On the complexity of solving feasibility problems with regularized models*

E. G. Birgin[†] L. F. Bueno [‡] J. M. Martínez[§]

June 10, 2019[¶]

Abstract

The complexity of solving feasibility problems is considered in this work. It is assumed that the constraints that define the problem can be divided into two sets, namely, expensive and cheap constraints. It is also assumed that the set of solutions of the cheap constraints is non-empty and bounded and that minimizing a pth-order model of the sum of squares of the expensive constraints subject to the cheap constraints is an affordable task. At each iteration, the introduced method minimizes a regularized pth-approximation of the sum of squares of the expensive constraints subject to the cheap constraints. Under a Hölder continuity property with constant $\beta \in (0,1]$ on the pth derivatives of the expensive constraints, it is shown that finding a feasible point with precision $\varepsilon > 0$ or an infeasible point that is stationary with tolerance $\gamma > 0$ of minimizing the Euclidean norm of the expensive constraints residual subject to the cheap constraints has iteration complexity $O\left(|\log(\varepsilon)|\,\gamma^{\zeta_p^\beta}\,\omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right)$ and evaluation complexity (of the expensive constraints) $O\left(|\log(\varepsilon)|\left[\gamma^{\zeta_p^\beta}\,\omega_p^{1+\frac{1}{2}\zeta_p^\beta} + \frac{1-\beta}{p+\beta-1}|\log(\gamma\sqrt{\varepsilon})|\right]\right)$, where $\zeta_p^\beta = -(p+\beta)/(p+\beta-1)$ and $\omega_p = \varepsilon$ if $p = 1$, while $\omega_p = \Phi(x^0)$ if $p > 1$. When the pth derivatives of the expensive constraints satisfy a Lipschitz condition, both complexities reduce to $O\left(|\log(\varepsilon)|\,\gamma^{-\frac{p+1}{p}}\right)$. Still under the Hölder continuity property on the pth derivatives of the expensive constraints, and under a stronger regularity assumption with constant κ, that avoids KKT points of minimizing the sum of squares of the expensive constraints subject to the cheap constraints of being infeasible, the iteration complexity is shown to be $O\left(|\log(\varepsilon)|\,\omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right)$; while the evaluation complexity is given by $O\left(|\log(\varepsilon)|\left[\omega_p^{1+\frac{1}{2}\zeta_p^\beta} + \frac{1-\beta}{p+\beta-1}\,|\log(\sqrt{\varepsilon})|\right]\right)$. When the pth derivatives of the expensive constraints satisfy a Lipschitz condition, both complexities reduce

*This work was supported by FAPESP (grants 2013/07375-0, 2016/01860-1, and 2018/24293-0) and CNPq (grants 309517/2014-1 and 303750/2014-6).

[†] Dept. of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, Rua do Matão, 1010, Cidade Universitária, 05508-090, São Paulo, SP, Brazil. email: egbirgin@ime.usp.br

[‡] Dept. of Science and Technology, Federal University of São Paulo, São José dos Campos, SP, Brazil. email: lfelipebueno@gmail.com

[§] Dept. of Applied Mathematics, Institute of Mathematics, Statistics, and Scientific Computing, State University of Campinas, 13083-859, Campinas, SP, Brazil. email: martinez@ime.unicamp.br

[¶] Revision made on November 6, 2019.

to $O\left(|\log(\varepsilon)|\right)$, independently of p. If the Lipschitz condition is *not* satisfied but $p > 1$, the iteration complexity reduces to $O\left(|\log(\varepsilon)|\right)$; while the evaluation complexity reduces to $O\left(|\log(\varepsilon)|^2\right)$. The most costly case is the case $p = 1$ and $\beta < 1$ in which both complexities reduce to $O(|\log(\varepsilon)|\,\varepsilon^{\frac{\beta-1}{2\beta}})$. The results introduced in the present work generalize the results obtained for p even in [C. Cartis, N. I. M. Gould, and Ph. L. Toint, Improved worst-case evaluation complexity for potentially rank-deficient nonlinear least-Euclidean-norm problems using higher-order regularized models, Technical Report NA 15-17, University of Oxford, 2015].

Key words: Complexity, feasibility problem, high order methods.

Mathematics Subject Classification (2010): 90C30, 65K05, 49M37, 90C60, 68Q25.

1 Introduction

Many practical problems require the solution of systems of equalities and inequalities. Socio-economic models usually aim targets with respect to education, housing, eradication of poverty, sustainability, and public health. (See, for example, [18].) Molecular dynamics is a powerful technique for comprehension at the molecular level of a great variety of chemical processes. In molecular dynamics, simulations need starting points with adequate energy requirements. If the starting configuration has close atoms, the temperature scaling is disrupted by excessive potentials that accelerate molecules over the accepted velocities for almost any reasonable integration time step. In Packmol [22], the problem of finding an adequate initial configuration for a molecular dynamics simulation is tackled as a feasibility problem. The goal is to place molecules within an arbitrary finite domain in such a way that distances between every pair of atoms belonging to different molecules are larger than a given threshold tolerance. Matrix completion (see, for example, [19, 20]) is another problem that can be classified as a feasibility problem and that includes several practical applications. The problem consists of reconstructing a matrix possessing certain properties knowing only a subset of its entries. Properties may include positive semidefinite matrices, Euclidean distance matrices, contraction matrices, matrices of a given rank, correlation matrices, doubly stochastic matrices, and Hadamard matrices, among others. (See, [2] for details.)

In this work, it is assumed that the constraints that define the problem can be divided into expensive and cheap constraints. It is also assumed that the set of solutions of the cheap constraints is non-empty and bounded, that it is relatively easy to maintain feasibility with respect to the cheap constraints, and that minimizing a regularized model of the sum of squares of the expensive constraints subject to the cheap constraints is affordable. In contrast, it is assumed that expensive constraints are expensive to evaluate. Thus, at each iteration, the proposed method computes a new iterate as an approximate minimizer of a regularized model of the sum of squares of the expensive constraints subject to the cheap constraints. Examples of cheap constraints are bounds on the variables, linear constraints, spherical and ball constraints, intersections of balls and polytopes, and matrices with some property such as idempotency or semidefiniteness, among others.

Feasibility problems that come from linear parameter estimation in the case that prior information on the parameters is represented by the level set of a nonconvex function was considered in [4]. The problem of minimizing a sum of squares subject to convex constraints was considered in [11, 12]. In [11], a cubic regularization scheme of ARC-type [9, 10] was employed and complexity results were given, considering stopping criteria based on the residual norm and on the gradient of the residual norm. In [12], the approach of [11] was extended in order to consider arbitrary pth order Taylor approximations of the objective function in the sense of [8]. The case in which p is odd (including $p = 1$) was not addressed in [12]. In the present paper $p \geq 1$ is arbitrary and only Hölder conditions are assumed to be satisfied by pth derivatives of the expensive constraints as in [13, 15, 16, 21]. In fact, results presented in the present work are a direct consequence of recent developments on the minimization of functions with Hölder continuity assumptions introduced in [21]. In the present work, it is shown that finding a feasible point with precision $\varepsilon > 0$ or an infeasible point that is stationary of minimizing the Euclidean norm of the expensive constraints residual subject to the cheap constraints with tolerance $\gamma > 0$ has iteration complexity

$$O\left(|\log(\varepsilon)|\, \gamma^{\zeta_p^\beta}\, \omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right)$$

and evaluation complexity (of the expensive constraints and their derivatives)

$$O\left(|\log(\varepsilon)| \left[\gamma^{\zeta_p^\beta}\, \omega_p^{1+\frac{1}{2}\zeta_p^\beta} + \frac{1-\beta}{p+\beta-1}\, |\log(\gamma\sqrt{\varepsilon})|\right]\right),$$

where $\beta \in (0, 1]$ is the constant in the Hölder continuity property of the pth derivatives of the expensive constraints, $\zeta_p^\beta = -\frac{p+\beta}{p+\beta-1}$ and

$$\omega_p = \begin{cases} \varepsilon, & \text{if } p = 1, \\ \Phi(x^0), & \text{if } p > 1. \end{cases}$$

When the pth derivatives of the expensive constraints satisfy a Lipschitz condition (so, $\beta = 1$), both complexities reduce to $O\left(|\log(\varepsilon)|\, \gamma^{-(p+1)/p}\right)$. Still under the Hölder continuity property on the pth derivatives of the expensive constraints, and under a gradient-domination property (see [23]) with constant κ, which guarantees that KKT points of the sum of squares of the expensive constraints subject to the cheap constraints are feasible, the iteration complexity is shown to be

$$O\left(|\log(\varepsilon)|\, \omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right);$$

while the evaluation complexity is given by

$$O\left(|\log(\varepsilon)| \left[\omega_p^{1+\frac{1}{2}\zeta_p^\beta} + \frac{1-\beta}{p+\beta-1}\, |\log(\sqrt{\varepsilon})|\right]\right).$$

When the pth derivatives of the expensive constraints satisfy a Lipschitz condition (so, $\beta = 1$), both complexities reduce to $O\left(|\log(\varepsilon)|\right)$, independently of p. When the pth derivatives of the expensive constraints do *not* satisfy a Lipschitz condition (so, $\beta < 1$), the iteration complexity reduces to $O\left(|\log(\varepsilon)|\right)$ and the evaluation complexity reduces to $O\left(|\log(\varepsilon)|^2\right)$ if $p > 1$. The most costly case is the case $\beta < 1$ and $p = 1$, in which both complexities reduce to $O(|\log(\varepsilon)|\, \varepsilon^{\frac{\beta-1}{2\beta}})$.

Similar bounds for the ARC-like method, with convex cheap constraints and using $(p+1)$-regularized pth order Taylor models with p even, have been presented in [12, Thm.3.6].

The rest of this work is organized as follows. Section 2 introduces the proposed algorithms. Section 3 presents the complexity results. At the end of the section, it is presented an example showing that the bound $|\log(\varepsilon)|$ is sharp; and, thus, it provides a reliable estimation of the computer work in practical cases. Illustrative numerical examples are given in Section 4. Final remarks are given in Section 5.

Notation. $\|\cdot\|$ denotes the Euclidean norm. If $v \in \mathbb{R}^n$ is a vector with components v_i, v_+ is the vector with components $\max\{0, v_i\}$, $i = 1, \dots, n$. If $\psi(x)$ is a vectorial function, we denote its Jacobian by $\psi'(x) = \left(\frac{\partial \psi_i}{\partial x_j}(x)\right)$. $\mathbb{R}_+$ denotes the set of nonnegative elements of $\mathbb{R}$. If v and w are vectors with components v_i and w_i, respectively, $\min\{v, w\}$ denotes the vector with components $\min\{v_i, w_i\}$. If $\mathcal{C}$ is a set, we denote its diameter by $\text{diam}(\mathcal{C})$.

2 Proposed algorithm

The problem tackled in this work consists in finding $x \in \mathbb{R}^n$ such that

$$h(x) = 0, \ g(x) \leq 0, \tag{1}$$

$$\underline{h}(x) = 0, \text{ and } \underline{g}(x) \leq 0, \tag{2}$$

where $h : \mathbb{R}^n \to \mathbb{R}^m$, $g : \mathbb{R}^n \to \mathbb{R}^q$, $\underline{h} : \mathbb{R}^n \to \mathbb{R}^{\underline{m}}$, and $\underline{g} : \mathbb{R}^n \to \mathbb{R}^{\underline{q}}$ have continuous first derivatives for all $x \in \mathbb{R}^n$. Constrains (2) are considered to be "cheap;" while constraints (1) are considered "expensive" constraints. It is assumed that the set of points $x \in \mathbb{R}^n$ satisfying (2) is non-empty and bounded. For all $x \in \mathbb{R}^n$, we define

$$\Phi(x) = \frac{1}{2}(\|h(x)\|^2 + \|g(x)_+\|^2). \tag{3}$$

Thus, problem (1,2) can be reformulated as

$$\text{Minimize } \Phi(x) \text{ subject to } \underline{h}(x) = 0 \text{ and } \underline{g}(x) \leq 0. \tag{4}$$

The connection between the feasibility problem (1,2) and its reformulation (4) as an optimization problem is that a solution to (4) at which the objective function vanishes is also a solution to (1,2). Unfortunately, under standard assumptions, stationary points of (4) that do not satisfy the expensive constraints (1) are also a possible outcome of a method concerned with the resolution of (4), even in the case in which the feasibility problem (1,2) has a solution. Certificates of infeasibility could only be obtained using additional assumptions on the constraints or using global optimization algorithms.

The method considered in this paper for solving (4) is iterative. Each iteration k rests upon the resolution of a sequence of subproblems of the form

$$\text{Minimize } M_{x^{k,\ell}}(x) + \sigma_{k,\ell}\|x - x^{k,\ell}\|^{p+1} \text{ subject to } \underline{h}(x) = 0 \text{ and } \underline{g}(x) \leq 0 \tag{5}$$

for $\ell = 0, 1, 2, \ldots$; where $\sigma_{k,\ell} > 0$ and $M_{x^{k,\ell}}(x)$ is a pth-order model of $\Phi(x)$ at $x^{k,\ell}$ (for some integer $p \geq 1$), in a sense that will be defined later. The motivation for this approach relies on the characterization of constraints (1,2) as cheap and expensive constraints. Constraints (2) are said to be cheap in the sense that it is assumed that obtaining a feasible point which is an approximate solution to (5) is affordable. Constraints (1) are said to be expensive in the sense that it is assumed that evaluating them is much more expensive than evaluating the cheap constraints. The description of the algorithm follows.

Algorithm 2.1. Assume that $\varepsilon > 0$, $\gamma > 0$, $p \in \{1, 2, \ldots\}$, $\alpha \in (0, 1)$, $\sigma_{\min} > 0$, $\theta > 0$, and $x^0 \in \mathbb{R}^n$ such that $\underline{h}(x^0) = 0$, $\underline{g}(x^0) \leq 0$, and $\Phi(x^0) > \varepsilon$ are given. Initialize $k \leftarrow 0$.

Step 1. Compute $x^{k+1} \in \mathbb{R}^n$, $\lambda^{k+1} \in \mathbb{R}^{\underline{m}}$, and $\mu^{k+1} \in \mathbb{R}^{\underline{q}}_{+}$ satisfying

$$\underline{h}(x^{k+1}) = 0,\ \underline{g}(x^{k+1}) \leq 0 \text{ and } \min\{\mu^{k+1}, -\underline{g}(x^{k+1})\} = 0 \tag{6}$$

and such that either

$$\Phi(x^{k+1}) \leq \frac{1}{2}\Phi(x^k) \tag{7}$$

or

$$\left\| \nabla \Phi(x^{k+1}) + \underline{h}'(x^{k+1})^T \lambda^{k+1} + \underline{g}'(x^{k+1})^T \mu^{k+1} \right\| \leq \gamma \sqrt{\Phi(x^k)}. \tag{8}$$

Step 2. If (7) holds and $\Phi(x^{k+1}) > \varepsilon$, update $k \leftarrow k + 1$, and go to Step 1. Otherwise, stop.

We now analyze the situation in which, at iteration k, Algorithm 2.1 stops. This means that $x^{k+1} \in \mathbb{R}^n$, $\lambda^{k+1} \in \mathbb{R}^{\underline{m}}$, and $\mu^{k+1} \in \mathbb{R}^{\underline{q}}_{+}$ are such that (6) holds and either $\Phi(x^{k+1}) \leq \varepsilon$ or (7) does not hold. In the first case, x^{k+1} is the feasible point with precision ε we were looking for. In the latter case, by definition of the algorithm, if (7) does not hold, we must have that (8) holds. Dividing both sides of (8) by $\sqrt{\Phi(x^{k+1})}$, we obtain

$$\left\| \frac{\nabla \Phi(x^{k+1})}{\sqrt{\Phi(x^{k+1})}} + \underline{h}'(x^{k+1})^T \left(\frac{\lambda^{k+1}}{\sqrt{\Phi(x^{k+1})}} \right) + \underline{g}'(x^{k+1})^T \left(\frac{\mu^{k+1}}{\sqrt{\Phi(x^{k+1})}} \right) \right\| \leq \gamma \frac{\sqrt{\Phi(x^k)}}{\sqrt{\Phi(x^{k+1})}}. \tag{9}$$

But, since (7) does not hold, one has that $\sqrt{\Phi(x^k)}/\sqrt{\Phi(x^{k+1})} \leq \sqrt{2}$. Moreover,

$$\nabla \left[\sqrt{\Phi(x^{k+1})} \right] = \frac{1}{2} \frac{\nabla \Phi(x^{k+1})}{\sqrt{\Phi(x^{k+1})}}.$$

Therefore, by (9),

$$\left\| 2\nabla \left[\sqrt{\Phi(x^{k+1})} \right] + \underline{h}'(x^{k+1})^T \left(\frac{\lambda^{k+1}}{\sqrt{\Phi(x^{k+1})}} \right) + \underline{g}'(x^{k+1})^T \left(\frac{\mu^{k+1}}{\sqrt{\Phi(x^{k+1})}} \right) \right\| \leq \gamma \sqrt{2}. \tag{10}$$

Thus,

$$\left\| \nabla \sqrt{\Phi(x^{k+1})} + \underline{h}'(x^{k+1})^T \left(\frac{\lambda^{k+1}}{2\sqrt{\Phi(x^{k+1})}} \right) + \underline{g}'(x^{k+1})^T \left(\frac{\mu^{k+1}}{2\sqrt{\Phi(x^{k+1})}} \right) \right\| \leq \gamma \frac{\sqrt{2}}{2} = \frac{\gamma}{\sqrt{2}}. \tag{11}$$

By (6), (11) means that x^{k+1}, with multipliers $\lambda^{k+1}/(2\sqrt{\Phi(x^{k+1})})$ and $\mu^{k+1}/(2\sqrt{\Phi(x^{k+1})})$, is a KKT point with tolerance γ for the minimization of $\sqrt{\Phi(x)}$ subject to $\underline{h}(x) = 0$ and $\underline{g}(x) \le 0$. If γ is chosen to be much smaller than ε, this stopping criterion may be a symptom of the fact that x^{k+1} is an approximate infeasible local minimizer of infeasibility. This is not a guarantee that the original feasibility problem is infeasible. Certificates of infeasibility could only be obtained using additional properties on the constraints or using global optimization algorithms. Unfortunately, all practical algorithms for constrained optimization stop when infeasible points that are almost stationary with respect to infeasibility are found.

Algorithm 2.2 below is used at Step 1 of Algorithm 2.1 to compute x^{k+1}, λ^{k+1}, and μ^{k+1}. The description of Algorithm 2.2 below is an instantiation of [21, Alg.1] applied to the minimization of $\Phi(x)$ subject to $\underline{h}(x) = 0$ and $\underline{g}(x) \le 0$, using x^k as initial guess, $\frac{1}{2}\Phi(x^k)$ as a target for the functional value, and $\min\{0.99, \gamma\sqrt{\Phi(x^k)}\}$ as the tolerance for the norm of the gradient of the Lagrangian. Note that, in (8), a precision of $\gamma\sqrt{\Phi(x^k)}$ is required. The $\min\{0.99, \cdot\}$ appears because this quantity plays the role of a tolerance that, in [21, Alg.1], is assumed to be in $(0, 1)$. In this sense, the constant 0.99 could be replaced with any other value in $(0, 1)$. The constant $\frac{1}{2}$ in (7) could also be replaced with any other value in $(0, 1)$. It is implicit that, every time Algorithm 2.2 is used, its parameters γ, p, α, $\sigma_{\min}$, and θ correspond to those of Algorithm 2.1, i.e. the same at every call. This is why some of those values, that are not explicitly used in Algorithm 2.1, appear in its list of parameters.

Algorithm 2.2. Assume that $\gamma > 0$, $p \in \{1, 2, \dots\}$, $\alpha \in (0, 1)$, $\sigma_{\min} > 0$, $\theta > 0$ are given. Initialize $x^{k,0} = x^k$, $\sigma_{k,0} = \sigma_{\min}$, and $\ell \leftarrow 0$.

Step 1. Choose a pth-order model $M_{x^{k,\ell}}(x)$ for $\Phi(x)$ at $x^{k,\ell}$.

Step 2. Find $x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{\underline{m}}$, and $\mu \in \mathbb{R}^{\underline{q}}_+$ such that

$$M_{x^{k,\ell}}(x) + \sigma_{k,\ell}\|x - x^{k,\ell}\|^{p+1} \le M_{x^{k,\ell}}(x^{k,\ell}), \tag{12}$$

$$\left\|\nabla(M_{x^{k,\ell}}(x) + \|x - x^{k,\ell}\|^{p+1}) + \underline{h}'(x)^T\lambda + \underline{g}'(x)^T\mu\right\| \le \theta\|x - x^{k,\ell}\|^p, \tag{13}$$

$$\underline{h}(x) = 0,\ \underline{g}(x) \le 0,\ \text{and } \|\min\{\mu, -\underline{g}(x)\}\| = 0. \tag{14}$$

Step 3. If

$$\Phi(x) \le \frac{1}{2}\Phi(x^k) \tag{15}$$

or

$$\|\nabla\Phi(x) + \underline{h}'(x)^T\lambda + \underline{g}'(x)^T\mu\| \le \min\left\{0.99, \gamma\sqrt{\Phi(x^k)}\right\}, \tag{16}$$

stop returning x, λ, and μ.

Step 4. Test the sufficient descent condition

$$\Phi(x) \le \Phi(x^{k,\ell}) - \frac{\alpha}{(2p+4)^{\frac{p+1}{p}}} \frac{\min\left\{0.99, \gamma\sqrt{\Phi(x^k)}\right\}^{\frac{p+1}{p}}}{\sigma_{k,\ell}}. \tag{17}$$

If (17) does not hold, redefine $\sigma_{k,\ell} \leftarrow 2\sigma_{k,\ell}$ and go to Step 2. Otherwise, define $x^{k,\ell+1} = x$ and $\sigma_{k,\ell+1} = \sigma_{k,\ell}$, update $\ell \leftarrow \ell + 1$, and go to Step 1.

Remark. In the case $p = 1$, model $M_{x^{k,\ell}}$ may take the form

$$M_{x^{k,\ell}}(x) = \Phi(x^{k,\ell}) + \nabla\Phi(x^{k,\ell})^T(x - x^{k,\ell}) + \frac{1}{2}(x - x^{k,\ell})^T B_{k,\ell}(x - x^{k,\ell}),$$

with arbitrary choices of the matrix $B_{k,\ell}$. Even the null matrix may be chosen. This opens the possibility of using problem-oriented safeguarded quasi-Newton approximations of $\nabla^2\Phi(x)$. The Gauss-Newton approximation

$$\sum_{i=1}^{m} \nabla h_i(x)\nabla h_i(x)^T + \sum_{g_i(x)\geq 0} \nabla g_i(x)\nabla g_i(x)^T,$$

that only involves first-order information of the derivatives of h and g, is an interesting alternative due to its positive semidefiniteness and the fact that it approximates $\nabla^2\Phi(x)$ when x is almost feasible. Other possibilities in the quasi-Newton field are the rank-one correction SR1 and structured quasi-Newton approximations [14].

An attentive reader familiar with [21, Alg.1] may have noticed that parameter $\delta > 0$ of [21, Alg.1] is missing in the description of Algorithm 2.2. Parameter δ in [21, Alg.1] is a tolerance for the satisfaction of the constraints and the complementarity. In the current work, we are assuming that constraints $\underline{h}$ and $\underline{g}$ are cheap. This is why, at Step 2 of Algorithm 2.2, it is possible to require (14) instead of

$$\|\underline{h}(x)\| \leq \delta,\ \|\underline{g}(x)_+\| \leq \delta,\ \text{and } \|\min\{\mu, -\underline{g}(x)\}\| \leq \delta. \tag{18}$$

Since, naturally, (14) implies (18) for any $\delta > 0$, properties of [21, Alg.1] are preserved in Algorithm 2.2.

3 Complexity results

By the continuity of the derivatives of $h(x)$ and $g(x)$, function $\Phi(x)$ also has continuous first derivatives. However, in general, second derivatives of $\Phi(x)$ do not exist. This fact could restrict the applicability of the proposed method to the case $p = 1$. However, modifying the constraints $g_i(x) \leq 0$ by means of the introduction of slack variables, all expensive constraints become equalities and function $\Phi(x)$ inherits all the differentiability properties of the functions that define the expensive constraints.

Assumption A1 *Let C_0 be an open, convex, and bounded set that contains all solutions to (2). There exist $\beta \in (0, 1]$ and $L > 0$ such that for all $\bar{x}, x \in C_0$,*

$$\|\nabla\Phi(x) - \nabla M_{\bar{x}}(x)\| \leq L\|x - \bar{x}\|^{p+\beta-1},$$

$$M_{\bar{x}}(\bar{x}) = \Phi(\bar{x}),\ \textit{and}\ \Phi(x) \leq M_{\bar{x}}(x) + L\|x - \bar{x}\|^{p+\beta}.$$

Assumption A1 corresponds to assumptions (2) and (3) in [21, p.2448]. Note that constants L and β in Assumption A1 may depend on the diameter of C_0. Assumption A1 holds assuming Hölder continuity of the pth-derivatives of Φ if the model $M_{\bar{x}}$ is built as the pth Taylor polynomial around $\bar{x}$ plus an homogeneous polynomial of order $p+1$. See, for example, [13], [21], and [25, Lem.1].

Assumption A2 below ensures that Step 2 of Algorithm 2.2 is well defined.

Assumption A2 *For all $\theta > 0$, $\sigma \geq 0$, and $\bar{x} \in \mathbb{R}^n$ satisfying $\underline{h}(\bar{x}) = 0$ and $\underline{g}(\bar{x}) \leq 0$, there exist $x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{\underline{m}}$, and $\mu \in \mathbb{R}^{\underline{q}}_+$ such that*

$$M_{\bar{x}}(x) + \sigma\|x - \bar{x}\|^{p+1} \leq M_{\bar{x}}(\bar{x}),$$

$$\left\|\nabla(M_{\bar{x}}(x) + \sigma\|x - \bar{x}\|^{p+1}) + \underline{h}'(x)^T\lambda + \underline{g}'(x)^T\mu\right\| \leq \theta\|x - \bar{x}\|^p,$$

$$\|\underline{h}(x)\| = 0,\ \|\underline{g}(x)_+\| = 0,\ \text{and } \|\min\{\mu, -\underline{g}(x)\}\| = 0.$$

Step 2 of Algorithm 2.2 relies on the approximate minimization of the subproblem given by

$$\text{Minimize } M_{\bar{x}}(x) + \bar{\sigma}\|x - \bar{x}\|^{p+1} \text{ subject to } \underline{h}(x) = 0 \text{ and } \underline{g}(x) \leq 0,$$

with $\bar{x} = x^{k,\ell}$ and $\bar{\sigma} = \sigma_{k,\ell}$ for some k and some ℓ. Since $\bar{x}$ is a feasible point at which the subproblem's objective function value is $M_{\bar{x}}(\bar{x})$, the first condition in Assumption A2 must be satisfied at every minimizer of the subproblem. If the cheap constraints (that are the constraints of the subproblem) satisfy a constraint qualification then the KKT conditions hold at every minimizer of the subproblem; and, thus, every minimizer also satisfies the second and the third conditions in Assumption A2. Therefore, under the assumption that the cheap constraints satisfy a constraint qualification, the minimizers of the subproblem satisfy Assumption A2. Since it is assumed that the set of points that satisfy the cheap constraints is non-empty and bounded, the subproblem has at least a minimizer; and, thus, under the assumption that the cheap constraints satisfy a constraint qualification, at least a point that satisfies Assumption A2 exists.

The following theorem, that is a particular case of [21, Thm.2.3], limits the number of iterations of Algorithm 2.2 when it is called at iteration k of Algorithm 2.1.

Theorem 3.1 *Suppose that Assumptions A1 and A2 hold. Then, there exists $c_p > 0$, only dependent on α, β, θ, L, and p such that, when Algorithm 2.2 is called at iteration k of Algorithm 2.1, after at most*

$$\frac{\Phi(x^k)}{2}\frac{\left(\min\left\{0.99, \gamma\sqrt{\Phi(x^k)}\right\}\right)^{-\frac{p+\beta}{p+\beta-1}}}{\alpha c_p} \tag{19}$$

iterations, Algorithm 2.2 computes $x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{\underline{m}}$, and $\mu \in \mathbb{R}^{\underline{q}}_+$ verifying

$$\|\underline{h}(x)\| = 0,\ \|\underline{g}(x)_+\| \leq 0,\ \text{and } \|\min\{\mu, -\underline{g}(x)\}\| = 0 \tag{20}$$

that also satisfies either

$$\Phi(x) \leq \frac{1}{2}\Phi(x^k) \tag{21}$$

or

$$\left\|\nabla\Phi(x) + \underline{h}'(x)^T\lambda + \underline{g}'(x)^T\mu\right\| \leq \min\left\{0.99, \gamma\sqrt{\Phi(x^k)}\right\}. \tag{22}$$

Proof: The proof follows from [21, Thm.2.3]. The expression of c_p is given in [21, (26)]. □

The following theorem, that is a particular case of [21, Thm.2.4], provides a bound for the number of evaluations of Φ and its derivatives made by Algorithm 2.2 when it is called at iteration k of Algorithm 2.1.

Theorem 3.2 *Suppose that Assumptions A1 and A2 hold. Then, the number of evaluations of Φ and its derivatives made when Algorithm 2.2 is called at iteration k of Algorithm 2.1 is bounded above by*

$$\frac{\Phi(x^k)}{2} \frac{\left(\min\left\{0.99, \gamma\sqrt{\Phi(x^k)}\right\}\right)^{-\frac{p+\beta}{p+\beta-1}}}{\alpha c_p} + \frac{1-\beta}{p+\beta-1}\left|\log_2(\gamma\sqrt{\varepsilon})\right| + c_a, \tag{23}$$

where c_p and c_a only depends on α, β, θ, L, p, and $\sigma_{\min}$.

Proof: By Theorem 3.1, when Algorithm 2.2 is called at iteration k of Algorithm 2.1, a maximum of

$$\frac{\Phi(x^k)}{2} \frac{\left(\min\left\{0.99, \gamma\sqrt{\Phi(x^k)}\right\}\right)^{-\frac{p+\beta}{p+\beta-1}}}{\alpha c_p} \tag{24}$$

iterations is performed; and, by [21, Thm.2.4], the maximum number of function and derivatives evaluations of Φ is given by (24) plus

$$\max\left\{\log(\theta), \left(\frac{1-\beta}{p+\beta-1}\log\left(\left(\min\left\{0.99, \gamma\sqrt{\Phi(x^k)}\right\}\right)^{-1}\right) + c_e\right)\right\} + |\log_2(\sigma_{\min})| + 1, \tag{25}$$

where c_p and c_e only depends on α, β, θ, L, and p. (In fact, as already mentioned in the proof of Theorem 3.1, constant c_p is given by [21, (26)]; while c_e corresponds to c_ℓ in the statement of [21, Thm.2.4].) If we define

$$c_a = c_e + |\log(\theta)| + \left(\frac{1-\beta}{p+\beta-1}\right)\log(0.99^{-1}) + |\log(\sigma_{\min})| + 1 \tag{26}$$

then (23) follows from straightforward calculations using that, before termination of Algorithm 2.1, one has that $\Phi(x^k) > \varepsilon$. □

The following theorem limits the total number of iterations of Algorithm 2.2, as well as the total number of evaluations of Φ and its derivatives, during the whole execution of Algorithm 2.1.

Theorem 3.3 *Suppose that Assumptions A1 and A2 hold. Then, during the execution of Algorithm 2.1, the total number of iterations of Algorithm 2.2 is bounded above by*

$$\frac{1}{2\alpha c_p}\left|\log_2\left(\varepsilon/\Phi(x^0)\right)\right| \max\left\{0.99^{\zeta_p^\beta}\,\Phi(x^0), \gamma^{\zeta_p^\beta}\,\omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right\} \tag{27}$$

and the total number of evaluations of Φ *and its derivatives performed by Algorithm 2.2 is bounded above by*

$$\frac{1}{2\alpha c_p}\left|\log_2\left(\varepsilon/\Phi(x^0)\right)\right|\left(\max\left\{0.99^{\zeta_p^\beta}\,\Phi(x^0),\gamma^{\zeta_p^\beta}\,\omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right\}+\frac{1-\beta}{p+\beta-1}\left|\log_2\left(\gamma\sqrt{\varepsilon}\right)\right|+c_a\right),\quad (28)$$

where

$$\omega_p=\begin{cases}\varepsilon, & \textit{if } p=1,\\ \Phi(x^0), & \textit{if } p>1,\end{cases}$$

$\zeta_p^\beta=-\frac{p+\beta}{p+\beta-1}$*, and* c_p*, given by [21, (26)], and* c_a*, defined in (26), depend only on* α*,* β*,* θ*,* L*,* p*, and* $\sigma_{\min}$*.*

Proof: Theorem 3.1 gives the upper bound (19) for the number of iterations performed by Algorithm 2.2 when it is called at iteration k of Algorithm 2.1. Note that

$$\frac{\Phi(x^k)}{2}\frac{\left(\min\left\{0.99,\gamma\sqrt{\Phi(x^k)}\right\}\right)^{\zeta_p^\beta}}{\alpha c_p}=\frac{1}{2\alpha c_p}\max\left\{0.99^{\zeta_p^\beta}\,\Phi(x^k),\gamma^{\zeta_p^\beta}\,\Phi(x^k)^{1+\frac{1}{2}\zeta_p^\beta}\right\}.\quad (29)$$

If $p=1$ then $p+\beta\le 2$ and, therefore, $1+\frac{1}{2}\zeta_p^\beta=\frac{p+\beta-2}{2(p+\beta-1)}\le 0$. So, using that $\Phi(x^k)>\varepsilon$, we have that

$$\Phi(x^k)^{1+\frac{1}{2}\zeta_p^\beta}\le\varepsilon^{1+\frac{1}{2}\zeta_p^\beta}.\quad (30)$$

On the other hand, if $p>1$ then $p+\beta\ge 2$ and, therefore, $1+\frac{1}{2}\zeta_p^\beta=\frac{p+\beta-2}{2(p+\beta-1)}\ge 0$. So, since $\Phi(x^k)\le\Phi(x^0)$, we have that

$$\Phi(x^k)^{1+\frac{1}{2}\zeta_p^\beta}\le\Phi(x^0)^{1+\frac{1}{2}\zeta_p^\beta}.\quad (31)$$

Thus, (27) follows from (29), $\Phi(x^k)\le\Phi(x^0)$, (30), (31), and the fact that the number of times Algorithm 2.1 calls Algorithm 2.2 is obviously bounded by $|\log_2(\varepsilon/\Phi(x^0))|$. Bound (28) follows by using the same arguments but starting from the upper bound (23), given by Theorem 3.2, on the number of evaluations of Φ and its derivatives performed by Algorithm 2.2 when it is called at iteration k of Algorithm 2.1. □

Let us examine the result of Theorem 3.3 under the light of the new assumption below.

Assumption A3 *There exists* $\kappa>0$ *such that, for all* $x\in\mathbb{R}^n$*,* $\lambda\in\mathbb{R}^{\underline{m}}$*, and* $\mu\in\mathbb{R}_+^{\underline{q}}$ *satisfying* $\underline{h}(x)=0$*,* $\underline{g}(x)\le 0$*,* $\min\{\mu,-\underline{g}(x)\}=0$*, and* $\Phi(x)>0$*, we have that*

$$\left\|\frac{\nabla\Phi(x)}{\sqrt{\Phi(x)}}+\underline{h}'(x)^T\lambda+\underline{g}'(x)^T\mu\right\|\ge\kappa.\quad (32)$$

Consider the problem

$$\text{Minimize }\sqrt{\Phi(x)}\text{ subject to }\underline{h}(x)=0\text{ and }\underline{g}(x)\le 0.\quad (33)$$

Since $\nabla\left[\sqrt{\Phi(x)}\right] = \frac{1}{2}\frac{\nabla\Phi(x)}{\sqrt{\Phi(x)}}$, Assumption A3 means that, if $(x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^{\underline{m}} \times \mathbb{R}^{\underline{q}}_+$ is such that x is a feasible point of problem (33) at which $\Phi(x)$ does not vanish and μ satisfies the complementarity conditions of problem (33) then the gradient of the Lagrangian of problem (33) evaluated at (x, λ, μ) is bounded away from zero. Thus, KKT points of (33) must satisfy $\Phi(x) = 0$; and this means that they must be solutions to the original feasibility problem (1,2). If problem (1,2) has no cheap constraints then (32) represents a uniform regularity assumption of the constraints (1). Moreover, if problem (1,2) has no cheap constraints and it has a solution then (32) coincides with the gradient-domination property of degree 2 and constant $\tau_\Phi = \kappa^2$ as defined in [23, p.191].

Lemma 3.1 *Suppose that Assumption A3 holds. Then, given $\Phi(x^k) > 0$, for all $x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{\underline{m}}$, and $\mu \in \mathbb{R}^{\underline{q}}_+$ satisfying $\underline{h}(x) = 0$, $\underline{g}(x) \leq 0$, and $\min\{\mu, -\underline{g}(x)\} = 0$, whenever*

$$\left\|\nabla\Phi(x) + \underline{h}'(x)^T\lambda + \underline{g}'(x)^T\mu\right\| \leq \frac{\kappa}{\sqrt{2}}\sqrt{\Phi(x^k)} \tag{34}$$

we have that

$$\Phi(x) \leq \frac{1}{2}\Phi(x^k). \tag{35}$$

Proof: Let $x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{\underline{m}}$, and $\mu \in \mathbb{R}^{\underline{q}}_+$ be such that $\underline{h}(x) = 0$, $\underline{g}(x) \leq 0$, and $\min\{\mu, -\underline{g}(x)\} = 0$. If $\Phi(x) = 0$ then the thesis follows trivially. We consider $\Phi(x) > 0$ from now on. Assume that (34) holds and, by contradiction, that (35) does not hold. Then, $\Phi(x) > \frac{1}{2}\Phi(x^k)$ or, equivalently, $\sqrt{\Phi(x^k)} < \sqrt{2}\sqrt{\Phi(x)}$. Thus, by (34),

$$\left\|\nabla\Phi(x) + \underline{h}'(x)^T\lambda + \underline{g}'(x)^T\mu\right\| \leq \frac{\kappa}{\sqrt{2}}\sqrt{\Phi(x^k)} < \kappa\sqrt{\Phi(x)}. \tag{36}$$

Dividing both sides of (36) by $\sqrt{\Phi(x)}$, we obtain the negation of Assumption A3 with multipliers $\lambda/\sqrt{\Phi(x)}$ and $\mu/\sqrt{\Phi(x)}$. □

The purpose of tests (8) and (16) in Algorithms 2.1 and 2.2 is to detect convergence to an infeasible stationary point of the infeasibility, that no longer exists under Assumption A3. Therefore, under Assumption A3, these two tests and parameter γ should be eliminated from the algorithms. The consequence of Lemma 3.1 is that, if Assumption A3 holds, Algorithms 2.1 and 2.2 with tests (8) and (16) suppressed behave exactly as in the case in which the two tests are not suppressed and γ is chosen satisfying $\gamma = \kappa/\sqrt{2}$. Thus, the theorem below follows.

Theorem 3.4 *Suppose that Assumptions A1, A2, and A3 hold, and that, in Algorithms 2.1 and 2.2, parameter γ and tests (8) and (16) are suppressed. Then, during the execution of Algorithm 2.1, the total number of iterations of Algorithm 2.2 is bounded above by*

$$\frac{1}{2\alpha c_p}\left|\log_2\left(\varepsilon/\Phi(x^0)\right)\right| \max\left\{0.99^{\zeta_p^\beta}\Phi(x^0), \left(\frac{\kappa}{\sqrt{2}}\right)^{\zeta_p^\beta} \omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right\} \tag{37}$$

and the total number of evaluations of Φ *and its derivatives performed by Algorithm 2.2 is bounded above by*

$$\frac{1}{2\alpha c_p}\left|\log_2\left(\varepsilon/\Phi(x^0)\right)\right|\left(\max\left\{0.99^{\zeta_p^\beta}\Phi(x^0),\left(\frac{\kappa}{\sqrt{2}}\right)^{\zeta_p^\beta}\omega_p^{1+\frac{1}{2}\zeta_p^\beta}\right\}+\frac{1-\beta}{p+\beta-1}\left|\log_2\left(\frac{\kappa}{\sqrt{2}}\sqrt{\varepsilon}\right)\right|+c_a\right), \tag{38}$$

where

$$\omega_p=\begin{cases}\varepsilon, & \textit{if } p=1,\\ \Phi(x^0), & \textit{if } p>1,\end{cases}$$

$\zeta_p^\beta=-\frac{p+\beta}{p+\beta-1}$*, and* c_p*, given by [21, (26)], and* c_a*, defined in (26), depend only on* α*,* β*,* θ*,* L*,* p*, and* $\sigma_{\min}$*.*

Proof: The bounds in the thesis of this theorem are the bounds given in Theorem 3.3 with γ substituted by $\kappa/\sqrt{2}$. So the thesis follows from Theorem 3.3 and Lemma 3.1. □

The Corollary 3.1 below summarizes the results for the case in which Assumption A3 holds. It should be noted that when Assumption A1 holds with $\beta=1$, i.e. when the pth order derivatives of Φ satisfy a Lipschitz condition, the complexity bounds do not depend on p.

Corollary 3.1 *Suppose that Assumptions A1, A2, and A3 hold, and that, in Algorithms 2.1 and 2.2, parameter* γ *and tests (8) and (16) are suppressed. Then, during the execution of Algorithm 2.1:*

(a) *The total number of iterations of Algorithm 2.2 and the total number of evaluations of* Φ *and its derivatives performed by Algorithm 2.2 are both bounded by* $O(|log(\varepsilon)|)$ *if* $\beta=1$ *and independently of* p*;*

(b) *The total number of iterations of Algorithm 2.2 is bounded by* $O(|log(\varepsilon)|)$ *and the total number of evaluations of* Φ *and its derivatives performed by Algorithm 2.2 is bounded by* $O(|\log(\varepsilon)|^2)$ *if* $\beta<1$ *and* $p>1$*.*

(c) *The total number of iterations of Algorithm 2.2 and the total number of evaluations of* Φ *and its derivatives performed by Algorithm 2.2 are both bounded by* $O(|\log(\varepsilon)|\,\varepsilon^{\frac{\beta-1}{2\beta}})$ *if* $\beta<1$ *and* $p=1$*.*

Proof: The proof follows from Theorem 3.4 by substituting p and β in (37) and (38) by the corresponding values and eliminating all the terms that do not depend on ε. □

It is easy to see that the complexity bound given by Corollary 3.1 is sharp when $p=1$ and $\beta=1$. For that purpose, consider problem (1,2) given by $m=1$, $h(x)=x$, and $q=\underline{m}=\underline{q}=0$, i.e. $\Phi(x)=x^2$. Given x^0 arbitrary and considering $\sigma_{k,0}=2$ and $B_{k,0}=0$ for all k, we have that $x=x^{k,0}-\frac{1}{2\sigma_{k,0}}\nabla\Phi(x^{k,0})$ satisfies (12) and (13). But $x=\frac{1}{2}x^{k,0}$ and, thus, by the definition of Φ, (15) holds. Therefore, $x=x^{k,1}=x^{k+1}$. This implies that, $x^{k+1}=\frac{1}{2}x^k$ and $\Phi(x^{k+1})=\frac{1}{4}\Phi(x^k)$ for all $k\in\mathbb{N}$. Then, for all $k\in\mathbb{N}$, $\Phi(x^k)=\frac{1}{4^k}\Phi(x^0)$. So, $\Phi(x^k)\le\varepsilon$ if and only if $\frac{1}{4^k}\Phi(x^0)\le\varepsilon$ or, equivalently, $k\ge\frac{1}{2}\log_2(\Phi(x^0)/\varepsilon)=\frac{1}{2}|\log_2(\varepsilon/\Phi(x^0)|$.

4 Illustrative examples

In this section, we aim to illustrate with numerical examples the theoretical result given by Corollary 3.1 item (a), i.e. the $O(|\log(\varepsilon)|)$ iterations and evaluations complexity of Algorithm 2.1–2.2 to find x satisfying $\Phi(x) \leq \varepsilon$, $\underline{h}(x) = 0$, and $\underline{g}(x) \leq 0$ under Assumptions A1 (with $p = 1$ and $\beta = 1$), A2, and A3. With this purpose, we implemented Algorithms 2.1 and 2.2 in Fortran 90. In the implementation, the underlying problem at Step 2 of Algorithm 2.2, that consists in minimizing a quadratic function with arbitrary constraints, is solved using Algencan [1, 5]. In the numerical experiments, we set $\alpha = 10^{-8}$, $\sigma_{\min} = 10^{-8}$, and $\theta = 100$.

4.1 Toy illustrative examples

Example 1. Consider the feasibility problem with $m = 2$, $q = 0$, $\underline{m} = 0$, $\underline{q} = 4$, and $n = 2$ given by

$$h(x) = \begin{pmatrix} 10(x_2 - x_1^2) \\ 1 - x_1 \end{pmatrix}$$

and

$$\underline{g}(x) = \begin{pmatrix} x_1 - 2 \\ x_2 - 2 \\ -x_1 - 2 \\ -x_2 - 2 \end{pmatrix}.$$

This means that $\Phi(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ is the popular Rosenbrock function and the cheap constraints (2) are $x \in [-2, 2]$. Assumption A3 holds with $\kappa = 0.66$. We employed Algorithm 2.1–2.2 for minimizing $\Phi(x)$ with $x \in [-2, 2]$ using $B_{k,0}$ as the Barzilai-Borwein-Raydan diagonal estimation of the Hessian as in [3, 7, 24] and $B_{k,\ell} = B_{k,0} + \xi_{k,\ell} I$, where $\xi_{k,\ell} > 0$ for $\ell = 1, 2, \ldots$ is given by a safeguarded backtracking procedure. Starting from $x^0 = (-1.2, 1)^T$, Table 1 displays the sequence $\Phi(x^k)$ together with the accumulated number of iterations of Algorithm 2.2 (#it in the table) and evaluations of Φ (#Φ in the table). The correlations between the values of $|\log(\varepsilon)|$ and the accumulated number of iterations and functional evaluations performed by Algorithm 2.2 are 0.85 and 0.80, respectively, corroborating that the logarithmic estimation is reliable.

Example 2. Let $A \in \mathbb{R}^{n \times n}$ be symmetric and positive definite. Consider the feasibility problem with $m = 1$, $q = 0$, $\underline{m} = 0$, and $\underline{q} = 0$ given by

$$h(x) = \sqrt{\frac{1}{2}(x - \bar{x})^T A (x - \bar{x})}.$$

Then $\Phi(x) = \frac{1}{2}(x - \bar{x})^T A (x - \bar{x})$; thus, $\nabla \Phi(x) = A(x - \bar{x})$ and $\|\nabla \Phi(x)\|_2 = \sqrt{(x - \bar{x})^T A^T A (x - \bar{x})}$. Then, for all $x \in \mathbb{R}^n$,

$$\|\nabla \Phi(x)\| \geq \lambda_{\min} \|x - \bar{x}\|$$

and

$$\sqrt{\Phi(x)} \leq \lambda_{\max} \|x - \bar{x}\|,$$

k	#it	#Φ	$\Phi(x^k)$
0	0	1	2.42000000E+01
1	2	5	4.56006286E+00
2	9	17	1.74588298E+00
3	18	31	6.72291440E−01
4	24	38	1.69811030E−01
5	30	46	4.81342844E−02
6	36	54	3.57124613E−03
7	40	58	5.98641542E−04
8	46	65	1.88139425E−05
9	56	76	1.23125395E−06
10	59	79	4.75271921E−10

Table 1: Details of the execution of Algorithm 2.1–2.2 in the toy illustrative Example 1.

where $\lambda_{\min}$ and $\lambda_{\max}$ are the smallest and the largest eigenvalues of A, respectively. Thus, for all $x \neq \bar{x}$,

$$\frac{\|\nabla \Phi(x)\|}{\sqrt{\Phi(x)}} \geq \frac{\lambda_{\min}}{\lambda_{\max}}.$$

Therefore, Assumption A3 holds with κ equal to the inverse of the condition number of A. We solved this problem being A the diagonal matrix with eigenvalues $1/2, 1/3, \dots, n/(n+1)$ and $\bar{x}_i = 1$ for all i, employing Algorithm 2.1–2.2 with the definition of $B_{k,\ell}$ for all k and ℓ given in the previous example. Starting from $x^0 = 0$, Table 2 displays the sequence $\Phi(x^k)$ together with the accumulated number of iterations of Algorithm 2.2 (#it in the table) and evaluations of Φ (#Φ in the table). As it can be observed, for every k, Algorithm 2.2 performs a single model minimization to obtaining a point that reaches the desired target on $\Phi(\cdot)$. The number of evaluations is always equal to the number of iterations plus one. The "plus one" corresponds to the evaluation of $\Phi(x^0)$. The fact that the number of evaluations is equal to the number of iterations plus one also means that the regularization parameter was never increased. Using the data in the table, we can see that the correlation between $|\log(\varepsilon)|$ and the accumulated number of iterations of Algorithm 2.2 is 0.99. (Of course, the same is true for the accumulated number of function evaluations.)

Example 3. Consider the feasibility problem with $m = \underline{m} = 1$, $q = \underline{q} = 0$, and $n = 2$ given by

$$h(x) = \frac{\pi}{2} - \frac{4}{\pi} \sum_{j=1}^{\tilde{p}} \cos((2j+1)(x_1 + x_2))/(2j+1)^2,$$

$$\underline{h}(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 - 1,$$

and

$$-\pi \leq x_1 + x_2 \leq \pi,$$

k	#it	#Φ	$\Phi(x^k)$
0	0	1	3.99006133D+00
1	1	2	9.67457272D-02
2	2	3	9.07733377D-03
3	3	4	3.54371350D-04
4	4	5	2.25072660D-06
5	5	6	3.86115181D-07
6	6	7	1.35021882D-08
7	7	8	9.67220885D-11

Table 2: Details of the execution of Algorithm 2.1–2.2 in the toy illustrative Example 2.

where $\tilde{p} = 10^8$. Clearly, the cost of evaluating h, that approximates $|x_1 + x_2|$, is $O(\tilde{p})$; while the cost of evaluating $\underline{h}$ is $O(1)$. This means that h is a costly-to-evaluate, when compared to $\underline{h}$, but topologically simple function. Thus, it makes sense, as an illustrative example, to solve this feasibility problem by minimizing a model of $h(x)^2$ subject to $\underline{h}(x) = 0$ and $-\pi \leq x_1 + x_2 \leq \pi$.

In this example, we considered $B_{k,\ell} = \nabla^2 \Phi(x^{k,\ell})$ when $\nabla^2 \Phi(x^{k,\ell})$ is positive definite; and $B_{k,\ell} = \nabla^2 \Phi(x^{k,\ell}) + \xi_{k,\ell} I$, otherwise. In the latter case, $\xi_{k,\ell}$ is the smallest power of 10, not smaller than 10^{-8}, such that $B_{k,\ell}$ is numerically positive definite. Starting from a point that satisfies the cheap constraints, Table 3 displays the sequence $\Phi(x^k)$ together with the accumulated number of iterations of Algorithm 2.2 (#it in the table) and evaluations of Φ (#Φ in the table). The behavior of the method in this example is identical to the one observed in the previous example, i.e. the correlation between $|\log(\varepsilon)|$ and the accumulated number of iterations of and evaluations performed by Algorithm 2.2 is 0.99.

k	#it	#Φ	$\Phi(x^k)$
0	0	1	1.1781930563200053E−01
1	1	2	1.6522725284161665E−02
2	2	3	6.1041411149626642E−04
3	3	4	5.0318577198977945E−06
4	4	5	7.2445391801959614E−07
5	5	6	1.1568585375673346E−07
6	6	7	2.1071181599933004E−09

Table 3: Details of the execution of Algorithm 2.1–2.2 in the toy illustrative Example 3.

4.2 Illustration with problems of the CUTEst collection

In this section, we illustrate the behavior of Algorithm 21–2.2 in 40 feasibility problems from the CUTEst collection [17] with no bound constraints. Selected problems are problems for which Algencan [1, 5] was able to find a feasible point. (Problems for which a solution is found when looking for a point that satisfies the cheap constraints were discarded.) Constraints were arbitrarily divided into two sets: constraints with odd indices were considered expensive constraints; while constraints with even index were considered cheap constraints.

For all k and ℓ, we considered $B_{k,\ell}$ as the Gauss-Newton approximation to $\nabla^2\Phi(x^{k,\ell})$ plus $\xi_{k,\ell}I$, where $\xi_{k,\ell}$ is the smallest power of 10, not smaller than 10^{-8}, such that $B_{k,\ell}$ is numerically positive definite.

For each one of the 40 problem, we recorded the accumulated number of iterations of Algorithm 2.2 and the accumulated number of evaluations of $\Phi(\cdot)$ that were needed to find x such that $\Phi(x) \leq \varepsilon$, with $\varepsilon \in \{10^{-1}, 10^{-2}, \dots, 10^{-8}\}$. Table 4 shows the details. Comparing the number of iterations and the number of evaluations of Φ required to reach $\Phi(x) \leq 10^{-8}$, it can be seen that there are only 10 problems (HIMMELBC, POWELLBS, HELIXNE, HEART6, HEART8, HYDCAR6, HYDCAR20, VANDERM1, VANDERM2, and SPIN2) in which the number of function evaluations is greater than the number of iterations plus one, meaning the sufficient descent condition (17) was not satisfied and the regularization parameter had to be increased at Step 4 of Algorithm 2.2. Columns "Correlation #it" and "Correlation #Φ" show the correlation between the achieved value of $|\log(\Phi(\cdot))|$ and the cumulative number of iterations and evaluations of Φ performed by Algorithm 2.2, respectively. It is easy to see that, when correlations are not close to 1, it is because relations are even better than linear. The last column in the table, named $\hat{\kappa}$, shows a rough estimate of the constant κ in Assumption A3, computed as $\min\{\|\nabla\Phi(x) + \underline{h}'(x)^T\lambda + \underline{g}'(x)^T\mu\|_2/\sqrt{\Phi(x)}\}$ over all the triplets (x, λ, μ) computed at Step 2 of Algorithm 2.2. Note that the computed estimate $\hat{\kappa}$ suggests that κ is bounded away from zero in all problems.

Problem	n	m	p	$\Phi(x^*)$	Number of iterations before reaching $\Phi(\cdot) \leq \varepsilon$								Correlation #it	Number of evaluations of Φ before reaching $\Phi(\cdot) \leq \varepsilon$								Correlation #Φ	$\hat{\kappa}$
					10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}		10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}		
BOOTH	2	2	0	5.6E−16	1	1	1	1	1	1	1	1	1.00	2	2	2	2	2	2	2	2	1.00	1.9E+00
CLUSTER	2	2	0	1.3E−10	0	0	0	0	1	2	2	3	1.00	0	0	0	0	2	3	3	4	1.00	1.8E−01
CUBENE	2	2	0	2.2E−14	1	1	1	1	1	1	1	1	1.00	2	2	2	2	2	2	2	2	1.00	4.8E−01
GOTTFR	2	2	0	1.9E−11	0	0	0	0	1	2	2	3	0.99	0	0	0	0	2	3	3	4	0.99	1.3E+00
HIMMELBA	2	2	0	1.1E−16	1	1	1	1	1	1	1	1	1.00	2	2	2	2	2	2	2	2	1.00	5.7E+00
HIMMELBC	2	2	0	2.9E−15	2	3	3	3	3	4	4	4	0.78	11	12	12	12	12	13	13	13	0.41	8.6E+00
HS8	2	2	0	3.0E−16	1	2	2	2	2	2	3	3	0.94	2	3	3	3	3	3	4	4	0.94	1.4E+01
HYPCIR	2	2	0	3.3E−09	1	1	1	2	2	2	2	2	1.00	2	2	2	3	3	3	3	3	1.00	2.8E+00
POWELLBS	2	2	0	4.8E−14	8	8	8	8	9	9	9	9	0.78	10	10	10	10	11	11	11	11	0.73	1.8E+03
SINVALNE	2	2	0	2.0E−18	1	1	1	1	1	1	1	1	1.00	2	2	2	2	2	2	2	2	1.00	1.4E+01
HATFLDF	3	3	0	1.1E−09	0	0	1	1	7	9	10	11	0.94	0	0	2	2	8	10	11	12	0.94	3.2E+00
HELIXNE	3	3	0	2.6E−20	7	7	7	7	8	8	8	8	0.11	47	47	47	47	48	48	48	48	0.06	1.5E+00
HIMMELBE	3	3	0	2.5E−17	1	2	2	2	2	2	2	2	1.00	2	3	3	3	3	3	3	3	1.00	1.0E+00
RECIPE	3	3	0	4.6E−13	1	2	2	2	2	2	2	2	1.00	2	3	3	3	3	3	3	3	1.00	2.8E−01
ZANGWIL3	3	3	0	1.0E−12	1	1	1	1	1	1	1	1	1.00	2	2	2	2	2	2	2	2	1.00	1.6E+00
POWELLSE	4	4	0	1.3E−09	3	4	5	5	6	7	8	9	0.92	4	5	6	6	7	8	9	10	0.92	4.5E−02
HEART6	6	6	0	4.1E−14	40	260	316	334	348	360	360	360	0.80	223	1534	1870	1978	2062	2123	2123	2123	0.80	1.8E+00
HEART8	8	8	0	1.1E−09	18	18	18	19	19	19	19	19	0.29	126	126	126	127	127	127	127	127	0.32	3.9E+00
COOLHANS	9	9	0	2.2E−14	0	0	0	2	2	2	2	3	0.88	0	0	0	3	3	3	3	4	0.88	6.0E+01
MOREBVNE	10	10	0	3.9E−16	0	1	1	1	1	1	2	2	1.00	0	2	2	2	2	2	3	3	1.00	3.8E+00
OSCIPANE	10	10	0	5.7E−16	1	1	1	1	1	1	2	2	1.00	2	2	2	2	2	2	3	3	1.00	2.9E+03
VARDIMNE	10	12	0	1.7E−24	0	0	0	0	1	1	1	1	1.00	0	0	0	0	2	2	2	2	1.00	1.2E+01
INTEQNE	12	12	0	4.2E−16	0	1	1	1	1	1	2	2	1.00	0	2	2	2	2	2	3	3	1.00	1.6E+00
WATSONNE	12	31	0	3.2E−11	0	0	0	0	0	0	1	1	1.00	0	0	0	0	0	0	2	2	1.00	1.3E+10
HATFLDG	25	25	0	2.7E−13	3	4	4	4	4	5	5	5	0.82	4	5	5	5	5	6	6	6	0.82	2.1E+00
HYDCAR6	29	29	0	6.1E−14	8	16	26	36	47	57	57	57	0.98	19	41	81	121	165	198	198	198	0.98	9.2E+00
METHANB8	31	31	0	4.0E−11	1	1	1	1	2	2	2	2	1.00	2	2	2	2	3	3	3	3	1.00	2.2E+02
METHANL8	31	31	0	1.4E−09	3	4	4	4	4	4	4	4	0.17	4	5	5	5	5	5	5	5	0.17	3.6E+02
CHNRSBNE	50	98	0	2.6E−17	1	1	1	1	1	1	1	1	1.00	2	2	2	2	2	2	2	2	1.00	7.9E+00
HYDCAR20	99	99	0	9.9E−09	4	7	9	12	25	135	226	524	0.85	6	14	20	29	68	396	578	1165	0.87	4.4E−01
VANDERM1	100	100	99	2.1E−09	0	0	0	1	1	2	3	4	1.00	0	0	0	2	2	3	4	5	1.00	2.3E−01
VANDERM2	100	100	99	2.1E−09	0	0	0	1	1	2	3	4	1.00	0	0	0	2	2	3	4	5	1.00	2.3E−01
VANDERM3	100	100	99	7.8E−09	0	0	0	1	2	3	4	4	1.00	0	0	0	2	3	4	5	5	1.00	3.1E−01
SPIN2	102	100	0	8.1E−09	3	3	3	3	3	4	4	5	0.61	4	4	4	4	4	7	7	10	0.70	5.3E+00
ARGTRIG	200	200	0	7.0E−11	1	1	1	2	2	2	2	2	1.00	2	2	2	3	3	3	3	3	1.00	1.6E+02
BROWNALE	200	200	0	1.2E−18	0	0	0	1	1	1	1	1	1.00	0	0	0	2	2	2	2	2	1.00	7.0E+02
CHANDHEU	500	500	0	2.2E−09	3	4	5	5	6	7	7	8	0.76	4	5	6	6	7	8	8	9	0.76	6.0E−01
EIGENAU	2550	2550	0	8.1E−12	1	1	1	1	1	1	1	1	1.00	2	2	2	2	2	2	2	2	1.00	1.1E+02
BROYDN3D	5000	5000	0	4.8E−09	3	3	3	4	4	4	4	4	0.38	4	4	4	5	5	5	5	5	0.38	8.3E+00
BROYDNBD	5000	5000	0	4.8E−14	4	4	4	5	5	5	5	5	0.56	5	5	5	6	6	6	6	6	0.56	5.7E+00

Table 4: Performance of Algorithm 2.1–2.2.

5 Final remarks

The objective of Optimization is to find a point in the feasible region at which the objective function takes a value as small as possible. Optimality conditions, which relate the local variation of the objective function with the local variations of the constraints, are tools for recognizing whether a point is close to a solution or not, but do not have an intrinsic value for most users. Lagrange multipliers are generally used to estimate the variation of the minimum with respect to the variation of different constraints but this utility is challenged in the case that constraint qualifications do not hold or when the set of Lagrange multipliers is infinity. As a matter of fact, the variation of the minimum with respect to constraints is more reliably estimated by means of running the solver with the desired modification of constraints, that does not need to be small. This is the reason why the default version of many constrained optimization solvers, after satisfying a (successful or unsuccessful) stopping criterion at some iterate x, try to find a very accurate point in the feasible region, starting with x as initial approximation. Alternatively, these solvers address the problem of finding a feasible point subject to the additional feasibility constraint $f(z) \leq f(x)$.

Constrained optimization problems are usually formulated in the form

$$\text{Minimize } f(x) \tag{39}$$

subject to

$$h(x) = 0, \; g(x) \leq 0, \; \underline{h}(x) = 0, \; \underline{g}(x) = 0, \tag{40}$$

where the constraints $\underline{h}(x) = 0$ and $\underline{g}(x) = 0$ are cheap in the sense discussed in this work. Augmented Lagrangian (AL) methods are appropriate for these formulations. At each outer iteration of an AL method the augmented Lagrangian function, which combines f, h, and g, is approximately minimized subject to the cheap constraints. See, for example, [5] and [1], where this approach is developed and analyzed. The complexity of solving each subproblem by means of regularization methods is similar to the complexity of solving unconstrained optimization problems. The difficulty of extending this result to the whole constrained minimization process relies on the fact that, in the worst situation, penalty parameters could grow indefinitely, affecting the Lipschitz constants associated with each subproblem. (See [6].) However, in many practical cases, users do not need to "minimize" $f(x)$ and would be happy after finding a feasible point for which $f(x)$ is smaller than a given target f_{target}. In this case, the requirement (39) may be replaced by the inequality $f(x) - f_{\text{target}} \leq 0$, so far defining a feasibility problem together with the constraints (40). In this work, we showed that solving this feasibility problem may be overwhelmingly easier than solving (39,40) by means of constrained optimization solvers.

Acknowledgements. The authors are indebted to Prof. Nick Gould for his comments on a first version of this work. The authors are also indebted to the anonymous referees whose comments helped a lot to improve the submitted version of this work.

References

[1] R. Andreani, E. G. Birgin, J. M. Martínez, and M. L. Schuverdt, On augmented Lagrangian methods with general lower-level constraints, *SIAM Journal on Optimization* 18, pp. 1286–1309, 2007.

[2] F. J. Aragón Artacho, J. M. Borwein, and M. K. Tam, Douglas-Rachford feasibility methods for matrix completion problems, *The ANZIAM Journal* 55, pp. 299–326, 2014.

[3] J. Barzilai and J. M. Borwein, Two point step size gradient methods, *IMA Journal of Numerical Analysis* 8, pp. 141–148, 1988.

[4] A. Beck and M. Teboulle, A linearly convergent algorithm for solving a class of nonconvex/affine feasibility problems, in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, H. Bauschke, R. Burachik, P. Combettes, V. Elser, D. Luke, and H. Wolkowicz (eds), Springer Optimization and Its Applications, volume 49, Springer, New York, NY, pp. 33–48, 2011.

[5] E. G. Birgin and J. M. Martínez, *Practical Augmented Lagrangian Methods for Constrained Optimization*, vol. 10 of Fundamentals of Algorithms, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2014, doi:10.1137/1.9781611973365.

[6] E. G. Birgin and J. M. Martínez, Complexity and performance of an Augmented Lagrangian algorithm, arXiv:1907.02401.

[7] E. G. Birgin, J. M. Martínez, and M. Raydan, Nonmonotone spectral projected gradient methods on convex sets, *SIAM Journal on Optimization* 10, pp. 1196–1211, 2000.

[8] E. G. Birgin, J. L. Gardenghi, J. M. Martínez, S. A. Santos, and Ph. L. Toint, Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models, *Mathematical Programming* 163, pp. 359–368, 2017.

[9] C. Cartis, N. I. M. Gould, and Ph. L. Toint, Adaptive cubic regularization methods for unconstrained optimization. Part I: motivation, convergence and numerical results, *Mathematical Programming* 127, pp. 245–295, 2011.

[10] C. Cartis, N. I. M. Gould, and Ph. L. Toint, Adaptive cubic regularization methods for unconstrained optimization. Part II: worst-case function and derivative complexity, *Mathematical Programming* 130, pp. 295–319, 2011.

[11] C. Cartis, N. I. M. Gould, and Ph. L. Toint, On the evaluation complexity of cubic regularization methods for potentially rank-deficient nonlinear least-squares problems and its relevance to constrained nonlinear optimization, *SIAM Journal on Optimization* 23, pp. 1553–1574, 2013.

[12] C. Cartis, N. I. M. Gould, and Ph. L. Toint, Improved worst-case evaluation complexity for potentially rank-deficient nonlinear least-Euclidean-norm problems using higher-order regularized models, Technical Report NA 15-17, Numerical Analysis Group, University of Oxford, 2015.

[13] C. Cartis, N. I. M. Gould, and Ph. L. Toint, Universal regularization methods: varying the power, the smoothness and the accuracy, *SIAM Journal on Optimization* 29, pp. 595–615, 2019.

[14] J. E. Dennis and R. S. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Prentice Hall, New Jersey, 1983.

[15] G. N. Grapiglia and Yu. Nesterov, Regularized Newton methods for minimizing functions with Hölder continuous Hessians, *SIAM Journal on Optimization* 27, pp. 478–506, 2017.

[16] G. N. Grapiglia and Yu. Nesterov, Accelerated regularized Newton methods for minimizing composite convex functions, *SIAM Journal on Optimization* 29, pp. 77–99, 2019.

[17] N. I. M. Gould, D. Orban, and P. L. Toint, CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization, *Computational Optimization and Applications* 60, pp. 545–557, 2014.

[18] A. O. Herrera, H. D. Scolnik, G. Chichilnisky, G. C. Gallopin, J. E. Hardoy, *Catastrophe or New Society? A Latin America world model*, IDRC, Ottawa, ON, CA, 1976.

[19] C. R. Johnson, Matrix completion problems: a survey, *Proceedings of Symposia in Applied Mathematics* 40, pp. 171–198, 1990.

[20] M. Laurent, Matrix completion problems, in *Encyclopedia of Optimization*, C. A. Floudas and P. Pardalos eds., Springer, Boston, MA, 2008, pp. 1967–1975.

[21] J. M. Martínez, On high-order model regularization for constrained optimization, *SIAM Journal on Optimization* 27, pp. 2447–2458, 2017.

[22] L. Martínez, R. Andrade, E. G. Birgin, and J. M. Martínez, PACKMOL: A package for building initial configurations for molecular dynamics simulations, *Journal of Computational Chemistry* 30, pp. 2157–2164, 2009.

[23] Yu. Nesterov and B. T. Polyak, Cubic regularization of Newton method and its global performance, *Mathematical Programming* 108, pp. 177–205, 2006.

[24] M. Raydan, The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem, *SIAM Journal on Optimization* 7, pp. 26–33, 1997.

[25] M. Yashtini, On the global convergence rate of the gradient descent method for functions with Hölder continuous gradients, *Optimization Letters* 10, pp. 1361–1370, 2016.