The complexity estimate of Theorem 1 does not directly involve the constant L_g, but only multipliers. This will be useful to recover existing complexity results for algorithms such as the forward–backward splitting algorithm in Sect. 3.2.4.

Explicit complexity bounds We now provide an explicit bound for the condition number which will in turn provide explicit complexity bounds for Multiprox. Our approach relies on a thorough study of the multipliers and on a measure of the Slater's like assumption through the term

$$\text{dist}[F(\bar{x}), \text{bd dom } g] > 0,$$

whose positivity follows from Assumption 4. Our results on multipliers are recorded in the following fundamental lemma. Its proof is quite delicate and it is postponed in the appendix.

Given a matrix $A \in \mathbb{R}^{m \times n}$, its operator norm is denoted by $\|A\|_{\text{op}}$.

Lemma 4 (Bounds for the multipliers of $(\mathcal{P}_x)$) *For any $x \in F^{-1}(\text{dom } g)$, $y \in p(x)$ and $v \in \mathcal{V}(x, y)$, the following statements hold:*

(i) *if $H(x, y) \in \text{int dom } g$, then $L^T v \leq L_g\|L\|$;*

(ii) *if $H(x, y) \in \text{bd dom } g$, then*

$$L^T v \leq 8L_g \left(\|\nabla F(x)\|_{\text{op}} + (3\|x - y\| + \|\bar{x} - x\|)\frac{\|L\|}{2} \right)^2$$

$$\times \frac{\text{dist}[F(\bar{x}), \text{bd dom } g] + \|L\|\|\bar{x} - x\|^2/2}{\text{dist}[F(\bar{x}), \text{bd dom } g]^2}$$

where $\bar{x}$ is as in Assumption 4.

The full proof of this Lemma is postponed to "Appendix B". A pretty direct consequence of Lemma 4 is a complexity result with explicit constants.

Theorem 2 (Explicit complexity bound for Multiprox) *Let* $(x_k)_{k \in \mathbb{N}}$ *be a sequence generated by* Multiprox. *Then, for any* $x_* \in S$ *and for all* $k \geq 1$,

$$g(F(x_k)) - g(F(x_*)) \leq \max \left\{ \|\boldsymbol{L}\|, \gamma \right\} \frac{L_g \|x_0 - x_*\|^2}{2k}, \tag{21}$$

where

$$\gamma = 8 \left(\|\nabla F(x_0)\|_{\mathrm{op}} + \frac{\|\boldsymbol{L}\|}{2} \left(11\|x_0 - x_*\| + \|\bar{x} - x_*\| \right) \right)^2$$
$$\times \frac{d_{\bar{x}} + \frac{\|\boldsymbol{L}\|}{2} \left(\|x_* - \bar{x}\| + \|x_* - x_0\| \right)^2}{d_{\bar{x}}^2}$$

with $d_{\bar{x}} = \mathrm{dist}[F(\bar{x}), \mathrm{bd} \, \mathrm{dom} \, g]$.

Proof Fix $k > 0$ in $\mathbb{N}$. Recall that $\|x_k - x_*\| \leq \|x_0 - x_*\|$ (see the proof of Theorem 1). The bound on the operator norm of the Jacobian is then computed as follows:

$$\begin{aligned}
\|\nabla F(x_k)\|_{\mathrm{op}} &= \|\nabla F(x_0) + \nabla F(x_k) - \nabla F(x_0)\|_{\mathrm{op}} \\
&\leq \|\nabla F(x_0)\|_{\mathrm{op}} + \|\nabla F(x_k) - \nabla F(x_0)\|_{\mathrm{op}} \\
&\leq \|\nabla F(x_0)\|_{\mathrm{op}} + \|\boldsymbol{L}\| \|x_k - x_0\| \\
&\leq \|\nabla F(x_0)\|_{\mathrm{op}} + 2\|\boldsymbol{L}\| \|x_* - x_0\|,
\end{aligned}$$

where we have used the fact that row i of ∇F is L_i Lipschitz, for $i = 1, \ldots, m$, implying that ∇F is $\|\boldsymbol{L}\|$ Lipschitz with respect to the Frobenius norm. One concludes by using Theorem 1 and Lemma 4. $\qquad\square$

3.2 Consequences of the main result

3.2.1 Complexity for Lipschitz continuous models

It is very useful to make the following elementary observation:

$$\begin{aligned}
\mathscr{C}_\gamma(K) &\leq \sup\{\boldsymbol{L}^T \nu : x \in K, \, g(F(x)) \leq \gamma, \, y \in p(x), \, \nu \in \mathcal{V}(x, y)\} \\
&\leq \sup\{\boldsymbol{L}^T \lambda : x \in K, \, g(F(x)) \leq \gamma, \, y \in p(x), \, \lambda \in \partial g(H(x, y))\} \\
&\leq \sup\{\boldsymbol{L}^T \lambda : z \in \mathrm{dom} \, g, \, \lambda \in \partial g(z)\}, \tag{22}
\end{aligned}$$

where the first inequality is because we just removed an infimum from the definition of $\mathscr{C}_\gamma(K)$ in Lemma 2, the second follows because for any $x \in F^{-1}(\mathrm{dom} \, g)$ and $y \in p(x)$, $\mathcal{V}(x, y) \subset \partial g(H(x, y))$, and the third follows because for any $x \in F^{-1}(\mathrm{dom} \, g)$ and any $y \in p(x)$, we have $H(x, y) \in \mathrm{dom} \, g$. The above upper bound is finite whenever g has full domain and is globally Lipschitz continuous. Indeed, in that case $\sup_{z,\lambda}\{\|\lambda\| : \lambda \in \partial g(z)\} \leq L_g$ (see e.g., [41, Theorem 9.13]). Assumptions 3 and 4 are

automatically satisfied. An immediate application of the Cauchy–Schwartz inequality leads to the following bound for the condition number

$$\mathscr{C}_{g(F(x_0))}(B_{x_0,x_*}) \leq L_g \|L\|, \quad \text{for all feasible data } x_0, x_*. \tag{23}$$

Thus we have the general result for composite Lipschitz continuous problems:

Corollary 2 (Global complexity for Lipschitz continuous model) *In addition to Assumptions* 1 *and* 2, *suppose that g is finite valued, and thus globally Lipschitz continuous with Lipschitz constant* L_g. *Let* $(x_k)_{k\in\mathbb{N}}$ *be a sequence generated by* Multiprox, *then it converges to a minimizer and for any* $x_* \in S$:

$$g(F(x_k)) - g(F(x_*)) \leq \frac{L_g \|L\|}{2k} \|x_0 - x_*\|^2, \quad \forall k \geq 1. \tag{24}$$

Remark 7 (a) Note that instead of using the Cauchy–Schwartz inequality, one could use Hölder's inequality if g is Lipschitz with respect to a different norm. For example, suppose that each coordinate of g is L_g Lipschitz continuous (the others being fixed), or in other words that the supremum norm of the subgradients of g is bounded by L_g. In this case, a result similar to (27) holds with $\|L\|_1$ in place of the Euclidean norm.

(b) The bound given above is sharper than the general bound provided in Theorem 2.

3.2.2 Proximal Gauss–Newton's method for min–max problems

Let us illustrate how Theorem 1 can give new insights into the proximal Gauss–Newton method (PGNM) when g is a componentwise maximum. As in Sect. 2.3.2 consider $g(z) = \max(z_1, \ldots, z_m)$ for z in $\mathbb{R}^m$. Take F as in Assumption 1, set $L = \max_{i\in\{1,\ldots,m\}} L_i$ and

$$L = (L, \ldots, L)^T. \tag{25}$$

Multiprox writes:

$$x_{k+1} = \underset{y\in\mathbb{R}^n}{\operatorname{argmin}} \max_{i\in\{1,\ldots,m\}} \left\{ f_i(x_k) + \nabla f_i(x_k)^T (y - x_k) \right\} + \frac{L}{2} \|y - x_k\|^2, \tag{26}$$

which is nothing else than PGNM applied to the problem $(\mathcal{P})$.

The kernel g is 1 Lipschitz continuous with respect to the L^1 norm. As in Corollary 2, a straightforward application of Hölder's inequality leads to the following complexity result:

Corollary 3 (Complexity for PGNM) *In addition to Assumptions* 1, 2, *suppose that g is the componentwise maximum. Let* $(x_k)_{k\in\mathbb{N}}$ *be a sequence generated by PGNM, then for any* $k \geq 1$ *and any* x_* *in S, we have*

$$g(F(x_k)) - g(F(x_*)) \leq \frac{L}{2k} \|x_0 - x_*\|^2, \tag{27}$$

furthermore, the sequence $(x_k)_{k\in\mathbb{N}}$ *converges to a solution of* $(\mathcal{P})$.

Remark 8 (a) As far as we know, this complexity result for the classical PGNM is new. We suspect that similar results could be derived for much more general kernels g, this is a matter for future research.

(b) Note that the accelerated algorithm for PGNM described in [16] would achieve a convergence rate of the form $2\sqrt{m}L\|x_0 - x_*\|^2/k^2$. Indeed, the multiplicative constant appearing in the convergence rate of [16, Theorem 8.5] involves the Lipschitz constant of ∇F^T measured in term of operator norm. We do not know if the constant $\sqrt{m}$ (which can be big for some problems) could be avoided, and thus, at this stage of our understanding, we cannot draw any comparative conclusion between these two complexity results.

(c) Although the worst-case complexity estimate of Multiprox is the same as the one for PGNM, we have observed a dramatic difference in practice, see Sect. 5. The intuitive reason is quite obvious since Multiprox is much more adapted to the geometry of the problem. Better performances might also be connected to the estimate (20) given in Remark 6.

3.2.3 Complexity of the Moving balls method

We provide here an enhanced nonsmooth version of the moving balls method, introduced in Sect. 2.3.1, which allows to handle sparsity constraints. Consider the following nonlinear convex programming problem:

$$\min_{x \in \mathbb{R}^n} f(x) + h(x)$$
$$\text{s.t. } f_i(x) \leq 0, \ i = 1, \ldots, m. \tag{28}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex, differentiable with L_f Lipschitz continuous gradient, each $f_i : \mathbb{R}^n \to \mathbb{R}$ is defined as in Assumption 1, and $h : \mathbb{R}^n \to \mathbb{R}$ is a convex lower semicontinuous function, for instance $h = \|\cdot\|_1$.

Choosing g and F adequately (details can be found in the proof of Corollary 4), Multiprox gives an algorithm combining/improving ideas presented in [2,46][7]:

$$x_{k+1} \in \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \quad f(x_k) + \nabla f(x_k)^T(y - x_k) + \frac{L_f}{2}\|y - x_k\|^2 + h(y)$$
$$\text{s.t.} \quad f_i(x_k) + \nabla f_i(x_k)^T(y - x_k) + \frac{L_i}{2}\|y - x_k\|^2 \leq 0, \ i = 1, \ldots, m. \tag{29}$$

Our main convergence result in Theorem 1 can be combined with Lemma 2 to recover and extend the convergence results of [2,46]. More importantly we derive explicit complexity bounds of the form $O(\frac{1}{k})$. We are not aware of any such quantitative result for general nonlinear programming problems.

Corollary 4 (Complexity of the moving balls method) *Assume that h is L_h Lipschitz continuous and that there exists $\bar{x}$ in $\mathbb{R}^n$ such that $f_i(\bar{x}) < 0$, $i = 1, \ldots, m$. Assume*

---

[7] Observe that the subproblems are simple convex quadratic problems.

that (28) has a solution x^* and that there exists i in $\{1, \ldots, m\}$ such that $L_i > 0$. Let $(x_k)_{k \in \mathbb{N}}$ be a sequence generated by Multiprox. *Then for any $k \geq 1$, x_k is feasible[8] and*

$$(f + h)(x_k) - (f + h)(x_*) \leq \max\left\{1, \zeta\right\} \frac{\|\boldsymbol{L}\|(1 + L_h)\|x_0 - x_*\|^2}{2k}, \qquad (30)$$

where $\boldsymbol{L} = (L_f, L_1, \ldots, L_m, 0, \ldots, 0)^T \in \mathbb{R}^{n+m+1}$ and

$$\zeta = 8\|\boldsymbol{L}\| \left(\frac{\|\nabla F(x_0)\|_{\mathrm{op}}}{\|\boldsymbol{L}\|} + \frac{1}{2}\left(11\|x_0 - x_*\| + \|\bar{x} - x_*\|\right)\right)^2$$
$$\times \frac{\bar{f} + \frac{\|\boldsymbol{L}\|}{2}\left(\|x_* - \bar{x}\| + \|x_* - x_0\|\right)^2}{\bar{f}^2}$$

with $\bar{f} = |\max_{i=1,\ldots,m}\{f_i(\bar{x})\}|$.

Proof We set $F \colon x \mapsto (f(x), f_1(x), \ldots, f_m(x), x)$ and $g \colon (z_0, z_1, \ldots, z_m, z) \mapsto z_0 + h(z) + \sum_{i=1}^{m} i_{\mathbb{R}_-}(z_i)$. With this choice, we obtain problem (28) and algorithm (29) (we set the smoothness modulus of the identity part in F to 0, whence the value of $\boldsymbol{L}$). Assumptions 1, 2 and 4 are clearly satisfied. Assumption 3 is satisfied since one of the L_i is positive, indicating that the subproblems in (29) are strongly convex and have bounded constraint sets. Finally g is $(1 + L_h)$ Lipschitz continuous on its domain. Hence Theorem 2 can be applied. It remains to notice that $\bar{f} = \mathrm{dist}[F(\bar{x}), \mathrm{bd\,dom}\,g]$ to conclude the proof.   □

3.2.4 Forward–backward splitting algorithm

To illustrate further the flexibility of our method, we explain how our approach allows to recover the classical complexity results of the classical forward–backward splitting algorithm within the convex setting. Let $f \colon \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable convex function with L Lipschitz gradient and $h \colon \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper lower semicontinuous convex function. Consider the following problem

$$\inf_{x \in \mathbb{R}^n} f(x) + h(x). \qquad (31)$$

This problem is a special case of the optimization objective of problem $(\mathcal{P})$, by choosing

$$F : \begin{cases} \mathbb{R}^n \to \mathbb{R}^{n+1} \\ x \quad \to (f(x), x) \end{cases} \qquad (32)$$

and

$$g : \begin{cases} \mathbb{R} \times \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\} \\ (a, z) \quad \to a + h(z) \end{cases}$$

---

[8] For the original problem (28).

Finally, setting

$$L = \begin{pmatrix} L \\ 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^{n+1}, \tag{33}$$

Multiprox eventually writes:

$$x_{k+1} = \underset{y \in \mathbb{R}^n}{\mathrm{argmin}} \quad f(x_k) + \nabla f(x_k)^T (y - x_k) + h(y) + \frac{L}{2} \|y - x_k\|^2, \tag{34}$$

which is exactly the forward–backward splitting algorithm. It is immediate to check that Assumptions 1, 2, 3 and 4 hold true as long as the minimum is achieved in (31) and that dom h has nonempty interior with h being Lipschitz continuous on dom h.[9] Given the form of g in Eq. (32) and L in (33), Theorem 1 yields the classical convergence and complexity results for the forward–backward algorithm (see e.g., [11] for convergence and [5] for complexity): x_k converges to a minimizer and

$$(f + h)(x_k) - (f + h)(x_*) \leq \frac{L\|x_0 - x_*\|^2}{2k}, \quad \forall k \geq 1. \tag{35}$$

Remark 9 (Complexity estimates depend on the formulation) Assume that h is the indicator function of a ball $B(a, r)$ so that the above method is the gradient projection method on this ball and its complexity is recovered by Eq. (35). Another way of modeling the problem is to consider minimizing $g_1 \circ F_1$ taking the following forms

$$F_1 : \mathbb{R}^n \to \mathbb{R}^2$$
$$x \to \begin{pmatrix} f(x) \\ f_1(x) \end{pmatrix},$$

where $f_1(x) = \|x - a\|^2 - r^2$, and

$$g_1 : \mathbb{R}^2 \to \mathbb{R} \cup \{+\infty\}$$
$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \to z_1 + h_1(z_2),$$

where h_1 is the indicator function of $\mathbb{R}_-$, so that for every $x \in \mathbb{R}^n$ it holds $g \circ F(x) = g_1 \circ F_1(x)$ and Multiprox for $g_1 \circ F_1$ is equivalent to the moving balls method. Since

$$f_1(x) + \nabla f_1(x)^T (y - x) + \frac{L_1}{2} \|y - x\|^2 = f_1(y), \quad \forall x, y \in \mathbb{R}^n,$$

---

[9] Lipschitz continuity is actually superfluous for Theorem 1 to hold.

where $L_1 = 2$ is the Lipschitz constant of the gradient of f_1, it follows that for any $x \in \mathbb{R}^n$ it holds

$$\underbrace{\operatorname*{argmin}_{y \in \mathbb{R}^n} \left(\frac{1}{2} \left\| y - x + \frac{\nabla f(x)}{L} \right\|^2 + h(y) \right)}_{\text{gradient projection method}} = \underbrace{\operatorname*{argmin}_{y \in \mathbb{R}^n} \left(\frac{1}{2} \left\| y - x + \frac{\nabla f(x)}{L} \right\|^2 + h_1(f_1(y)) \right)}_{\text{moving balls method}}.$$

Thus, if the initial point is the same, the sequence of the moving balls method is the same as that of gradient projection method. However, considering the third item of Remark 6, the best estimate our analysis can provide for the moving-balls method is

$$
\begin{aligned}
(f + h_1)(x_k) &- (f + h_1)(x_*) \qquad\qquad\qquad\qquad\qquad\qquad (36) \\
&\leq \frac{\max\limits_{i=1,\dots,k} \left\{ L + 2\min\{v_1 \in \mathbb{R} : (1, v_1) \in \mathcal{V}(x_{i-1}, x_i)\} \right\}}{2k} \|x_0 - x_*\|^2, \ \forall k \geq 1.
\end{aligned}
$$

This is different from the complexity of the gradient projection method in (35). Indeed the infimum over the variable v_1 appearing in the numerator is nonnegative. Furthermore it is non zero in many situations because otherwise the constraints would never been binding. As an example, one can consider a linear objective function for which the infimum in the numerator in (36) is strictly positive.

Observe that the numerator in (36) is strictly greater than L which is the classical constant attached the projected gradient, even though both algorithms are actually the same. This highlights the second item of Remark 6 on the dependance of the estimate on the choice of equivalent composite models.

4 Backtracking and linesearch

In practice, the collection L of Lipschitz constants may not be known or efficiently computable. Lipschitz continuity might not even be global. To handle these fundamental cases, we provide now our algorithmic scheme in (5) with a linesearch procedure (see e.g., [5,35]).

First let us define a space search for our steps[10]

$$\Gamma := \left\{ (\alpha_1, \dots, \alpha_m) \in \mathbb{R}_+^m : \alpha_i = 0 \text{ if } L_i = 0, \ \alpha_i > 0 \text{ if } L_i > 0, \ i = 1, \dots, m \right\},$$

and for every $\boldsymbol{\alpha} \in \Gamma$ we set

$$H_{\boldsymbol{\alpha}}(x, y) := F(x) + \nabla F(x)(y - x) + \frac{\boldsymbol{\alpha}}{2} \|y - x\|^2, \ \forall (x, y) \in \mathbb{R}^n \times \mathbb{R}^n.$$

In order to design an algorithm with this larger family of surrogates, we need a stronger version of Assumption 3.

---

[10] Actually the inverse of our steps.

Assumption 5 For any $x \in F^{-1}(\operatorname{dom} g)$ and every $\boldsymbol{\alpha} \in \Gamma$, the function $g \circ H_{\boldsymbol{\alpha}}(x, \cdot)$ has a minimizer.

For every $x \in F^{-1}(\operatorname{dom} g)$, the basic subproblem we shall use is defined for any $\boldsymbol{\alpha} \in \Gamma$,

$$\left(\mathcal{P}_{\boldsymbol{\alpha}, x}\right): \quad p_{\boldsymbol{\alpha}}(x) := \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \ g \circ H_{\boldsymbol{\alpha}}(x, y).$$

The Multiprox algorithm with backtracking step sizes is defined as:

<div style="border:1px solid">

Multiproximal method with backtracking step sizes

Take $x_0 \in F^{-1}(\operatorname{dom} g)$, $\boldsymbol{\alpha}_0 \in \Gamma$ and $\eta > 1$. Then iterate for $k \in \mathbb{N}$:

step 1. set $\tilde{\boldsymbol{\alpha}} = \boldsymbol{\alpha}_k$, $\tilde{x} \in p_{\tilde{\boldsymbol{\alpha}}}(x_k)$.

step 2. while the inequality $H_{\tilde{\boldsymbol{\alpha}}}(x_k, \tilde{x}) > F(\tilde{x})$ is not satisfied :
   for $i = 1, \ldots, m$:
      if $f_i(\tilde{x}) > f_i(x_k) + \nabla f_i(x_k)^T(\tilde{x} - x_k) + \frac{\tilde{\alpha}_i}{2}\|\tilde{x} - x_k\|^2$:
         set $\tilde{\alpha}_i \leftarrow \eta \tilde{\alpha}_i$
   set $\tilde{x} \in p_{\tilde{\boldsymbol{\alpha}}}(x_k)$.
   set $\boldsymbol{\alpha}_{k+1} = \tilde{\boldsymbol{\alpha}}$.

step 3. if $x_k \in p_{\boldsymbol{\alpha}_{k+1}}(x_k)$:
      $x_{k+1} = x_k$
   else :
      $x_{k+1} = \tilde{x}$

(37)

</div>

Remark 10 (A finite while-loop)

(a) One needs to make sure that the scheme is well defined and that each while-loop stops after finitely many tries. Under Assumption 1, it is naturally the case. To see this let $(\boldsymbol{\alpha}_k)_{k \in \mathbb{N}}$ be a sequence generated by the scheme in (37), and let $L \in \mathbb{R}_+^m$ be the collection of Lipschitz constants associated to ∇F. Then, for every integer k, the following statements must obviously hold for each $i = 1, \ldots, m$:

$$(\boldsymbol{\alpha}_k)_i \leq \max\{\eta L_i, (\boldsymbol{\alpha}_0)_i\}.$$

(b) More importantly we shall also see that local Lipschitz continuity and coercivity also ensure that the while-loop is finite, see Theorem 4 below.

Arguments similar to those of Sect. 2.4 allow to derive chain rules and to eventually consider the following sets (note that a proposition equivalent to Proposition 2 holds for the problem $\left(\mathcal{P}_{\boldsymbol{\alpha}, x}\right)$).

Definition 2 *(Lagrange multipliers of the subproblems)* Given any fixed point $x \in F^{-1}(\operatorname{dom} g)$, any $\boldsymbol{\alpha} \in \Gamma$ and any $y \in p_{\boldsymbol{\alpha}}(x)$, we set

$$\mathcal{V}_{\boldsymbol{\alpha}}(x, y) := \{v \in g(H_{\boldsymbol{\alpha}}(x, y)) : \nabla_y H_{\boldsymbol{\alpha}}(x, y)^T v = 0\}.$$

We are now able to extend Theorem 1 to a larger setting: Lipschitz constants do exist but they are unknown to the user.

Theorem 3 (Multiprox with backtracking) *Suppose that Assumptions 1, 2, 4 and 5 hold. Let $(x_k)_{k \in \mathbb{N}}$ and $(\boldsymbol{\alpha}_k)_{k \in \mathbb{N}}$ be any sequences generated by the algorithmic scheme in (37). Then, for every $x_* \in S$ and any sequence $(v_k)_{k \in \mathbb{N}}$ such that $v_j \in \mathcal{V}_{\boldsymbol{\alpha}_j}(x_{j-1}, x_j)$ for all $j \geq 1$, one has,*

$$g \circ F(x_k) - g \circ F(x_*) \leq \frac{\max\limits_{j=1,\ldots,k} \{\hat{\boldsymbol{\alpha}}^T v_j\}}{2k} \|x_0 - x_*\|^2, \ \forall k \geq 1, \tag{38}$$

where $\hat{\boldsymbol{\alpha}} \in \Gamma$ is the vector
whose entries are given by the upper bound in Remark 10. Furthermore, the sequence $(x_k)_{k \in \mathbb{N}}$ converges to a point in S.

Proof The proof is in the line of that of Theorem 1. We observe first that we have a descent method. Let $(x_k)_{k \in \mathbb{N}}$ and $(\boldsymbol{\alpha}_k)_{k \in \mathbb{N}}$ be sequences generated by the algorithmic scheme in (37). Fix $k \geq 1$. One has $F(x_k) \leq H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k)$ as the while-loop stops after finitely many steps (see Remark 10). Using the monotonicity properties of g, one deduces that $g(F(x_k)) \leq g(H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k))$. Considering that x_k is a minimizer of $g(H_{\boldsymbol{\alpha}_k}(x_{k-1}, \cdot))$, it follows further that $g(H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k)) \leq g(H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_{k-1})) = g(F(x_{k-1}))$, indicating that the algorithm is a descent method.

For any v_k in $\mathcal{V}_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k)$ and any x_* in S, one has

$$\begin{aligned}
&g \circ F(x_k) - g \circ F(x_*) \\
&\stackrel{(a)}{\leq} g \circ H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k) - g \circ F(x_*) \\
&\stackrel{(b)}{\leq} [H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k) - F(x_*)]^T v_k \\
&\stackrel{(c)}{=} H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_*)^T v_k - F(x_*)^T v_k - \frac{\boldsymbol{\alpha}_k^T v_k}{2} \|x_k - x_*\|^2 \\
&\stackrel{(d)}{=} [F(x_{k-1}) + \nabla F(x_{k-1})(x_* - x_{k-1})]^T v_k - F(x_*)^T v_k \\
&\quad + \frac{\boldsymbol{\alpha}_k^T v_k}{2}(\|x_{k-1} - x_*\|^2 - \|x_k - x_*\|^2) \\
&\stackrel{(e)}{\leq} \frac{\boldsymbol{\alpha}_k^T v_k}{2}(\|x_{k-1} - x_*\|^2 - \|x_k - x_*\|^2),
\end{aligned}$$

where: (a) is obtained by combining the descent property with $F(x_k) \leq H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k)$, (b) follows from the convexity of g and the fact that $v_k \in \mathcal{V}_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k)$, (c) is obtained by combining $[\nabla_y H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k)]^T v_k = 0$ with Eq. (11), (d) is an explicit expansion of $H_{\boldsymbol{\alpha}_k}(x_{k-1}, x_k)^T v_k$ explicitly, and eventually, (e) stems from the property that the i-th coordinate of v_k is nonnegative if $L_i > 0$ (cf. Assumption 1), the construction of Γ and the coordinatewise convexity of F. We therefore obtain:

$$g(F(x_k)) - g(F(x_*)) \leq \frac{C_k}{2k} \|x_0 - x_*\|^2, \ \ k \geq 1, \tag{39}$$

where $C_k = \max\limits_{j=1,\ldots,k} \min\{\boldsymbol{\alpha}_j{}^T \nu : \nu \in \mathcal{V}_{\boldsymbol{\alpha}_j(x_{j-1},x_j)}\}$ is a bounded sequence.

The convergence of the sequence $(x_k)_{k\in\mathbb{N}}$ follows by similar arguments as in the proof of Theorem 1. $\qquad\square$

We now consider the fundamental and ubiquitous case when global Lipschitz constants do not exist but exist locally. A typical case of such a situation is given by problems involving C^2 mappings F (local Lipschitz continuity follows indeed from a direct application of the mean value theorem to the ∇f_i, $i = 1, \ldots, m$). To compensate for this lack of global Lipschitz continuity, we make the following coercivity assumption:

Assumption 6 The problem $g \circ F$ is coercive and, for any $x \in F^{-1}(\operatorname{dom} g)$, the function $g \circ H_{\boldsymbol{\alpha_0}}(x, \cdot)$ is coercive.

Theorem 4 (Convergence without any global Lipschitz condition) *Suppose that Assumptions 2, 4 and 6 hold and that Assumption 1 is weakened in the sense that we assume that each ∇f_i is only locally Lipschitz continuous. Then any sequence $(x_k)_{k\in\mathbb{N}}$ generated by (37) converges to a single point in the solution set S.*

Proof The first and main point to be observed is that the while-loop is finite. Fix $k \geq 1$. Observe that since $\tilde{\alpha} \geq \alpha_0$ any of the sublevel sets of the subproblems in the while loop are contained in a common compact set $K := \{y \in \mathbb{R}^n : g \circ H_{\boldsymbol{\alpha_0}}(x_k, y) \leq g \circ F(x_k))\}$. Since each gradient is Lipschitz continuous globally on K the while loop ends in finitely many runs. Using arguments similar to those previously given allows to prove that the sequence x_k is a descent sequence. As a consequence it evolves in set $\{y \in \mathbb{R}^n : g(F(y)) \leq g(F(x_0))\}$ which is a compact set by coercivity. Since on this set ∇F is Lipschitz continuous, usual arguments apply. One can thus prove that the sequence converges as previously. $\qquad\square$

Remark 11 "Complexity estimates" could also be derived but the constants appearing in the estimates would be unknown a priori. In a numerical perspective, they could be updated online and used to forecast the efficiency of the method step after step.

5 Numerical experiments

In this section, we present numerical experiments which illustrate the performance of the proposed algorithm on some collections of min–max problems. The proposed method is compared to both the Proximal Gauss–Newton method (PGNM) and an accelerated variant [16, Algorithm 8], which we refer to as APGNM. In this setting the convergence rate for both Multiprox and PGNM is of the order $O(1/k)$, while it is of the order of $O(1/k^2)$ for APGNM (see [16] and the discussion in Sect. 3.2.2).

5.1 Problem class and algorithms

We consider the problem of minimizing the maximum of finitely many quadratic convex functions:

$$
\begin{aligned}
&\min_{x\in\mathbb{R}^n} \quad \max\{f_1(x),\ldots,f_m(x)\},\\
&\text{with} \quad f_i: x \to x^T Q_i x + b_i^T x + c_i \quad \text{for } i = 1,\ldots,m,
\end{aligned}
\tag{40}
$$

where $Q_1,\ldots,Q_m$ are real $n \times n$ positive semidefinite matrices, $b_1,\ldots,b_m$ are in $\mathbb{R}^n$, and $c_1,\ldots,c_m$ in $\mathbb{R}$. Choosing g as the coordinatewise maximum and each coordinate of F to be one of the f_i, $i = 1,\ldots,m$, one sees that this problem is of the form of $(\mathcal{P})$. Assumptions 1 and 4 are satisfied and we assume that the data of the problem (40) are chosen so that Assumption 2 holds. It is easily checked that Assumption 3 holds. Under these conditions, all the results established in Sects. 2 and 3 hold for problem (40).

We consider two choices for the vector $\boldsymbol{L}$,

$$
\boldsymbol{L}_{\text{GN}} = \big(\tilde{L},\ldots,\tilde{L}\big) \text{ and } \boldsymbol{L}_{\text{MProx}} = (L_1,\ldots,L_m)
$$

where $\tilde{L} := \max\{L_1,\ldots,L_m\}$. By replacing $\boldsymbol{L}$ in Multiprox with $\boldsymbol{L}_{\text{GN}}$, we recover the PGNM method. Similarly, by replacing $\boldsymbol{L}$ in Multiprox with $\boldsymbol{L}_{\text{MProx}}$, we have the Multiprox algorithm. In both cases, the subproblem $(\mathcal{P}_x)$ writes

$$
\begin{aligned}
x_{k+1} \in \underset{y\in\mathbb{R}^n, s\in\mathbb{R}}{\operatorname{argmin}} \quad & s\\
\text{s.t.} \quad & s \geq f_i(x_k) + \langle \nabla f_i(x_k), y - x_k \rangle + \frac{L_i'}{2}\|y - x_k\|^2, \ i = 1,\ldots,m,
\end{aligned}
\tag{41}
$$

the choice of L_i', $i = 1,\ldots,m$, being the only difference between the two methods. Problem (41) is a quadratically constrained quadratic program (QCQP) which can be solved by appropriate solvers.

As for APGNM, the accelerated variant of PGNM, it requires to solve iteratively QCQP subproblems which are analogous to (41), their solutions can be computed similarly. However APGNM requires to solve two QCQP subproblems at each step (we refer the interested readers to [16] for more details on this algorithm).

5.2 Performance comparison

In this section, we provide numerical comparison of the performances of Multiprox, PGNM, and APGNM on problem (40), with synthetic, randomly generated data. First, let us explain the data generation process. The matrix Q_m is set to zero so that the resulting component, f_m is actually an affine function (and L_m is set to 0). The rest of problem data is generated as follows. For $i = 1,\ldots,m-1$ we generate positive

semidefinite matrices $Q_i = Y_i D_i Y_i$, where Y_i are random Householder orthogonal matrices

$$Y_i = I_n - 2\frac{\omega_i \omega_i^T}{\|\omega_i\|^2},$$

where I_n is the $n \times n$ identity matrix and the coordinates of $\omega_i \in \mathbb{R}^n$ are chosen as independent realizations of a unit Gaussian. For $i = 1, \ldots, m - 1$, D_i is chosen as an $n \times n$ diagonal matrix whose diagonal elements are randomly shuffled from the set

$$\{i \times 10^{\frac{j}{n-1}}, j = 1, \ldots, n\}.$$

For $i = 1, \ldots, m$, the coordinates of b_i are chosen as independent realizations of a Gaussian $N(0, (1/3)^2)$. Finally, we choose

$$c_i = 10^{2i/m}.$$

Note that for each $i = 1, \ldots, m$, the Lipschitz constant of f_i is twice the maximum eigenvalue of Q_i, i.e.,

$$L_i = 2 \times \max\{\text{eigenvalue}(Q_i)\}.$$

To iteratively solve the surrogate QCQP subproblems, we used *MOSEK* solver [31] interfaced with *YALMIP* toolbox [27] in Matlab. We fix the number of variables to $n = 100$ and choose the origin as the initial point. For each $m \in \{5, 10, 15, 20, 25, 30\}$, we repeat the random data generation process 20 times and run the three algorithms to solve each corresponding problem.

As a measure of performance, in order to compare efficiency between different random runs of the data generation process, we use the following normalized suboptimality gap

$$\left(\frac{g \circ F(x_k) - g \circ F(x_*)}{g \circ F(x_0) - g \circ F(x_*)}\right)_{k \in \mathbb{N}}.$$

Statistics for the three algorithms are presented in Table 1. It is clear from these figures, that Multiprox is dramatically faster than both PGNM and APGNM. Furthermore, Multiprox seems to suffer less from increasing values of m. Finally APGNM is less consistent in terms of performances.

A graphical view of the same results is presented in Fig. 1 and a log-scale view is given in Fig. 2. One can see from Figs. 1 and 2 that the sequences for Multiprox and PGNM are nonincreasing as predicted by our theory. Note that the sequence generated by APGNM is not necessarily nonincreasing, although all the sequences represented in Fig. 1 are strictly decreasing. It is clear that the decreasing slopes for PGNM and APGNM are much smaller than that of Multiprox, coinciding with the data in Table 1. This situation is actually not surprising since the data of the problem were chosen in a way ensuring that the L_i's $(i = 1, \ldots, m)$ can take very different values as in many

Table 1 Comparisons of Multiprox, PGNM, and APGNM in terms of normalized suboptimality gap in percentage at 10th and 20th iterations

	$100 \times \frac{g \circ F(x_k) - g \circ F(x_*)}{g \circ F(x_0) - g \circ F(x_*)}$					
	Multiprox		PGNM		APGNM	
	$k = 10$		$k = 10$		$k = 10$	
m	Mean	SD	Mean	SD	Mean	SD
5	0.48	1.12×10^{-1}	94.44	5.01×10^{-1}	88.15	1.07
10	0.46	9.81×10^{-2}	95.20	3.34×10^{-1}	89.78	7.14×10^{-1}
15	0.47	8.49×10^{-2}	95.62	2.84×10^{-1}	90.66	6.05×10^{-1}
20	0.47	8.73×10^{-2}	95.72	3.71×10^{-1}	90.88	7.92×10^{-1}
25	0.44	1.07×10^{-1}	95.79	4.04×10^{-1}	91.03	8.64×10^{-1}
30	0.46	8.01×10^{-2}	95.82	4.33×10^{-1}	91.09	9.26×10^{-1}
	$k = 20$		$k = 20$		$k = 20$	
m	Mean	SD	Mean	SD	Mean	SD
5	2.24×10^{-2}	5.70×10^{-3}	88.88	1.00	62.34	3.40
10	2.24×10^{-2}	5.86×10^{-3}	90.40	6.68×10^{-1}	67.53	2.27
15	2.15×10^{-2}	5.43×10^{-3}	91.24	5.67×10^{-1}	70.35	1.92
20	2.13×10^{-2}	4.67×10^{-3}	91.44	7.42×10^{-1}	71.03	2.51
25	2.08×10^{-2}	6.45×10^{-3}	91.57	8.10×10^{-1}	71.49	2.74
30	2.07×10^{-2}	5.46×10^{-3}	91.64	8.67×10^{-1}	71.71	2.94

The mean and standard deviations represent the central tendency and dispersion for 20 random runs of the data generation process
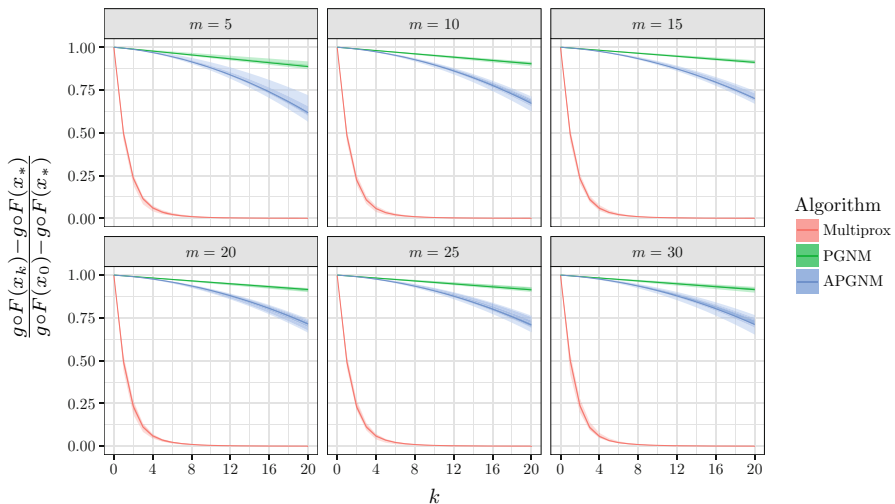

Fig. 1 Comparisons of Multiprox, PGNM, and APGNM in terms of normalized suboptimality gap. The central line is the median over the 20 random runs of the data generation process and the ribbons contain 25%, 50%, 75% and 100% of the simulations respectively
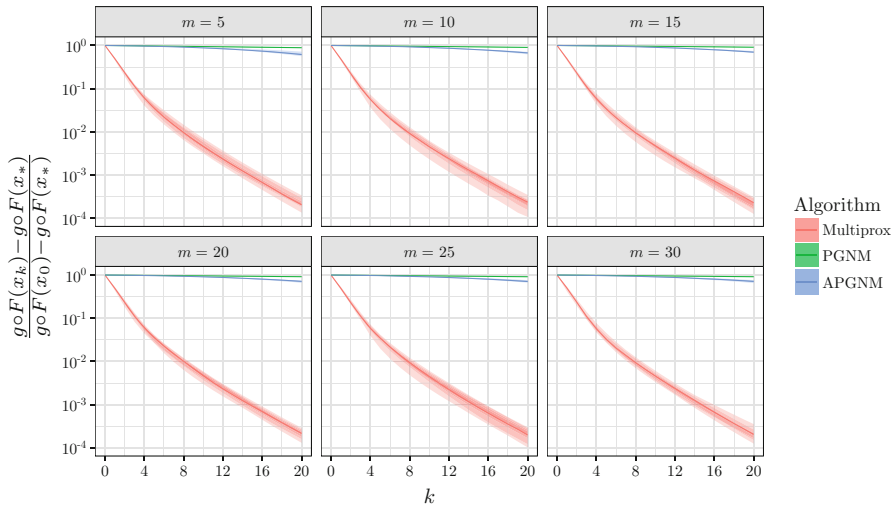
Fig. 2 Same as Fig. 1 with the y axis on log scale

ill posed problems. The strength of Multiprox is that L can be chosen appropriately to adapt to this disparity. On the other hand the use of a single parameter (as in PGNM or APGNM) yields smaller steps and thus slower convergence.

Appendix A: Proof of Proposition 1

Let us recall a qualification condition from [41]. Given any $x \in F^{-1}(\text{dom } g)$, let

$$J(x, \cdot) : \begin{cases} \mathbb{R}^n \to \mathbb{R}^m \\ \omega \mapsto F(x) + \nabla F(x)\omega, \end{cases} \tag{42}$$

be the linearized mapping of F at x. Proposition 1 follows immediately from the classical chain rule given in [41, Theorem 10.6] and the following proposition.

Proposition 3 (Two equivalent qualification conditions) *Under Assumptions 1 on F and g, Assumption 4 holds if and only if*

   (QC) dom g *cannot be separated from* $J(x, \mathbb{R}^n)$ *for any* $x \in F^{-1}(\text{dom } g)$.

Proof We first suppose that **(QC)** is true. We begin with a remark showing that this implies that dom g is not empty. Let A and B be two subsets of $\mathbb{R}^m$. The logical negation of the sentence "A and B can be separated" can be written as follows: for all a in $\mathbb{R}^m$ and for all $b \in \mathbb{R}$, there exists $y \in A$ such that

$$a^T y + b > 0,$$

or, there exists $z \in B$ such that

$$a^T z + b < 0.$$

In particular if A and B cannot be separated, then either A or B is not empty. Note that if dom g is empty, then so is the set $\{J(x, \mathbb{R}^n), x \in F^{-1}(\text{dom } g)\}$. Hence **(QC)** actually implies that dom g is not empty. Pick a point $\tilde{x} \in F^{-1}(\text{dom } g)$. If $F(\tilde{x}) \in \text{int dom}(g)$, there is nothing to prove, so we may suppose that $F(\tilde{x}) \in \text{bd dom } g$. If we had $[\text{int dom}(g)] \cap J(\tilde{x}, \mathbb{R}^n) = \emptyset$, then, dom g and $J(\tilde{x}, \mathbb{R}^n)$ could be separated by Hahn-Banach theorem contradicting **(QC)**. Hence, there exists $\tilde{\omega} \in \mathbb{R}^n$ such that $J(\tilde{x}, \tilde{\omega}) \in \text{int dom}(g)$. Note that, since $F(\tilde{x}) \in \text{dom}(g)$ and g is nondecreasing with respect to each argument, it follows $F(\tilde{x}) - d \in \text{dom}(g)$ for any $d \in (\mathbb{R}_+^*)^m$, indicating that $\text{int}(\text{dom}(g)) \neq \emptyset$. Since dom g is convex, a classical result yields

$$J(\tilde{x}, \lambda \tilde{\omega}) \in \text{int dom}(g), \ \forall \lambda \in (0, 1]. \tag{43}$$

On the other hand F is differentiable thus

$$\|F(\tilde{x} + \lambda \tilde{\omega}) - J(\tilde{x}, \lambda \tilde{\omega})\| = o(\lambda), \tag{44}$$

where $o(\lambda)/\lambda$ tends to zero as λ goes to zero.

After these basic observations, let us recall an important property of the signed distance (see [19, p. 154]). Let $D \subset \mathbb{R}^m$ be a nonempty closed convex set. Then, the function

$$D \to \mathbb{R}_+, \ z \mapsto \text{dist}(z, \text{bd}(D)),$$

is concave. Using this concavity property for $D = \text{dom } g$ and the fact that $F(\tilde{x}) = J(\tilde{x}, 0)$, it holds that

$$\lambda \text{dist}[J(\tilde{x}, \tilde{\omega}), \text{bd}(\text{dom } g)] + (1 - \lambda)\text{dist}[F(\tilde{x}), \text{bd}(\text{dom } g)]$$
$$\leq \text{dist}[J(\tilde{x}, \lambda \tilde{\omega}), \text{bd}(\text{dom } g)], \ \forall \lambda \in [0, 1].$$

Since $\text{dist}[F(\tilde{x}), \text{bd}(\text{dom } g)] = 0$, it follows that

$$\lambda \text{dist}[J(\tilde{x}, \tilde{\omega}), \text{bd}(\text{dom } g)] \leq \text{dist}[J(\tilde{x}, \lambda \tilde{\omega}), \text{bd}(\text{dom } g)], \ \lambda \in [0, 1]. \tag{45}$$

Note that $\text{dist}[J(\tilde{x}, \tilde{\omega}), \text{bd}(\text{dom } g)] > 0$ since $J(\tilde{x}, \tilde{\omega}) \in \text{int dom}(g)$. Hence, Eq. (44) indicates that there exists $\epsilon > 0$ such that for any $0 < \lambda \leq \epsilon$, we have

$$\|F(\tilde{x} + \lambda \tilde{\omega}) - J(\tilde{x}, \lambda \tilde{\omega})\| < \lambda \text{dist}[J(\tilde{x}, \tilde{\omega}), \text{bd}(\text{dom } g)].$$

Substituting this inequality into Eq. (45) indicates that for any $0 < \lambda \leq \epsilon$, we have

$$\|F(\tilde{x} + \lambda \tilde{\omega}) - J(\tilde{x}, \lambda \tilde{\omega})\| < \text{dist}[J(\tilde{x}, \lambda \tilde{\omega}), \text{bd}(\text{dom } g)].$$

Using Eq. (43), for any $0 < \lambda \le \epsilon$, we have $F(\tilde{x} + \lambda\tilde{\omega}) \in \text{int dom}(g)$. This shows the first implication of the equivalence.

Let us prove the reverse implication by contraposition and assume that **(QC)** does not hold, that is, there exists a point $\tilde{x} \in F^{-1}(\text{dom } g)$ such that dom g can be separated from $J(\tilde{x}, \mathbb{R}^n)$. In this case, there exists $a \ne 0 \in \mathbb{R}^m$ and $b \in \mathbb{R}$ such that

$$\begin{cases} a^T z + b \le 0, \ \forall z \in \text{dom } g, \\ a^T J(\tilde{x}, \omega) + b \ge 0, \ \forall \omega \in \mathbb{R}^n. \end{cases} \tag{46}$$

Since $J(\tilde{x}, 0) = F(\tilde{x}) \in \text{dom } g$, it follows

$$a^T J(\tilde{x}, 0) + b = 0. \tag{47}$$

By the coordinatewise convexity of F, for every $i \in \{1, \dots, m\}$ one has

$$f_i(y) \begin{cases} \ge f_i(x) + \nabla f_i(x)^T (y - x), \ L_i > 0, \\ = f_i(x) + \nabla f_i(x)^T (y - x), \ L_i = 0, \end{cases} \quad \forall (x, y) \in \mathbb{R}^n \times \mathbb{R}^n.$$

We thus have the componentwise inequality

$$J(\tilde{x}, 0) - [F(\tilde{x} + \omega) - J(\tilde{x}, \omega)] \le F(\tilde{x}).$$

The monotonicity properties of g implies thus that

$$J(\tilde{x}, 0) - [F(\tilde{x} + \omega) - J(\tilde{x}, \omega)] \in \text{dom } g, \ \forall \omega \in \mathbb{R}^n.$$

As a result, combining Eq. (46) with Eq. (47), one has

$$a^T \{J(\tilde{x}, 0) - [F(\tilde{x} + \omega) - J(\tilde{x}, \omega)]\} + b \le a^T J(\tilde{x}, 0) + b, \ \forall \omega \in \mathbb{R}^n,$$

which reduces to $a^T [F(\tilde{x} + \omega) - J(\tilde{x}, \omega)] \ge 0, \ \forall \omega \in \mathbb{R}^n$. Hence, for any $\omega \in \mathbb{R}^n$ one has

$$a^T F(\tilde{x} + \omega) + b = a^T J(\tilde{x}, \omega) + b + a^T (F(\tilde{x} + \omega) - J(\tilde{x}, \omega))$$
$$\ge a^T J(\tilde{x}, \omega) + b \ge 0,$$

where for the last inequality, Eq. (46) is used. This inequality combined with the fact $a^T z + b < 0, \ \forall z \in \text{int dom } g$ obtained according to the first item of Eq. (46), shows that $F(\tilde{x} + \omega) \notin \text{int dom } g$, for all $\omega \in \mathbb{R}^n$, and thus $F^{-1}(\text{int dom } g) = \emptyset$, that is Assumption 4 does not hold. This provides the reverse implication and the proof is complete. $\qquad \square$

Appendix B: Proof of Lemma 4

In this section, we present an explicit estimate of the condition number appearing in our complexity result. Let us first introduce a notation. For any $D \subset \mathbb{R}^m$, nonempty closed set, we define a signed distance function as

$$D \to \mathbb{R}, \ z \mapsto \mathrm{sdist} = \begin{cases} \mathrm{dist}(z, \mathrm{bd}(D)), & \text{if } z \in \mathrm{int}(D), \\ -\mathrm{dist}(z, \mathrm{bd}(D)), & \text{otherwise.} \end{cases} \tag{48}$$

It is worth recalling that the signed distance function is concave (see [19, p. 154]). We begin with a lemma which describes a monotonicity property of the signed distance function.

Lemma 5 *Given any $z \in \mathrm{dom}\, g$ and any $d = (d_1, \ldots, d_m) \in \mathbb{R}_+^m$ with $d_i = 0$ if $L_i = 0$, if $\mathrm{bd}\, \mathrm{dom} g \neq \emptyset$, one has*

$$\mathrm{sdist}(z, \mathrm{bd}\, \mathrm{dom}\, g) \geq \mathrm{sdist}(z + d, \mathrm{bd}\, \mathrm{dom}\, g). \tag{49}$$

Proof Fix an arbitrary $z \in \mathrm{dom}\, g$ and an arbitrary $d = (d_1, \ldots, d_m) \in \mathbb{R}_+^m$ such that $d_i = 0$ whenever $L_i = 0$ in the sequel of the proof. If $z + d \notin \mathrm{dom}\, g$, Eq. (49) holds true by the definition in Eq. (48).

From now on, we suppose $z + d \in \mathrm{dom}\, g$. Let $\bar{z} \in \mathrm{bd}\, \mathrm{dom}\, g$ be a point such that

$$\bar{z} \in \operatorname*{argmin}_{\hat{z} \in \mathrm{bd}\, \mathrm{dom}\, g} \|z - \hat{z}\|.$$

Then, one has

$$\|z - \bar{z}\| = \mathrm{sdist}(z, \mathrm{bd}\, \mathrm{dom}\, g). \tag{50}$$

Since $\bar{z}$ lies on the boundary of $\mathrm{dom}\, g$, it follows that $\bar{z} + d \notin \mathrm{int}\, \mathrm{dom}\, g$ because of the monotonicity property of g in Assumption 1. Hence, by the definition of sdist in Eq. (48), we have

$$\mathrm{sdist}(z + d, \mathrm{bd}\, \mathrm{dom}\, g) = \mathrm{dist}(z + d, \mathbb{R}^m \setminus \mathrm{int}\, \mathrm{dom}\, g)$$
$$\leq \|(z + d) - (\bar{z} + d)\| = \|z - \bar{z}\|.$$

Combining this inequality with Eq. (50) completes the proof. □

The following lemma shows that it is possible to construct a convex combination between the current x and the Slater point $\bar{x}$ given in Assumption 4 which will be a Slater point for the current sub-problem with a uniform control over the "degree" of qualification.

Lemma 6 *Let $\bar{x}$ be given as in Assumption 4 and $x \in F^{-1}(\text{dom } g)$ and assume that* bd dom$g \neq \emptyset$. *Set*

$$\gamma(\bar{x}, x) := \min\left\{1, \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]}{2\{\text{sdist}[F(\bar{x}), \text{bd dom } g] - \text{sdist}[F(\bar{x}) + \boldsymbol{L}\|x - \bar{x}\|^2/2, \text{bd dom } g]\}}\right\}.$$

Then,

$$\text{sdist}[H(x, x + \gamma(\bar{x}, x)(\bar{x} - x)), \text{bd dom } g] \geq \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]^2}{4\{\text{sdist}[F(\bar{x}), \text{bd dom } g] + \|\boldsymbol{L}\|\|x - \bar{x}\|^2/2\}}. \tag{51}$$

Proof Fix an arbitary $x \in K$. Then, for any $t \in (0, 1]$ one has

$$H(x, x + t(\bar{x} - x)) = F(x) + t\nabla F(x)(\bar{x} - x) + \frac{\boldsymbol{L}}{2}\|\bar{x} - x\|^2 t^2$$

$$\leq (1 - t)F(x) + tF(\bar{x}) + \frac{\boldsymbol{L}}{2}\|x - \bar{x}\|^2 t^2, \tag{52}$$

where the last inequality is obtained by applying the coordinatewise convexity of F. Therefore, for any $t \in (0, 1]$, we have

$$\begin{aligned}
&\text{sdist}[H(x, x + t(\bar{x} - x)), \text{bd dom } g] \\
&\overset{(a)}{\geq} \text{sdist}[(1 - t)F(x) + tF(\bar{x}) + \frac{\boldsymbol{L}}{2}\|x - \bar{x}\|^2 t^2, \text{bd dom } g] \\
&\overset{(b)}{\geq} \text{sdist}[F(x), \text{bd dom } g](1 - t) + \text{sdist}[F(\bar{x}) + t\frac{\boldsymbol{L}}{2}\|x - \bar{x}\|^2, \text{bd dom } g]t \\
&\overset{(c)}{\geq} \text{sdist}[(1 - t)F(\bar{x}) + t(F(\bar{x}) + \frac{\boldsymbol{L}}{2}\|x - \bar{x}\|^2), \text{bd dom } g]t \\
&\overset{(d)}{\geq} \text{sdist}[F(\bar{x}), \text{bd dom } g]t(1 - t) + \text{sdist}[F(\bar{x}) + \frac{\boldsymbol{L}}{2}\|x - \bar{x}\|^2, \text{bd dom } g]t^2 \\
&=: \delta(t), \tag{53}
\end{aligned}$$

where for (a) we combine Eq. (52) with Lemma 5, for (b) we use the concavity of the signed distance function (see [19, p. 154]), for (c) we use the fact that $(1 - t)\text{sdist}[F(x), \text{bd dom } g] \geq 0$, and for (d) we use the concavity of the signed distance function again.

It is easy to verify that $\gamma(\bar{x}, x) \in (0, 1]$ is the maximizer of $\delta(t)$ over the interval $(0, 1]$. We now consider the following inequality.

$$\text{sdist}[F(\bar{x}) + \boldsymbol{L}\|x - \bar{x}\|^2/2, \text{bd dom } g] \geq -\frac{\|\boldsymbol{L}\|}{2}\|x - \bar{x}\|^2, \tag{54}$$

Inequality (54) holds true: indeed, either $F(\bar{x}) + L\|x - \bar{x}\|^2/2 \in \text{dom } g$ and the result is trivial or otherwise, the result holds by the definition of the distance as an infimum. If $\gamma(\bar{x}, x) = 1$, by its definition, one immediately has

$$\frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]}{2\{\text{sdist}[F(\bar{x}), \text{bd dom } g] - \text{sdist}[F(\bar{x}) + L\|x - \bar{x}\|^2/2, \text{bd dom } g]\}} \geq 1,$$

which implies

$$\text{sdist}[F(\bar{x}) + L\|x - \bar{x}\|^2/2, \text{bd dom } g] \geq \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]}{2}. \qquad (55)$$

Substituting $\gamma(\bar{x}, x) = 1$ into Eq. (53) yields

$$\begin{aligned}
\delta(\gamma(\bar{x}, x)) &= \text{sdist}[F(\bar{x}) + \frac{L}{2}\|x - \bar{x}\|^2, \text{bd dom } g] \\
&\geq \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]}{2} \\
&\geq \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]}{2} \times \frac{1}{2} \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]}{\text{sdist}[F(\bar{x}), \text{bd dom } g] + \|L\|\|x - \bar{x}\|^2/2} \\
&= \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]^2}{4\{\text{sdist}[F(\bar{x}), \text{bd dom } g] + \|L\|\|x - \bar{x}\|^2/2\}}, \qquad (56)
\end{aligned}$$

where the first inequality is obtained by considering Eq. (55). As a result, Eq. (51) holds true if $\gamma(\bar{x}, x) = 1$.

From now on, let us consider $\gamma(\bar{x}, x) < 1$. In this case, one has

$$\gamma(\bar{x}, x) = \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]}{2\{\text{sdist}[F(\bar{x}), \text{bd dom } g] - \text{sdist}[F(\bar{x}) + L\|x - \bar{x}\|^2/2, \text{bd dom } g]\}}.$$

Substituting into Eq. (53) and using (54) leads to

$$\begin{aligned}
\delta(\gamma(\bar{x}, x)) &= \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]^2}{4\{\text{sdist}[F(\bar{x}), \text{bd dom } g] - \text{sdist}[F(\bar{x}) + L\|x - \bar{x}\|^2/2, \text{bd dom } g]\}} \\
&\geq \frac{\text{sdist}[F(\bar{x}), \text{bd dom } g]^2}{4\{\text{sdist}[F(\bar{x}), \text{bd dom } g] + \|L\|\|x - \bar{x}\|^2/2\}}.
\end{aligned}$$

Eventually, combining this equation with (53) and (56) completes the proof.   □

We are now ready to describe the proof of Lemma 4

Proof of Lemma 4 (i) As the function g is L_g Lipschitz continuous on its domain, an immediate application of the Cauchy–Schwartz inequality leads to $L^T \nu \leq L_g \|L\|$ (see also Sect. 3.2.1).

(ii) The claim is trivial if bd dom$(g) = \emptyset$, hence we will assume that it is not so that we can use Lemmas 5 and 6. Set $w = x + \gamma(\bar{x}, x)(\bar{x} - x)$ with $\gamma(\bar{x}, x)$ given as in Lemma 6. By Lemma 6, one has $w \in \text{dom}(g \circ H(x, \cdot))$. Then, one obtains

$$\frac{\boldsymbol{L}^T v}{2} \|w - y\|^2 = [H(x, w) - H(x, y)]^T v$$
$$\leq g \circ H(x, w) - g \circ H(x, y)$$
$$\leq L_g \|H(x, w) - H(x, y)\|, \qquad (57)$$

where the equality follows from Eq. (11), the first inequality is obtained by the convexity of g, and the last inequality is due to the assumption that g is L_g Lipschitz continuous on its domain. On the other hand, a direct calculation yields

$$\|H(x, w) - H(x, y)\| = \|\nabla F(x)(w - y) + \frac{\boldsymbol{L}}{2}\|w - x\|^2 - \frac{\boldsymbol{L}}{2}\|y - x\|^2\|$$
$$= \|\nabla F(x)(w - y) + \frac{\boldsymbol{L}}{2}(\|w - y\|^2 + 2(w - y)^T(y - x))\|$$
$$\leq \|\nabla F(x)\|_{\text{op}}\|w - y\| + \frac{\|\boldsymbol{L}\|}{2}\|w - y\|^2 + \|\boldsymbol{L}\|\|w - y\|\|y - x\|$$
$$= \|w - y\|\left[\|\nabla F(x)\|_{\text{op}} + \frac{\|\boldsymbol{L}\|}{2}\|w - y\| + \|\boldsymbol{L}\|\|y - x\|\right]$$
$$\leq \|w - y\|\left[\|\nabla F(x)\|_{\text{op}} + \frac{\|\boldsymbol{L}\|}{2}\|x - y\|\right.$$
$$\left. + \gamma(\bar{x}, x)\frac{\|\boldsymbol{L}\|}{2}\|\bar{x} - x\| + \|\boldsymbol{L}\|\|y - x\|\right]$$
$$\leq \|w - y\|\left[\|\nabla F(x)\|_{\text{op}} + \frac{3\|\boldsymbol{L}\|}{2}\|x - y\| + \frac{\|\boldsymbol{L}\|}{2}\|\bar{x} - x\|\right]$$

Substituting this inequality into Eq. (57) leads to

$$\frac{\boldsymbol{L}^T v}{2}\|H(x, w) - H(x, y)\| \leq L_g\left[\|\nabla F(x)\|_{\text{op}} + \frac{3\|\boldsymbol{L}\|}{2}\|x - y\| + \frac{\|\boldsymbol{L}\|}{2}\|\bar{x} - x\|\right]^2.$$
$$(58)$$

As $H(x, y)$ is on the boundary of dom g, it follows that

$$\|H(x, w) - H(x, y)\| \geq \text{sdist}[H(x, w), \text{bd dom } g]. \qquad (59)$$

Combining this inequality with Lemma 6 eventually completes the proof. $\square$

References

1. Auslender, A., Teboulle, M.: Interior gradient and proximal methods for convex and conic optimization. SIAM J. Optim. **16**(3), 697–725 (2006)
2. Auslender, A., Shefi, R., Teboulle, M.: A moving balls approximation method for a class of smooth constrained minimization problems. SIAM J. Optim. **20**(6), 3232–3259 (2010)
3. Auslender, A.: An extended sequential quadratically constrained quadratic programming algorithm for nonlinear, semidefinite, and second-order cone programming. J. Optim. Theory Appl. **156**(2), 183–212 (2013)
4. Bauschke, H.H., Combettes, P.L.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer, New York (2017)
5. Beck A., A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**(1), 183–202 (2009)
6. Bolte, J., Pauwels, E.: Majorization-minimization procedures and convergence of SQP methods for semi-algebraic and tame programs. Math. Oper. Res. **41**(2), 442–465 (2016)
7. Burke, J.V.: Descent methods for composite nondifferentiable optimization problems. Math. Program. **33**(3), 260–279 (1985)
8. Burke, J.V., Ferris, M.C.: A Gauss–Newton method for convex composite optimization. Math. Program. **71**(2), 179–194 (1995)
9. Cartis, C., Gould, N.I., Toint, P.L.: On the evaluation complexity of composite function minimization with applications to nonconvex nonlinear programming. SIAM J. Optim. **21**(4), 1721–1739 (2011)
10. Cartis, C., Gould, N., Toint, P.: On the complexity of finding first-order critical points in constrained nonlinear optimization. Math. Program. **144**(1), 93–106 (2014)
11. Combettes, P.L., Wajs, V.R.: Signal recovery by proximal forward–backward splitting. Multiscale Model. Simul. **4**(4), 1168–2000 (2005)
12. Combettes, P.L., Pesquet, J.-C.: Proximal Splitting Methods in Signal Processing, Fixed-Point Algorithm for Inverse Problems in Science and Engineering. Optimization and Its Applications, pp. 185–212. Springer, New York (2011)
13. Combettes, P.L.: Systems of structured monotone inclusions: duality, algorithms, and applications. SIAM J. Optim. **23**(4), 2420–2447 (2013)
14. Combettes, P.L., Eckstein, J.: Asynchronous block-iterative primal-dual decomposition methods for monotone inclusions. Math. Program. **168**(1–2), 645–672 (2018). https://doi.org/10.1007/s10107-016-1044-0
15. Drusvyatskiy, D., Lewis, A.S.: Error bounds, quadratic growth, and linear convergence of proximal methods. Math. Op. Res. **43**(3), 919–948 (2018). https://doi.org/10.1287/moor.2017.0889
16. Drusvyatskiy, D., Paquette, C.: Efficiency of minimizing compositions of convex functions and smooth maps. Math. Program. (2018). https://doi.org/10.1007/s10107-018-1311-3
17. Eckstein, J.: Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming. Math. Oper. Res. **18**(1), 202–226 (1993)
18. Fletcher, R.: A model algorithm for composite nondifferentiable optimization problems. In: Sorensen, D.C., Wets, R.J.B. (eds.) Nondifferential and Variational Techniques in Optimization. Mathematical Programming Studies, vol. 17. Springer, Berlin, Heidelberg (1982)
19. Hiriart-Urruty, J.-B., Lemarechal, C.: Convex Analysis and Minimization Algorithm I. Springer, New York (1993)
20. Hiriart-Urruty, J.B.: A note on the Legendre–Fenchel transform of convex composite functions. In: Alart, P., Maisonneuve, O., Rockafellar R.T. (eds.) Nonsmooth Mechanics and Analysis, pp. 35–46. Springer US (2006)
21. Le Roux, N., Schmidt, M., Bach, F.: A stochastic gradient method with an exponential convergence rate for finite training sets. Adv. Neural Inf. Process. Syst. **25**, 2663–2671 (2012)
22. Levitin, E.S., Polyak, B.T.: Constrained minimization methods. USSR Comput. Math. Math. Phys. **6**(5), 1–50 (1966)
23. Lewis, A.S., Wright, S.J.: A proximal method for composite minimization. Mathe. Program. Math. Program. **158**(1–2), 501–546 (2016). https://doi.org/10.1007/s10107-015-0943-9
24. Li, C., Ng, K.F.: Majorizing functions and convergence of the Gauss–Newton method for convex composite optimization. SIAM J. Optim. **18**(2), 613–642 (2007)
25. Li, C., Wang, X.: On convergence of the Gauss–Newton method for convex composite optimization. Math. Program. **91**(2), 349–356 (2002)

26. Lions, P.-L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Anal. **16**(6), 964–979 (1979)
27. Lofberg, J.: YALMIP: A toolbox for modeling and optimization in MATLAB. In: IEEE International Symposium on Computer Aided Control Systems Design (2004)
28. Martinet, B.: Revue française d'informatique et de recherche opérationnelle, série rouge. Brève communication. Régularisation d'inéquations variationnelles par approximations successives **4**(3), 154–158 (1970)
29. Moreau, J.-J.: Proximité et dualité dans un espace hilbertien. Bulletin de la Société mathématique de France. **93**, 273–299 (1965)
30. Moreau, J.-J.: Evolution problem associated with a moving convex set in a Hilbert space. J. Differ. Equ. **26**(3), 347–374 (1977)
31. Mosek Aps: The MOSEK optimization toolbox for MATLAB manual. Version **7**, 1 (2016). https://www.yumpu.com/en/document/view/54768342/the-mosek-optimization-toolbox-for-matlab-manual-version-70-revision-141/32
32. Nesterov, Y., Nemirovskii, A.: Interior-Point Polynomial Algorithms in Convex Programming. Society for Industrial and Applied Mathematics, Philadelphia (1994)
33. Nesterov, Y.: Introductory Lectures on Convex Programming, Volumne I: Basis Course. Springer, New York (2004)
34. Nemirovskii, A., Yudin, D.: Problem Complexity and Method Efficiency in Optimization. Wiley, New York (1983)
35. Nocedal, J., Wright, S.: Numerical Optimization. Springer, New York (2006)
36. Ortega, J.M., Rheinboldt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. SIAM, Philadelphia (2000)
37. Passty, G.B.: Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. J. Math. Anal. Appl. **72**(2), 383–390 (1979)
38. Pauwels, E.: The value function approach to convergence analysis in composite optimization. Oper. Res. Lett. **44**(6), 790–795 (2016)
39. Pshenichnyi, B.N.: The linearization method. Optimization **18**(2), 179–196 (1987)
40. Rockafellar, R.T.: Augmented Lagrangians and applications of the proximal point algorithm in convex programming. Math. Oper. Res. **1**(2), 97–116 (1976)
41. Rockafellar, R.T., Wets, R.: Variational Analysis. Springer, New York (1998)
42. Rosen, J.B.: The gradient projection method for nonlinear programming. Part I. Linear constraints. J. Soc. Ind. Appl. Math. **8**(1), 181–217 (1960)
43. Rosen, J.B.: The gradient projection method for nonlinear programming. Part II. Nonlinear constraints. J. Soc. Ind. Appl. Math. **9**(4), 514–532 (1961)
44. Salzo, S., Villa, S.: Convergence analysis of a proximal Gauss–Newton method. Comput. Optim. Appl. **53**(2), 557–589 (2012)
45. Schmidt, M., Le Roux, N., Bach, F.: Minimizing finite sums with the stochastic average gradient. Math. Program. **162**(1–2), 83–112 (2017)
46. Shefi, R., Teboulle, M.: A dual method for minimizing a nonsmooth objective over one smooth inequality constraint. Math. Program. **159**(1–2), 137–164 (2016)
47. Solodov, : Global convergence of an SQP method without boundedness assumptions on any of the iterative sequences. Math. Program. **118**(1), 1–12 (2009)
48. Tseng, P.: Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. SIAM J. Control Optim. **29**(1), 119–138 (1991)
49. Villa, S., Salzo, S., Baldassarre, L., Verri, A.: Accelerated and inexact forward–backward algorithms. SIAM J. Optim. **23**(3), 1607–1633 (2013)
50. Ye, Y.: Interior Point Algorithms: Theory and Analysis. Yinyu Ye Wiley & Sons, New York (1997)