A concise and friendly introduction to complexity in continuous optimization

Ernesto G. Birgin

Abstract This paper aims to give a brief introduction to the concept of computational complexity in the context of continuous optimization.

1 Introduction

This paper is related to the plenary lecture titled "Complexity results in nonlinear optimization" given by Ernesto G. Birgin at the Brazil-Portugal Joint Meeting on Mathematics (*Encontro Conjunto Brasil-Portugal em Matemática*) that took place from August 14 to 20, 2022, at the Federal University of Bahia, Salvador, Brazil. In the same way as the presentation, this paper aims to introduce the concept of complexity in continuous optimization to a wide audience of non-specialists and to briefly review some recent results obtained by the author and collaborators.

Optimization is the area of Mathematics that studies the problem of finding $x \in \mathbb{R}^n$ that realizes the smallest possible value of a given objective function, among those points that belong to a predefined feasible region. If the objective function is a continuous function and the feasible region is determined by the points that satisfy a given set of equalities and inequalities defined by continuous functions, then we are dealing with a continuous optimization problem.

Given $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^p$, $h : \mathbb{R}^n \to \mathbb{R}^m$, and a set $\Omega \subseteq \mathbb{R}^n$, a standard continuous optimization problem can be written as follows

$$\underset{x \in \mathbb{R}^n}{\text{Minimize}}\ f(x) \text{ subject to } h(x) = 0, g(x) \leq 0, x \in \Omega. \tag{1}$$

If $\Omega = \mathbb{R}^n$ and g and h are absent, the problem is an unconstrained optimization problem. If g and h are absent and $\Omega = \{x \in \mathbb{R}^n \mid \ell \leq x \leq u\}$, where $\ell = (\ell_i)$,

Ernesto G. Birgin
Dept. of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, Rua do Matão, 1010, Cidade Universtária, São Paulo, SP, Brazil, 05508-090, e-mail: egbirgin@ime.usp.br

$u = (u_i)$, $\ell_i \in \mathbb{R} \cup \{-\infty\}$, and $u_i \in \mathbb{R} \cup \{+\infty\}$ for $i = 1, \ldots, n$, the problem is a bound-constrained minimization problem. If g is missing the problem is an equality-constrained optimization problem (with or without bound constraints), while if h is missing the problem is an inequality-constrained optimization problem.

Even in the simplest case of unconstrained minimization, a continuous optimization problem can rarely be solved analytically. Therefore, optimization methods are developed. In general, optimization methods are iterative and, given an initial approximation x^0, generate a sequence of iterates $\{x^k\}_{k=0}^{\infty}$ that, hopefully, as k increases, will better approximate a solution or satisfy some desirable condition.

Given an optimization method, there are basically two tasks necessary to show its value. First, it must be shown that it is well defined. That is, that given x^{k-1}, the steps that must be followed at iteration k to define x^k can be carried out. Secondly, the asymptotic properties of the sequence generated by the method must be studied. For example, in unconstrained minimization, a necessary condition for a point to be a local minimizer is that the gradient of the objective function vanishes at the point. Accordingly, a classical asymptotic result consists in showing that $\lim_{k\to+\infty} \nabla f(x^k) = 0$ or that $\lim_{k\in K} \nabla f(x^k) = 0$ for some $K \subset_\infty \mathbb{N}$.

Optimization problems are an object of study of Mathematics. However, besides that, they have an enormous number of practical applications in the most diverse areas of science such as Physics, Chemistry, Engineering, Economics, Medicine, and Social Sciences, among others. This practical appeal means that, in practice, optimization methods are equipped with a stopping criterion, since practitioners cannot wait infinite time. When equipped with a stopping criterion, the methods are transformed into algorithms (finite and well-defined sequence of operations). For the example of unconstrained minimization considered in the previous paragraph, given $\epsilon > 0$, a natural stopping criterion would be "to interrupt the execution of the method when finding x^k such that $\|\nabla f(x^k)\| \leq \epsilon$." This is the point where computational complexity comes in to answer the following question: In the worst case, what is the computational cost of finding x^k that satisfies the pre-specified stopping criterion of a given optimization algorithm? It is worth noting that this "computational cost" can be measured in terms of number of arithmetic operations, number of iterations of the algorithm or number of evaluations of the objective function f, among other possibilities. In the last two cases, we are talking about iteration complexity and evaluation complexity, respectively.

The convergence theory of optimization methods consisted basically in the asymptotic study of the infinite sequences generated by the methods until the beginning of the current millennium. It was only in 2006 that a paper by Nesterov and Polyak [21] introduced the idea of computational complexity in continuous optimization. The idea caught the interest of the academic optimization community and gained great prominence in the last sixteen years. In 2022, the first book [18] specifically dedicated to the subject was released.

The rest of this paper is organized as follows. In Section 2, we present as simply as possible an algorithm for unconstrained minimization and its computational complexity analysis. In Section 3, we deal with the computational complexity of an augmented Lagrangian algorithm applied to the solution of the most general contin-

uous optimization problem. The final section presents some perspectives of current and future work.

Notation. The symbol $\|\cdot\|$ denotes the Euclidean norm of vectors and matrices. For $v \in \mathbb{R}^n$, $v_+ = (\max\{0, v_1\}, \ldots, \max\{0, v_n\})^T$. If $K = \{k_1, k_2, \ldots\} \subseteq \mathbb{N}$ (with $k_j < k_{j+1}$ for all j), we denote $K \subset_\infty \mathbb{N}$. Given $\phi : \mathbb{R}^n \to \mathbb{R}^m$, $\phi = (\phi_1, \ldots, \phi_m)^T$, we denote $\nabla\phi(x) = (\nabla\phi_1(x), \ldots, \nabla\phi_m(x))$. $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \mid x \geq 0\}$. Given two functions $a : \mathbb{R} \to \mathbb{R}$ and $b : \mathbb{R} \to \mathbb{R}$, we say $a(\epsilon) = O(b(\epsilon))$ as $\epsilon \to 0$ if there exist positive numbers δ and M such that $|a(\epsilon)| \leq Mb(x)$ for all $0 < \epsilon < \delta$.

2 A didactic example of computational complexity for unconstrained minimization

In this section we consider an unconstrained minimization problem given by

$$\operatorname*{Minimize}_{x\in\mathbb{R}^n} f(x), \tag{2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$. To tackle problem (2), we introduce a method based on "cubic regularization" that, given an initial guess $x^0 \in \mathbb{R}^n$, generates a sequence $\{x^k\}_{k=0}^\infty$. We show that, given a tolerance $\epsilon > 0$, the method uses a finite number of iterations of order $\epsilon^{-3/2}$ to find a point x^k such that $\|\nabla f(x^k)\| \leq \epsilon$. In fact, we show more than that. We show that the (finite) number of iterations such that $\|\nabla f(x^k)\| > \epsilon$ is of the order of $\epsilon^{-3/2}$. The results are different because from the former one it would follow that $\lim_{k\in K} \nabla f(x^k) = 0$ for some $K \subset_\infty \mathbb{N}$, while from the latter one it follows that $\lim_{k\to+\infty} \nabla f(x^k) = 0$. The complexity results on the number of iterations also follow for the number of evaluations of f, because in the introduced method the number of evaluations of f per iterations is $O(1)$ with respect to ϵ.

The method follows below.

Method 2.1. Let $\alpha > 0$ and $x^0 \in \mathbb{R}^n$ be given. Set $k \leftarrow 1$.

Step 1. Define $x^k = x^{k-1} + s^k$, where s^k is such that

$$f(x^k) \leq f(x^{k-1}) - \alpha\|s^k\|^3. \tag{3}$$

Step 2. Set $k \leftarrow k + 1$ and go to Step 1.

At this point, Method 2.1 is not very elucidative, because we did not say yet in what way an s^k satisfying (3) can be computed. (In fact we did not even mention under what conditions an s^k satisfying (3) exists, so it is not clear yet under what hypotheses the method is well defined.) However, it is interesting to understand what properties Method 2.1 has under suitable assumptions.

Let us assume that there exists $\gamma > 0$ such that

$$\sqrt{\frac{\|\nabla f(x^k)\|}{\gamma}} \leq \|s^k\| \tag{4}$$

for all $k \in \mathbb{N}$. (Later in this section we will give sufficient conditions for this assumption to be satisfied.) From (3) and (4), it trivially follows that, for all $k \in \mathbb{N}$,

$$f(x^k) \leq f(x^{k-1}) - c\|\nabla f(x^k)\|^{3/2} \tag{5}$$

with $c = \alpha/\gamma^{3/2}$.

Assume now that, given a tolerance $\epsilon > 0$, we desire to stop Method 2.1 the first time an iterate x^k satisfying $\|\nabla f(x^k)\| \leq \epsilon$ is computed. Assume the method is at (the end of) iteration k_{current} and it did not stop. It means that $\|\nabla f(x^k)\| > \epsilon$ for $k = 1, \ldots, k_{\text{current}}$. Thus, by (5), $f(x^k) \leq f(x^{k-1}) - c\epsilon^{3/2}$ for $k = 1, \ldots, k_{\text{current}}$, i.e. the value of the objective function decreased at least $c\epsilon^{3/2}$ at every iteration already executed. At this point there are two alternatives. Or the method is in the way to compute a sequence $\{x^k\}_{k=0}^{\infty}$ such that $\lim_{k\in K} f(x^k) = -\infty$ for some $K \subset_\infty \mathbb{N}$ (in which case f is unbounded below) or there exists $f_{\text{low}} \in \mathbb{R}$ such that $f(x^k) \geq f_{\text{low}}$ for all $k \in \mathbb{N}$ and

$$k_{\text{current}} \leq k_\epsilon := \left\lfloor \frac{f(x^0) - f_{\text{low}}}{c\epsilon^{3/2}} \right\rfloor = O(\epsilon^{-3/2}).$$

We name the assumption "there exists $f_{\text{low}} \in \mathbb{R}$ such that $f(x^k) \geq f_{\text{low}}$ for all $k \in \mathbb{N}$" **Assumption A1** from now on. This assumption is an assumption in the sequence generated by the method, which is undesired. On the other hand, it holds trivially if there exists f_{low} such that $f(x) \geq f_{\text{low}}$ for all $x \in \mathbb{R}^n$.

Up to this point, we have shown that, under assumption (4) and Assumption A1, given $\epsilon > 0$, there exists $k \leq k_\epsilon + 1$ such that $\|\nabla f(x^k)\| \leq \epsilon$ and that the number of iterations such that $\|\nabla f(x^k)\| > \epsilon$ is limited by k_ϵ. From the former it follows $\lim_{k\in K} \nabla f(x^k) = 0$ for some $K \subset_\infty \mathbb{N}$, while from the latter it follows $\lim_{k\to+\infty} \nabla f(x^k) = 0$. It remains to show how to compute s^k at each iteration k using $O(1)$ functional evaluations. By showing this we will also address the satisfaction of (4) and we will reveal from where the qualifier "cubic regularized" for Method 2.1 comes from.

There are several alternatives to compute, at iteration k, a step s^k satisfying the sufficient descent condition (3) plus (4). A simple choice will be present here for didactical purposes. Consider the the cubic regularized second-order Taylor polynomial of f around x^{k-1} given by

$$M_k(s, \sigma) := T_k(s) + \frac{\sigma}{3}\|s\|^3,$$

where

$$T_k(s) := f(x^{k-1}) + \nabla f(x^{k-1})^T s + \frac{1}{2} s^T \nabla^2 f(x^{k-1}) s$$

and $\sigma > 0$ plays the role of a regularization parameter. The procedure to compute s^k is described by the method below.

Method 2.2. Let $\sigma_{k,1} \geq \sigma_{\text{low}} > 0$ and $\theta > 0$ be given (with σ_{low} and θ being the same for all k.) Let $\ell \leftarrow 1$.

Step 1. Compute $s^{k,\ell}$ such that

$$M_k(s^{k,\ell}, \sigma_{k,\ell}) \leq M_k(0, \sigma_{k,\ell}) \tag{6}$$

and

$$\|\nabla_s M_k(s^{k,\ell}, \sigma_{k,\ell})\| \leq \theta \|s^{k,\ell}\|^2. \tag{7}$$

Step 2. If (3) does not hold with $s^k \equiv s^{k,\ell}$ then define $\sigma_{k,\ell+1} := 2\sigma_{k,\ell}$, set $\ell \leftarrow \ell+1$ and go to Step 1.

Step 3. Define $\sigma_k := \sigma_{k,\ell}$ and $s^k := s^{k,\ell}$.

The first task is to show that Method 2.2 is well defined and that it is in fact an algorithm, i.e. that it stops in a finite number of iterations (having computed a step s^k that satisfies (3) plus (6,7) as desired.) The fact that the method is well defined comes from the fact that $s^{k,\ell}$ satisfying (6,7) can be computed for all k and ℓ. This is because the models $M_k(s, \sigma_{k,\ell})$ with $\sigma_{k,\ell} > 0$ have bounded level sets and, thus, have at least one global minimizer. At the global minimizer, the functional value is upper bounded by $M_k(0, \sigma_{k,\ell})$ and the gradient vanishes. Therefore, (6,7) hold. Moreover, the step $s^{k,\ell}$ can be computed in finite time by any monotone unconstrained minimization method that possesses worst-case complexity starting from $s \equiv 0$. It is worth noting that this task does not depend on ϵ and does not require evaluations of the objective function f.

The task of showing that Method 2.2 stops in a finite number of iterations requires to assume that there exist non-negative constants ξ_1 and ξ_2 such that the second-order Taylor polynomials T_k satisfy

$$f(x^{k-1} + s^{k,\ell}) - T_k(s^{k,\ell}) \leq \xi_1 \|s^{k,\ell}\|^3 \tag{8}$$

and

$$\|\nabla f(x^{k-1} + s^{k,\ell}) - \nabla_s T_k(s^{k,\ell})\| \leq \xi_2 \|s^{k,\ell}\|^2 \tag{9}$$

for all k and ℓ. We name it **Assumption A2**. This assumption is fulfilled when f is three times continuously differentiable on $\mathbb{R}^n$ and the third-order derivative of f is bounded or when f is twice continuously differentiable and the second-order derivative is Lipschitz continuous; see, for example, [4]. With these assumptions, in particular using (8), (6), and the fact that $M_k(0, \cdot) = f(x^{k-1})$, we have that, if

$$\sigma_{k,\ell} \geq 3(\xi_1 + \alpha), \tag{10}$$

then

$$\begin{aligned} f(x^{k-1}+s^{k,\ell}) &\leq T_k(s^{k,\ell}) + \xi_1\|s^{k,\ell}\|^3 \\ &= T_k(s^{k,\ell}) + \tfrac{\sigma_{k,\ell}}{3}\|s^{k,\ell}\|^3 - (\tfrac{\sigma_{k,\ell}}{3} - \xi_1)\|s^{k,\ell}\|^3 \\ &= M_k(s^{k,\ell},\sigma_{k,\ell}) - (\tfrac{\sigma_{k,\ell}}{3} - \xi_1)\|s^{k,\ell}\|^3 \\ &\leq M_k(0,\sigma_{k,\ell}) - \alpha\|s^{k,\ell}\|^3 \\ &= f(x^{k-1}) - \alpha\|s^{k,\ell}\|^3, \end{aligned}$$

i.e. that $s^{k,\ell}$ satisfies (3). Since $\sigma_{k,1} \geq \sigma_{\text{low}}$ and $\sigma_{k,\ell} = 2\sigma_{k,\ell-1}$ for $\ell = 2, 3, \ldots$ then it is clear that Method 2.2 achieves a sufficiently large value for the regularization parameter in at most $\lceil \log_2((3(\xi_1+\alpha))/\sigma_{\text{low}}) + 1 \rceil$ iterations, a quantity that does not depend on ϵ. At each iteration ℓ, Method 2.2 needs to check if the computed trial step $s^{k,\ell}$ satisfies (3). This verification involves an evaluation of f.

Now observe that, by the increasing rule of the regularization parameter in Method 2.2 (i.e. doubling it) and the fact that $\sigma_{k,\ell} \geq 3(\xi_1 + \alpha)$ makes $s^{k,\ell}$ to satisfy the sufficient descent condition (3), it holds that

$$\sigma_k < \sigma_{\max} := 6(\xi_1 + \alpha) \tag{11}$$

for all k. We now aim to show that (11), (9), and (7) imply (4). Let us write

$$\|\nabla f(x^k)\| = \|\nabla f(x^{k-1}+s^k)\| = \|\nabla f(x^{k-1}+s^k) - \nabla_s M_k(s^k,\sigma_k) + \nabla_s M_k(s^k,\sigma_k)\|. \tag{12}$$

Now, since

$$\nabla_s M_k(s,\sigma) = \nabla_s T_k(s) + \sigma\|s\|^2 \frac{s}{\|s\|}, \tag{13}$$

by substituting (13) in (12), applying the triangle inequality, and then using (9,11,7), we have

$$\begin{aligned} \|\nabla f(x^k)\| &= \|\nabla f(x^{k-1}+s^k) - \nabla_s T_k(s^k) - \sigma_k\|s^k\|^2\tfrac{s^k}{\|s^k\|} + \nabla_s M_k(s^k,\sigma_k)\| \\ &\leq \|\nabla f(x^{k-1}+s^k) - \nabla_s T_k(s^k)\| + \|\sigma_k\|s^k\|^2\| + \|\nabla_s M_k(s^k,\sigma_k)\| \\ &\leq (\xi_2 + \sigma_{\max} + \theta)\|s^k\|^2. \end{aligned}$$

So (4) holds with $\gamma = \xi_2 + \sigma_{\max} + \theta$.

Summarizing, we showed in this section an algorithm that, given $f : \mathbb{R}^n \to \mathbb{R}$ and $\epsilon > 0$, finds a point x^k that satisfies $\|f(x^k)\| \leq \epsilon$ in a finite number of iterations of order $\epsilon^{-3/2}$. For that, Assumptions A1 and A2 are required, and sufficient conditions for their satisfaction are that (i) f is bounded from below by f_{low} in $\mathbb{R}^n$ and that (ii)a f is three times continuously differentiable on $\mathbb{R}^n$ and the third-order derivative of f is bounded or (ii)b f is twice continuously differentiable and the second-order derivative is Lipschitz continuous.

It is important to mention that the derivations presented in this section are nowadays standard in the literature and can be found, among many others, in, for example, [1, 5, 6, 7, 8, 10, 11, 12, 14, 15, 16, 17, 21] applied in proofs of complexity results of algorithms for unconstrained minimization, minimization with bound-constraints or convex sets, and nonlinear programming in general. Moreover, the presented results can also be extended, as in fact they already were, for arbitrary norms, models of

order p (other than Taylor polynomials) with regularization of order $p+1$ and to find stationary points of order q instead of first-order stationary points only.

3 The case of augmented Lagrangians for nonlinear optimization

In this section we consider the constrained problem given by

$$\underset{x\in\mathbb{R}^n}{\text{Minimize}}\ f(x) \text{ subject to } h(x)=0, g(x)\leq 0, \ell\leq x\leq u \tag{14}$$

where $f:\mathbb{R}^n\to\mathbb{R}$, $g:\mathbb{R}^n\to\mathbb{R}^p$ and $h:\mathbb{R}^n\to\mathbb{R}^m$ are continuously differentiable and $\ell, u\in\mathbb{R}^n$.

Augmented Lagrangians represent a well-established family of methods for solving nonlinear programming problems of the form (14). The computational complexity of different augmented Lagrangian methods was studied in [13, 19, 24]. In [13], the complexity of Algencan was analyzed. Algencan [2, 3, 9] is an implementation of a safeguarded augmented Lagrangian method relying on the Powell-Hestenes-Rockafellar augmented Lagrangian function [22, 20, 23]. Each iteration of Algencan consists of (a) minimization of the objective function plus a term that penalizes violation of the constraints with respect to appropriate shifted tolerances and (b) updating of the Lagrange multipliers approximations.

The Powell-Hestenes-Rockafellar augmented Lagrangian function is given by

$$L_\rho(x,\lambda,\mu) := f(x)+\frac{\rho}{2}\left[\sum_{i=1}^m\left(h_i(x)+\frac{\lambda_i}{\rho}\right)^2+\sum_{i=1}^p\left(g_i(x)+\frac{\mu_i}{\rho}\right)_+^2\right]$$

for all $x\in[\ell,u]$, $\rho>0$, $\lambda\in\mathbb{R}^m$, and $\mu\in\mathbb{R}^p_+$, where ρ is the penalty parameter and λ and μ represent the Lagrange multipliers associated with the equality constraints $h(x)=0$ and the inequality constraints $g(x)\leq 0$, respectively.

Method 3.1 below describes Algencan.

Method 3.1: Assume that $x^0\in\mathbb{R}^n$, $\lambda_{\min}<\lambda_{\max}$, $\bar\lambda^1\in[\lambda_{\min},\lambda_{\max}]^m$, $\mu_{\max}>0$, $\bar\mu^1\in[0,\mu_{\max}]^p$, $\rho_1>0$, $\gamma>1$, $0<\tau<1$, and $\{\epsilon_k\}_{k=1}^\infty$ are given. Initialize $k\leftarrow 1$.

Step 1. Compute $x^k\in[\ell,u]$ satisfying

$$\left\|P_{[\ell,u]}\left(x^k-\nabla L_{\rho_k}(x^k,\bar\lambda^k,\bar\mu^k)\right)-x^k\right\|_\infty\leq\epsilon_k \tag{15}$$

by approximately solving

$$\underset{x\in\mathbb{R}^n}{\text{Minimize}}\ L_{\rho_k}(x,\bar\lambda^k,\bar\mu^k) \text{ subject to } x\in[\ell,u].$$

Step 2. Define

$$V^k = \min\left\{-g(x^k), \frac{\bar{\mu}^k}{\rho_k}\right\}.$$

If $k = 1$ or

$$\max\left\{\|h(x^k)\|_\infty, \|V^k\|_\infty\right\} \leq \tau \max\left\{\|h(x^{k-1})\|_\infty, \|V^{k-1}\|_\infty\right\},$$

choose $\rho_{k+1} \geq \rho_k$. Otherwise, choose $\rho_{k+1} \geq \gamma\rho_k$.

Step 3. Compute

$$\lambda^{k+1} = \bar{\lambda}^k + \rho_k h(x^k) \text{ and } \mu^{k+1} = \left(\bar{\mu}^k + \rho_k g(x^k)\right)_+.$$

Compute $\bar{\lambda}^{k+1} \in [\lambda_{\min}, \lambda_{\max}]^m$ and $\bar{\mu}_i^{k+1} \in [0, \mu_{\max}]^p$. Set $k \leftarrow k + 1$ and go to Step 1.

Algencan was introduced in [2, 3] and it is fully described in the book [9]. In particular, [9] describes the asymptotic convergence theory of Algencan. On the other hand, [13] complements Algencan's theory presented in [9] by presenting its computational complexity theory. In this section we summarize the Algencan computational complexity theory presented in [13].

Method 3.1 has interesting properties for the case of sequences of arbitrary tolerances $\{\epsilon_k\}_{k=1}^\infty$, which are used as tolerance for the approximate solution of the subproblems in Step 1; see [9] for details. However, in the present work, we are interested in the case $\epsilon_k \to 0$. For that reason, from here on, we will analyze this case only. When Method 3.1 is applied to problem (14), the method generates sequences of primal iterands $\{x^k\}$ and sequences of Lagrange multipliers $\{\lambda^k\}$ and $\{\mu^k\}$. If $\lim_{k\in K} x^k = x^*$ for some $K \subset_\infty \mathbb{N}$, then there are two possibilities for x^*. Either x^* is *infeasible* and satisfies the first-order optimality conditions for the problem of minimizing the squared infeasibility given by

$$\underset{x\in[\ell,u]}{\text{Minimize}} \|h(x)\|^2 + \|g(x)_+\|^2$$

or x^* is *feasible* and satisfies the sequential optimality condition AKKT of problem (14) given by

$$\lim_{k\in K} \left\| P_{[\ell,u]}\left(x^k - \left(\nabla f(x^k) + \nabla h(x^k)\lambda^{k+1} + \nabla g(x^k)\mu^{k+1}\right)\right) - x^k \right\| = 0$$

and

$$\lim_{k\in K} \max\left\{\|h(x^k)\|, \|\min\{-g(x^k), \mu^{k+1}\}\|\right\} = 0.$$

As in this work we are interested in complexity results associated with Method 3.1, our objective is to transform Method 3.1 into an algorithm incorporating stopping criteria associated with the two possibilities mentioned in the previous paragraph and to quantify the computational cost that the resulting algorithm consumes to stop due to either of the two criteria.

Stating the complexity results requires a couple of lemmas that we present below. Proofs of the lemmas can be found at [13]. We record here that these lemmas involve hypotheses on problem (14), namely, the bondedness of $[\ell, u]$ and the continuity of f, g h, and their gradients. The main complexity results of Method 3.1 are stated in the sequel. Additional results can be found in [13].

Lemma 3.1 *[13, Lem. 3.2] There exists $c_{\text{big}} > 0$ such that, for all $k \geq 1$,*

$$\max\{\|h(x^k)\|_\infty, \|V_k\|_\infty\} \leq c_{\text{big}}.$$

Lemma 3.2 *[13, Lem. 3.3] There exist $c_{\text{lips}} > 0$ and $c_f > 0$ such that, for all $x \in [\ell, u]$, $\lambda \in [\lambda_{\min}, \lambda_{\max}]^m$, and $\mu \in [0, \mu_{\max}]^p$, one has*

$$\|\nabla h(x)\|\|\lambda\| + \|\nabla g(x)\|\|\mu\| \leq c_{\text{lips}}$$

and

$$\|\nabla f(x)\| \leq c_f.$$

Theorem 3.3 *[13, Thm. 3.5] Let $\delta > 0$, $\delta_{\text{low}} \in (0, \delta)$, and $\epsilon > 0$ be given. Let $N(\delta_{\text{low}}, \epsilon)$ be such that $\epsilon_k \leq \min\{\epsilon, \delta_{\text{low}}\}/4$ for all $k \geq N(\delta_{\text{low}}, \epsilon)$. Then, after at most*

$$\max\left\{N(\delta_{\text{low}}, \epsilon), \left\lceil \frac{\log(\delta/c_{\text{big}})}{\log(\tau)} \right\rceil \times \left\lceil \frac{\log(\rho_{\max}/\rho_1)}{\log(\gamma)} \right\rceil\right\}$$

iterations, where

$$\rho_{\max} = \max\left\{1, \frac{4c_{\text{lips}}}{\delta_{\text{low}}}, \frac{\mu_{\max}}{\delta}, \frac{4c_f}{\delta_{\text{low}}}\right\}, \tag{16}$$

we obtain an iteration k such that one of the following two facts takes place:

1. The iterate $x^k \in [\ell, u]$ verifies

$$\left\|P_{[\ell,u]}\left(x^k - \nabla\left[\|h(x^k)\|^2 + \|g(x^k)_+\|^2\right]\right) - x^k\right\|_\infty \leq \delta_{\text{low}} \tag{17}$$

and

$$\max\{\|h(x^k)\|_\infty, \|g(x^k)_+\|_\infty\} > \delta. \tag{18}$$

2. The multipliers $\lambda^{k+1} \in \mathbb{R}^m$ and $\mu^{k+1} \in \mathbb{R}^p_+$ are such that

$$\left\|P_{[\ell,u]}\left(x^k - \left(\nabla f(x^k) + \nabla h(x^k)\lambda^{k+1} + \nabla g(x^k)\mu^{k+1}\right)\right) - x^k\right\|_\infty \leq \epsilon, \tag{19}$$

$$\|h(x^k)\|_\infty \leq \delta, \ \|g(x^k)_+\|_\infty \leq \delta, \tag{20}$$

and, for all $j = 1, \ldots, p$,

$$\mu_j^{k+1} = 0 \text{ whenever } g_j(x^k) < -\delta. \tag{21}$$

Theorem 3.4 *[13, Thm. 3.6] In addition to the hypotheses of Theorem 3.3, assume that there exist $\bar{c}_{\text{inner}} > 0$, $\nu > 0$, and $q > 0$, where $\bar{c}_{\text{inner}}$ only depends on $\lambda_{\min}$,*

$\lambda_{\max}$, $\mu_{\max}$, ℓ, u, and characteristics of the functions f, h, and g, such that the number of inner iterations, function and derivative evaluations that are necessary to obtain (15) is bounded above by $\bar{c}_{\text{inner}}\,\rho_k^v\epsilon_k^{-q}$. Then, the number of inner iterations, function evaluations, and derivative evaluations that are necessary to obtain k such that (17) and (18) hold or (19), (20) and (21) hold is bounded above by

$$\bar{c}_{\text{inner}}\,\rho_{\max}^v\epsilon_{\min,3}^{-q}\max\left\{N(\delta_{\text{low}},\epsilon),\left\lceil\frac{\log(\delta/c_{\text{big}})}{\log(\tau)}\right\rceil\times\left\lceil\frac{\log\left(\rho_{\max}/\rho_1\right)}{\log(\gamma)}\right\rceil\right\},$$

where $\rho_{\max}$ is given by (16) and

$$\epsilon_{\min,3}=\min\left\{\epsilon_k\mid k\leq\max\left\{N(\delta_{\text{low}},\epsilon),\left\lceil\frac{\log(\delta/c_{\text{big}})}{\log(\tau)}\right\rceil\times\left\lceil\frac{\log\left(\rho_{\max}/\rho_1\right)}{\log(\gamma)}\right\rceil\right\}\right\}.$$

Note that, in Theorem 3.4, it is assumed that the number of inner iterations, function and derivative evaluations that are necessary to obtain (15) is bounded above by $\bar{c}_{\text{inner}}\,\rho_k^v\epsilon_k^{-q}$. Therefore, due to (16), c_{inner} depends on the tolerances δ and δ_{low}. The complexity theory of the algorithm that Algencan uses to solve the subproblems, and which satisfies the hypotheses of Theorem 3.4, is presented in [13].

It is worth keeping in mind that the hypotheses used on problem (14) to obtain the complexity results mentioned above are minimal and they do not include any constraint qualification. Consider an algorithm that, given a problem of the form (14) and a tolerance $\epsilon > 0$, checks whether $(x,\lambda,\mu) = 0$ satisfies first-order optimality conditions with precision ϵ. If it satisfies, it returns $(x,\lambda,\mu) = 0$. Otherwise, it declares that it failed. With the hypothesis "$(x,\lambda,\mu) = 0$ is a first-order stationary point of the problem", we would prove that the method finds a solution to problem (14) with computational complexity $O(1)$. It would definitely be the most efficient method in the world, but it would be of little use.

4 Conclusions and perspectives for future work

In this text, we introduced the concept of computational complexity in the area of continuous optimization. We illustrated the idea with the simplest possible case of unconstrained minimization and concluded by addressing the more general case of an augmented Lagrangian algorithm for nonlinear programming. In between, many problems and algorithms can and have been considered in the literature. A careful look at the references in this work or a search on the author's web page, his Google Scholar profile or the references of the book [18] would be a good starting point for the interested reader.

One line of research widely used in the literature consists of, for a given problem, trying to develop the algorithm with the best possible complexity. Two important points must be considered when this point of view is used. The first is related to the hypotheses that the algorithm (with potentially low complexity) needs to find points with the desired properties. In other words, doesn't the search for a competitive

complexity leave out many problems that do not satisfy the necessary hypotheses? (See [8, Table 2].) The second question concerns the applicability of the proposed method. Often, an algorithm with low complexity is known to be useless in practice, because it suffers from known issues, such as generating too short steps far from a solution.

The effective contribution of complexity analysis to the development of novel continuous optimization algorithms that are clean, easy to understand and implement, and have significantly better performance than existing, well-established ones remains to be verified. For the reasons mentioned in the previous paragraph, this author believes that the genuine and already verified contribution of complexity analysis of continuous optimization algorithms lies in deepening or closing a gap in the study of existing consolidated algorithms. In-depth knowledge of existing algorithms helps to use them in the best possible way, to better interpret their behavior and, potentially, to incorporate improvements.

References

1. Amaral, V.A., Andreani, R., Birgin, E.G., Marcondes, D.S., Martínez, J.M.: On complexity and convergence of high-order coordinate descent algorithms for smooth nonconvex box-constrained minimization. Journal of Global Optimization **84**, 527–561 (2022).
2. Andreani, R., Birgin, E.G., Martínez, J.M., Schuverdt, M.L.: Augmented Lagrangian methods under the Constant Positive Linear Dependence constraint qualification. Mathematical Programming **111**, 5–32 (2008).
3. Andreani, R., Birgin, E.G., Martínez, J.M., Schuverdt, M.L.: On Augmented Lagrangian methods with general lower-level constraints. SIAM Journal on Optimization **18**, 1286–1309 (2008).
4. Bertsekas, D.P.: Nonlinear Programming. 2ed. Athena Scientific, Belmont, MA (1999).
5. Birgin, E.G., Bueno, L.F., Martínez, J.M.: On the complexity of solving feasibility problems with quadratically regularized models. Optimization Methods and Software **37**, 405–424 (2022).
6. Birgin, E.G., Gardenghi, J. L., Martínez, J.M., Santos, S.A.: On the use of third-order models with fourth-order regularization for unconstrained optimization. Optimization Letters **14**, 815–838 (2020).
7. Birgin, E.G., Gardenghi, J. L., Martínez, J.M., Santos, S.A., Toint, Ph.L.: Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. Mathematical Programming **163**, 359–368 (2017).
8. Birgin, E.G., Gardenghi, J. L., Martínez, J.M., Santos, S.A., Toint, Ph.L.: Evaluation complexity for nonlinear constrained optimization using unscaled KKT conditions and high-order models. SIAM Journal on Optimization **26**, 951–967 (2016).
9. Birgin, E.G., Martínez, J.M.: Practical Augmented Lagrangian Methods for Constrained Optimization. Society for Industrial and Applied Mathematics, Philadelphia, PA (2014).
10. Birgin, E.G., Martínez, J.M.: The use of quadratic regularization with a cubic descent condition for unconstrained optimization. SIAM Journal on Optimization **27**, 1049–1074 (2017).
11. Birgin, E.G., Martínez, J.M.: On regularization and active-set methods with complexity for constrained optimization. SIAM Journal on Optimization **28**, 1367–1395 (2018).
12. Birgin, E.G., Martínez, J.M.: A Newton-like method with mixed factorizations and cubic regularization for unconstrained minimization. Computational Optimization and Applications **73**, 707–753 (2019).

13. Birgin, E.G., Martínez, J.M.: Complexity and performance of an Augmented Lagrangian algorithm. Optimization Methods and Software **35**, pp. 885-920 (2020).
14. Birgin, E.G., Martínez, J.M.: Block Coordinate Descent for smooth nonconvex constrained minimization. Computational Optimization and Applications **83**, 1–27 (2022).
15. Cartis, C., Gould, N.I.M., Toint, Ph.L.: On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization. SIAM Journal on Optimization, **20**, 2833–2852, (2010).
16. Cartis, C., Gould, N.I.M., Toint, Ph.L.: Adaptive cubic overestimation methods for un- constrained optimization. Part II: worst-case function-evaluation complexity. Mathematical Programming, Series A, **130**, 295-319 (2011).
17. Cartis, C., Gould, N.I.M., Toint, Ph.L.: An adaptive cubic regularization algorithm for nonconvex optimization with convex constraints and its function-evaluation complexity. IMA Journal of Numerical Analysis, **32**, 1662–1645 (2012).
18. Cartis, C., Gould, N.I.M., Toint, Ph.L.: Evaluation Complexity of Algorithms for Nonconvex Optimization: Theory, Computation, and Perspectives. SIAM, Philadelphia, PA (2022).
19. Grapiglia, G.N., Yuan, Y.: On the complexity of an augmented Lagrangian method for nonconvex optimization. IMA Journal of Numerical Analysis, **41**, 1546–1568 (2021).
20. Hestenes, M.R.: Multiplier and gradient methods. Journal of Optimization Theory and Applications **4**, 303–320 (1969).
21. Nesterov, Y., Polyak, B.T.: Cubic regularization of Newton method and its global performance. Mathematical Programming **108**, 177–205 (2006).
22. Powell, M.J.D.: A method for nonlinear constraints in minimization problems, in Optimization, R. Fletcher (ed.), Academic Press, New York, NY, 283–298 (1969).
23. Rockafellar, R.T.: Augmented Lagrange multiplier functions and duality in nonconvex programming. SIAM Journal on Control and Optimization **12**, 268–285 (1974).
24. Xie, Y., Wright, S.J.: Complexity of Proximal Augmented Lagrangian for Nonconvex Optimization with Nonlinear Equality Constraints. Journal of Scientific Computing **86**, 38 (2021).