SIAM J. OPTIM.
Vol. 30, No. 1, pp. 80–101

ON THE COMPLEXITY OF AN INEXACT RESTORATION METHOD FOR CONSTRAINED OPTIMIZATION*

LUÍS FELIPE BUENO† AND JOSÉ MARIO MARTÍNEZ‡

Abstract. Recent papers indicate that some algorithms for constrained optimization may exhibit worst-case complexity bounds that are very similar to those of unconstrained optimization algorithms. A natural question is whether well-established practical algorithms, perhaps with small variations, may enjoy analogous complexity results. In the present paper we show that the answer is positive with respect to inexact restoration algorithms in which first-order approximations are employed for defining the subproblems.

Key words. complexity, continuous optimization, constrained optimization, first-order methods, inexact restoration methods, regularization

AMS subject classifications. 68Q25, 90C30, 90C55

DOI. 10.1137/18M1216146

1. Introduction.

Inexact restoration (IR) algorithms were introduced with the aim of solving constrained optimization problems [40, 42]. Each iteration of an IR algorithm consists of two phases. In the first phase one improves feasibility and in the second case optimality is improved onto a linear tangent approximation of the constraints. When a sufficient descent criterion does not hold the trial point is modified in such a way that, eventually, acceptance occurs at a point that may be close to the solution of the restoration (first) phase. The acceptance criterion may use merit functions [40, 42] or filters [37]. IR techniques for constrained optimization were improved, extended, and analyzed in [32] and [18], among others.

Several studies on the numerical behavior of IR methods are available. The IR approach was applied to general constrained problems in [9], where the proposed algorithm obtained results compatible with well-established softwares. The IR with a regularization strategy, as used in this work, was proposed in [18], where derivative-free optimization problems were efficiently solved. IR methods are particularly useful when there exists some natural way to restore feasibility. One of the most successful applications of IR is for electronic structure calculations, given in [34]. In this paper the feasible region is the set of projection matrices onto N-dimentional subspaces. The projection onto this set is characterized in terms of the spectral decomposition of the matrix to be projected. Moreover, when spectral decompositions are not affordable, specific iterative methods can be developed that approximate the projection and make possible the fulfillment of the requirements of the restoration phase. These restoration procedures may be interpreted as projected gradient iterations [16]. The tangent subspace may also be represented in a way that make it easy the computation of projections, so that the spectral projected gradient method [17] is also used at the

*Received by the editors September 27, 2018; accepted for publication (in revised form) October 24, 2019; published electronically January 2, 2020.
https://doi.org/10.1137/18M1216146
Funding: The work of the authors was supported by CEPID-CEMeai, FAPESP grants 2018/24293-0 and 15/02528-8, CNPq, and CAPES.
†Institute of Science and Technology, Federal University of São Paulo, São José dos Campos SP, Brazil (lfelipebueno@gmail.com).
‡Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, Campinas, SP, Brazil (martinez@ime.unicamp.br).

optimization phase. Several IR papers are applied to bilevel problems. See, for example, [1, 5, 20]. IR algorithms were also successful when applied to problems with integer variables in [12]. Applications to control problems were given in [6].

Constrained optimization algorithms exhibiting unconstrained-like complexity results were given in [10, 27]. These algorithms are not reliable for practical calculations because they generate short-step sequences, due to the necessity of maintaining approximate feasibility at every iteration. However, the two-phase nature of those algorithms motivated the present research. Unlike the algorithms [10, 27], which employ an initial restoration phase followed by a second phase of short iterations that preserve approximate feasibility, IR algorithms alternate restoration and optimization phases, not limited by feasibility tolerances but subject to a global criterion of quality. In this way, large steps are possible when the current approximation is far from the solution. In this paper we will show that pleasant complexity results hold for an IR algorithm that is similar to the ones presented in [40] and other IR papers.

The analysis of the worst-case function-evaluation complexity of continuous optimization algorithms became relevant in the last 12 years. Given a stopping criterion based on a tolerance $\varepsilon > 0$, one tries to find a (sharp) upper bound for the number of evaluations that a given algorithm needs for satisfying the stopping requirement. In unconstrained optimization, if $f(x)$ is the objective function, the generally accepted first-order stopping criterion corresponds to the approximate annihilation of the gradient, $\|\nabla f(x)\| \leq \varepsilon$. For this case, relevant complexity results regarding global convergence, if the derivatives of the objective function satisfy Lipschitz-continuity requirements (see [7, 8, 14, 22, 23, 24, 26, 29, 38, 45]), are as follows:

1. Suitable gradient-related or quasi-Newton algorithms stop employing, at most, $O(\varepsilon^{-2})$ function and gradient evaluations.
2. Suitable Newton-like algorithms stop employing, at most, $O(\varepsilon^{-3/2})$ function, gradient, and Hessian evaluations.

The "$O(\varepsilon^{-q})$ statement" means that the maximal number of evaluations is smaller than a constant times ε^{-q}, where the constant depends on the initial approximation, parameters of the algorithm, and characteristics of the problem.

Essentially, the results above are also true for constrained problems in which the feasible set is simple enough [28, 41]. Roughly speaking, simplicity of the feasible set means that the minimization of a quadratic function onto such a set is relatively easy to perform.

The papers [10, 27] proved that, for minimization problems with general constraints, $O(\varepsilon^{-q})$ results can be proved that are similar to those in the unconstrained case, for some algorithms based on a single feasibility phase, followed by a short-step optimization phase.

However, the short-step characteristic of the algorithms [10, 27] makes them unsuitable for practical computations. Very short steps are necessary to guarantee pleasant worst-case behavior but the number of short steps that are necessary to prove complexity is close to the number of short steps that one would employ in a sensible implementation of the algorithms.

On the other hand, Cartis, Gould, and Toint [25] considered the complexity of nonconvex equality-constrained optimization employing a first-order exact penalty method. Under the assumption that penalty parameters are bounded, the complexity for finding an approximate KKT point was proved to be $O(\varepsilon^{-2})$.

This state of facts motivated us to study well-established constrained optimization algorithms from the point of view of worst-case complexity. The analogy (feasibility and optimality phases) between the algorithms [10, 27] and the framework of IR methods [18, 32, 40, 42] led us to define a suitable regularized form of IR and to study its complexity properties.

In this paper we analyze a first-order version of IR (no second-derivative information will be used) and we prove that the computer work necessary to achieve suitable stopping criteria is smaller than a constant times $(\epsilon_{feas}^{-1} + \epsilon_{opt}^{-2})$, where ϵ_{feas} is the feasibility tolerance, ϵ_{opt} is the optimality tolerance with respect to the approximate gradient projection (AGP) condition [43], and the constant depends on algorithmic parameters and characteristics of the optimization problem such as function bounds and Lipschitz constants.

In section 2 we present the main algorithm and we state some basic results. In section 3 we prove the complexity results, and in section 4 we state conclusions and lines for future research.

Notation. The symbol $\|\cdot\|$ will denote the Euclidean norm on $\mathbb{R}^n$. $P_D(z)$ denotes the Euclidean projection of z onto the convex set D. In other words, $P_D(z)$ is the point of D that is closest to z. We denote $\mathbb{N} = \{0, 1, 2, \dots\}$.

2. Inexact restoration algorithm.

The problem considered in this paper is

$$\text{Minimize } f(x) \quad \text{subject to} \quad h(x) = 0 \quad \text{and} \quad x \in \Omega, \tag{2.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^m$, and Ω is a nonempty, convex, and compact polytope. In most practical cases Ω is an n-dimensional box $\ell \le x \le u$. The generalization of our results for a general compact and convex Ω is also possible subject to constraint-qualification assumptions on Ω. The compactness assumption is only needed to ensure that the points generated by the algorithm lie in a bounded set. For that purpose one could also assume bounded level sets or strict convexity conditions.

We define, for all $x \in \mathbb{R}^n$,

$$c(x) = \frac{1}{2}\|h(x)\|^2. \tag{2.2}$$

Throughout this paper we will assume that the functions f and h are continuously differentiable. Then, by the compactness of Ω, there exists nonnegative constants C_f, C_h, L_f, and L_h such that, for all $x \in \Omega$,

$$f(x) \le C_f, \tag{2.3}$$

$$\|h(x)\| \le C_h, \tag{2.4}$$

$$\|\nabla f(x)\| \le L_f. \tag{2.5}$$

and

$$\|\nabla h(x)\| \le L_h. \tag{2.6}$$

By (2.5) an (2.6) we have that, for all $x, z \in \Omega$,

$$|f(x) - f(z)| \le L_f \|x - z\| \tag{2.7}$$

and

$$\|h(x) - h(z)\| \le L_h \|x - z\|. \tag{2.8}$$

Moreover, we will also assume that there exist nonnegative constants $L_{\nabla f}$ and $L_{\nabla h}$ such that, for all $x, z \in \Omega$,

$$\|\nabla f(x) - \nabla f(z)\| \le L_{\nabla f}\|x - z\| \tag{2.9}$$

and

$$\|\nabla h(x) - \nabla h(z)\| \le L_{\nabla h}\|x - z\|. \tag{2.10}$$

Defining $L_{\nabla c} \equiv L_h^2 + L_h L_{\nabla h}$, by (2.6), (2.8), and (2.10) we have that, for all $x, z \in \Omega$,

$$\begin{aligned} \|\nabla c(x) - \nabla c(z)\| &\le \|\nabla h(x)h(x) - \nabla h(x)h(z)\| + \|\nabla h(x)h(z) - \nabla h(z)h(z)\| \\ &\le \|\nabla h(x)\|\|h(x) - h(z)\| + \|\nabla h(x) - \nabla h(z)\|\|h(z)\| \\ &\le L_{\nabla c}\|x - z\|. \end{aligned} \tag{2.11}$$

By (2.9), (2.10), and (2.11) we have that, for all $x, z \in \Omega$,

$$f(z) \le f(x) + \nabla f(x)^T(z - x) + L_{\nabla f}\|z - x\|^2, \tag{2.12}$$

$$\|h(z)\| \le \|h(x) + \nabla h(x)^T(z - x)\| + L_{\nabla h}\|z - x\|^2, \tag{2.13}$$

and

$$c(z) \le c(x) + \nabla c(x)^T(z - x) + L_{\nabla c}\|z - x\|^2. \tag{2.14}$$

The (unknown) constants C_f, C_h, L_f, $L_{\nabla f}$, L_h, and $L_{\nabla h}$ will be called *characteristics of the problem* (2.1). The complexity results to be proved in this paper will be of the form

$$\text{Computer Work} \;\le\; \text{Constant} \;\times \left(\epsilon_{feas}^{-1} + \epsilon_{opt}^{-2}\right),$$

where "Constant" only depends on characteristics of the problem and algorithmic parameters, defined below. Moreover, ϵ_{feas} is a tolerance related to infeasibility, and ϵ_{opt} is a tolerance for optimality. Throughout the paper, expressions of the form $a = O(b)$ or $a \in O(b)$ will mean that the nonnegative quantity a is not bigger than a constant times the quantity b, where the constant only depends on characteristics of the problem and algorithmic parameters. In order to be more precise, we will also analyze the constant dependence on the characteristics of the problem.

For all $x \in \Omega$, and $\theta \in (0, 1)$, we define the merit function $\Phi(x, \theta)$ by

$$\Phi(x, \theta) = \theta f(x) + (1 - \theta)\|h(x)\|. \tag{2.15}$$

The main algorithm considered in this paper is presented below. Unlike other optimization algorithms for which complexity has been analyzed [10, 27], the description of Algorithm 2.1 does not depend of the possible stopping parameters ϵ_{feas} and ϵ_{opt}. In particular, we describe the algorithm without a stopping criterion regarding ϵ_{feas} and ϵ_{opt}. This makes it easy to show that the asymptotic convergence results follow as trivial consequences of the complexity ones. If restoration breakdown does not occur, the algorithm generates an infinite sequence and complexity results will follow from bounds on the number of iterations at which desired precisions are not achieved.

Algorithm 2.1. Inexact restoration.

Step 0 Initialization.
Let $\gamma > 0$, $M \geq 1$, $\kappa > 0$, $\mu_{max} \geq \mu_{min} > 0$, $\mu_{-1} = \mu_{max}$, $\sigma_{-1} = \mu_{max}$, $\theta_0 \in (0, 1)$, $r \in (0, 1)$, and $r_{feas} \in (0, r)$ be *algorithmic parameters.* Let $x^0 \in \Omega$. Set $k \leftarrow 0$.
Step 1 Restoration phase.
Compute $y^k \in \Omega$ using Algorithm 2.2 below with parameters M, r, r_{feas}, μ_{min}, and σ_{k-1}.
Test the inequality

$$\left\|h\left(y^k\right)\right\| \leq r\left\|h\left(x^k\right)\right\|. \tag{2.16}$$

If (2.16) does not hold, stop Algorithm 2.1 declaring *restoration failure.*
Step 2 Penalty parameter.
If

$$\Phi\left(y^k, \theta_k\right) - \Phi\left(x^k, \theta_k\right) \leq \frac{1}{2}(1-r)\left(\left\|h\left(y^k\right)\right\| - \left\|h\left(x^k\right)\right\|\right) \tag{2.17}$$

set $\theta_{k+1} = \theta_k$.
Else, compute

$$\theta_{k+1} = \frac{(1+r)\left(\|h\left(x^k\right)\| - \|h\left(y^k\right)\|\right)}{2\left[f\left(y^k\right) - f\left(x^k\right) + \|h\left(x^k\right)\| - \|h\left(y^k\right)\|\right]}. \tag{2.18}$$

Step 3 Optimization phase.
Choose $\mu \in [\mu_{min}, \mu_{k-1}]$ and $H_k \in \mathbb{R}^{n\times n}$, symmetric, such that $\|H_k\| \leq M$.
Step 3.1 Tangent set minimization.
Compute $x \in \mathbb{R}^n$ an approximate solution of the following subproblem:

(2.19)

$$\begin{array}{ll} \text{Minimize} & \nabla f\left(y^k\right)^T\left(x-y^k\right) + \frac{1}{2}\left(x-y^k\right)^T H_k\left(x-y^k\right) + \mu\|x-y^k\|^2 \\ \text{subject to} & \nabla h\left(y^k\right)^T\left(x-y^k\right) = 0, \\ & x \in \Omega. \end{array}$$

(The sense in which the solution of (2.19) must be approximated by x will be given later in Assumptions 3.2 and 3.5. Assumption 3.5 depends on the algorithmic parameter κ.)
Step 3.2 Descent conditions.
Test the conditions

$$f(x) \leq f\left(y^k\right) - \gamma\|x-y^k\|^2 \tag{2.20}$$

and

$$\Phi(x, \theta_{k+1}) \leq \Phi\left(x^k, \theta_{k+1}\right) + \frac{1}{2}(1-r)\left(\|h\left(y^k\right)\| - \|h(x^k)\|\right). \tag{2.21}$$

If both (2.20) and (2.21) are fulfilled, define $\mu_k = \mu$, $x^{k+1} = x$, update $k \leftarrow k+1$, and go to Step 1.
Else, update

$$\mu \in [2\mu, 10\mu] \tag{2.22}$$

and go to Step 3.1.

Algorithm 2.2. Restoration procedure.

Step 1 If $\|h(x^k)\| = 0$ set $y^k = x^k$, $\sigma_k = \sigma_{k-1}$ and return to Algorithm 2.1.
Step 2 Compute

$$c_{target} = \frac{1}{2} r^2 \|h\left(x^k\right)\|^2 \text{ and } \epsilon_c = r_{feas}\|h\left(x^k\right)\|. \tag{2.23}$$

Step 3 Initialize $z^0 = x^k$, $\sigma_{k,-1} = \sigma_{k-1}$, and $\ell \leftarrow 0$.
Step 4 If $c(z^\ell) \le c_{target}$ or $\|P_\Omega(z^\ell - \nabla c(z^\ell)) - z^\ell\| \le \epsilon_c$ set $y^k = z^\ell$, $\sigma_k = \sigma_{k,\ell-1}$, and return to Algorithm 2.1. Else, choose $\sigma \in [\mu_{min}, \sigma_{k,\ell-1}]$ and $B_\ell \in \mathbb{R}^{n\times n}$, symmetric and positive semidefinite, such that $\|B_\ell\| \le M$.
Step 4.1 Find $z \in \Omega$ an approximate solution of the following subproblem:

$$\begin{array}{ll} \text{Minimize} & \nabla c\left(z^\ell\right)^T\left(z - z^\ell\right) + \frac{1}{2}\left(z - z^\ell\right)^T B_\ell \left(z - z^\ell\right) + \frac{\sigma}{2}\|z - z^\ell\|^2 \\ \text{subject to} & z \in \Omega. \end{array} \tag{2.24}$$

(The sense in which the solution of (2.24) must be approximated by z will also be given later, in Assumptions 2.1, 2.2, and 2.3.)
Step 4.2 Test the condition

$$c(z) \le c\left(z^\ell\right) - \gamma\|z - z^\ell\|^2. \tag{2.25}$$

If (2.25) is fulfilled, define $z^{\ell+1} = z$, $\sigma_{k,\ell} = \sigma$, update $\ell \leftarrow \ell + 1$, and go to Step 4. Else, update

$$\sigma \in [2\sigma, 10\sigma] \tag{2.26}$$

and go to Step 4.1.

Algorithm 2.2 below describes the restoration procedure. This algorithm is similar to Algorithm 3.1 of [15] applied to the function $c(y) = \frac{1}{2}\|h(y)\|^2$ with appropriate choices of the target and the precision.

Remarks. The factors 2 and 10 in (2.22) and (2.26) represent the minimal and maximal increasing factors for the regularization parameter. Of course, they could be replaced by arbitrary constants bigger than 1. Practical experience of previous regularization schemes showed us that 2 and 10 are reasonable values. The initialization of the parameters σ and μ can also be done according to standard rules of regularization schemes. At first, making $\mu = \mu_{k-1}$ and $\sigma = \sigma_{k-1}$ would result in lower bounds in the complexity analysis of the algorithm. On the other hand, the reduction of these parameters could produce small values of σ or μ, implying the acceptance of larger steps which, in turn, can improve the practical performance of the algorithm. We believe that it is better to formulate the algorithm in its best practical version rather than to take profit of theoretical advantages for worst-case situations.

The use of H_k and B_ℓ allows one to employ quasi-Newton approximations of the Hessian. Since we do not mean that sufficient quasi-Newton properties are satisfied (like, say, the Dennis–Moré condition) we only assume that these matrices are bounded, since this is a condition easy to check and safeguard.

The condition $\|P_\Omega(y^k - \nabla c(y^k)) - y^k\| \le \varepsilon$ says that the projected gradient of the infeasibility measure is small. Then, this condition must be interpreted as a signal that we are close to a minimizer of the infeasibility. Minimizers of the infeasibility that are not feasible points are unavoidable in every algorithm for constrained optimization. For example, they necessarily appear when the feasible region is empty, but also when a strong local minimizer of the infeasibility unfortunately occurs. Although such situations are not possible if the feasible set is nonempty and $c(.)$ is convex, they are relatively frequent in the presence of nonconvexity. As a consequence, every nonlinear programming algorithm possesses a way of detecting nonfeasible local minimizers of infeasibility. In our algorithm we consider that improving feasibility is not possible when the condition of small projected gradient of c is combined with a relatively large value of $\|h(y)\|$. These are the situations in which Algorithm 2.1 declares restoration failure.

Algorithm 2.1 stops, employing a finite number of iterations, only when restoration fails. Failure of restoration is declared when a point y^k is found such that $\|P_\Omega(y^k - \nabla c(y^k)) - y^k\| \le r_{feas}\|h(x^k)\|$ but $\|h(y^k)\| > r\|h(x^k)\|$. Therefore,

$$\left\|P_\Omega\left(y^k - \nabla c\left(y^k\right)\right) - y^k\right\| \le r_{feas}\left\|h\left(x^k\right)\right\| \tag{2.27}$$

but

$$\left\|h\left(x^k\right)\right\| < \frac{1}{r}\left\|h\left(y^k\right)\right\|. \tag{2.28}$$

By (2.27) and (2.28),

$$\left\|P_\Omega\left(y^k - \nabla h\left(y^k\right)h\left(y^k\right)\right) - y^k\right\| \le \frac{r_{feas}}{r}\left\|h\left(y^k\right)\right\|. \tag{2.29}$$

If $r_{feas} << r$ this means that the infeasibility of y^k is considerably bigger than the projected gradient of the sum of infeasibility squares at y^k. If $\|h(y^k)\|$ is not small, (2.29) probably indicates proximity to a local minimizer of $\|h(y)\|^2$ at which $\|h(y)\|$ does not vanish. On the other hand, if $\|h(y^k)\|$ is small, the fulfillment of (2.29) reflects the nonfulfillment of a desirable constraint qualification. For example, if $y^k - \nabla h(y^k)h(y^k)$ is interior to Ω, (2.29) implies that there exists $v \in \mathbb{R}^n$ such that $\|v\| = 1$ and

$$\left\|\nabla h\left(y^k\right)v\right\| \le \frac{r_{feas}}{r}.$$

This property probably indicates the proximity of a feasible point at which the gradients $\nabla h_1(y), \ldots, \nabla h_m(y)$ are not linearly independent.

The following hypotheses define the conditions that an iterate should satisfy in order to be considered an approximate solution of the subproblem (2.24).

Assumption 2.1. For a fixed k, consider the iterates z^ℓ generated by Algorithm 2.2. We assume that, for all k and ℓ, the approximate solution z of the quadratic programming problem (2.24) satisfies

$$\nabla c\left(z^\ell\right)^T\left(z - z^\ell\right) + \frac{1}{2}\left(z - z^\ell\right)^T B_\ell\left(z - z^\ell\right) + \frac{\sigma}{2}\left\|z - z^\ell\right\|^2 \le 0. \tag{2.30}$$

Assumption 2.2. For a fixed k, consider the iterates z^ℓ generated by Algorithm 2.2. We assume that for every k and ℓ such that $z^{\ell+1} \ne z^\ell$, the approximate solution of the quadratic programming problem (2.24) satisfies

$$\left\|z^{\ell+1} - z^\ell\right\| - t_* \le \kappa t_*, \tag{2.31}$$

where, for $v = \frac{z^{\ell+1} - z^\ell}{\|z^{\ell+1} - z^\ell\|}$, t_* is the minimizer of $\varphi(t) = t\nabla c(z^\ell)^T v + \frac{t^2}{2} v^T(B_\ell + \sigma_{k,l} I)v$ subject to $z^\ell + tv \in \Omega$.

Assumption 2.3. For a fixed k, consider the iterates z^ℓ generated by Algorithm 2.2. For all k and ℓ, the approximate solution of the quadratic programming problem (2.24) satisfies

(2.32)
$$\left\|P_\Omega\left(z^{\ell+1} - \left[\nabla c\left(z^\ell\right) + B_\ell\left(z^{\ell+1} - z^\ell\right) + \sigma\left(z^{\ell+1} - z^\ell\right)\right]\right) - z^\ell\right\| \le \kappa \left\|z^{\ell+1} - z^\ell\right\|.$$

Assumption 2.1 merely states that the objective function of the subproblem (2.24) must decrease with respect to z^ℓ. This condition is satisfied by most quadratic minimization algorithms if we use z^ℓ as the starting point in the iterative process.

Note that, by Assumption 2.1 and the fact that B_ℓ is positive semidefinite, $\nabla c(z^\ell)^T v < 0$ and so $t_* > 0$. Assumption 2.2 holds if $z^{\ell+1}$ is not too far from $z^\ell + t_* v$. Since there is a closed formula for t_*, the minimizer of the one-dimensional parabola $\varphi(t)$ subject to the real interval defined by $z^\ell + tv \in \Omega$, this requirement can be easily satisfied.

Assumption 2.3 refers to the stopping criterion used when solving the subproblem (2.24). Since the constraints of (2.24) are linear, the exact solution of (2.24) has a null projected gradient. Therefore, any algorithm that generates sequences converging to the solution can reach the requested criterion. Complexity proofs of optimization algorithms based on regularization are generally based on the relation of a decreasing requirement for some merit function and the tolerance for solving subproblems. In our case, the decreasing requirement is (2.25). Roughly speaking, this requirement implies that $\|z^{\ell+1} - z^\ell\|$ tends to zero. On the other hand, the inequality (2.32) will impose that the projected gradient of the subproblem will also go to zero at least as fast as the increment $\|z^{\ell+1} - z^\ell\|$. This is crucial in order to bound the number of times in which the projected gradient can be large. The argument in Lemma 2.4 proving a complexity bound for Algorithm 2.2 would not be valid if one replaces $\kappa\|z^{\ell+1} - z^\ell\|$ with a fixed small tolerance.

There exist many efficient algorithms for solving quadratic programming problems. Classical books that address quadratic programming problems are, for example, [30, 35, 46]. More recent approaches can be found in [21, 33]. Moreover, since (2.24) is a strictly convex programming problem, there are efficient algorithms that exhibit finite convergence results to the exact solution. This can be done, for instance, by using active set strategies (see [46, section 16.4]) or using augmented Lagrangian methods [21]. In this way, it is reasonable to ask even for the exact solution of the subproblems, which would trivially satisfy Assumptions 2.1, 2.2, and 2.3.

The next lemma shows a bound on the amount of iterations, constraints, and derivatives evaluations at each call of Algorithm 2.2. In our complexity analysis we do not consider the cost of solving subproblems, implicitly assuming that the evaluation of functions and derivatives dominates the overall computational effort.

LEMMA 2.4. *Suppose that the approximate solutions of* (2.24) *satisfy Assumptions* 2.1 *and* 2.3. *Define* $C_\sigma \equiv \max\{20L_{\nabla c} + 20\gamma, \mu_{max}\}$. *Then, Algorithm* 2.2 *finishes finding* $y^k \in \Omega$ *that satisfies*

$$c\left(y^k\right) \le c_{target} \tag{2.33}$$

or

$$\left\|P_\Omega\left(y^k - \nabla c\left(y^k\right)\right) - y^k\right\| \le \epsilon_c \tag{2.34}$$

employing, at most, $N_R \equiv \lfloor \frac{(1-r^2)(C_\sigma + L_{\nabla c} + M + \kappa)^2}{2\gamma r_{feas}^2} \rfloor + 1$ *iterations and evaluations of the derivatives of* h*. Moreover, for all* k *and* ℓ*,* $\sigma_{k,\ell} \le C_\sigma$*, and so the number of evaluations of* h *is bounded by* $N_R N_{regfeas}$*, where* $N_{regfeas} \equiv \lfloor \log_2(\frac{C_\sigma}{\mu_{min}}) \rfloor + 1$.

Proof. By (2.14) and Assumption 2.1, we have that

$$\begin{aligned} c(z) - c\left(z^\ell\right) &\le \nabla c\left(z^\ell\right)^T \left(z - z^\ell\right) + L_{\nabla c} \left\|z - z^\ell\right\|^2 \\ &= \nabla c\left(z^\ell\right)^T \left(z - z^\ell\right) + \frac{1}{2}\left(z - z^\ell\right)^T B_\ell \left(z - z^\ell\right) + \frac{\sigma}{2}\|z - z^\ell\|^2 - \frac{\sigma}{2}\|z - z^\ell\|^2 \\ &\quad - \frac{1}{2}\left(z - z^\ell\right)^T B_\ell \left(z - z^\ell\right) + L_{\nabla c}\|z - z^\ell\|^2 \\ &\le \left(-\frac{\sigma}{2} + L_{\nabla c}\right) \left\|z - z^\ell\right\|^2. \end{aligned}$$

Therefore, if $\frac{\sigma}{2} \ge L_{\nabla c} + \gamma$, we have that (2.25) holds. Moreover, by (2.26),

$$\sigma \le C_\sigma \equiv \max\{20 L_{\nabla c} + 20\gamma, \mu_{max}\} \tag{2.35}$$

and, at each iteration of Algorithm 2.2, the descent condition (2.25) is tested at most $N_{regfeas}$ times.

On the other hand, by the contraction property of projections, the boundedness of B_l, and (2.11),

$$\begin{aligned} &\left\|P_\Omega\left(z^{\ell+1} - \nabla c\left(z^{\ell+1}\right)\right) - z^{\ell+1} - P_\Omega\left(z^{\ell+1} - \left[\nabla c\left(z^\ell\right) + B_\ell\left(z^{\ell+1} - z^\ell\right)\right]\right) + z^{\ell+1}\right\| \\ &\qquad \le \left\|\nabla c\left(z^{\ell+1}\right) - \nabla c\left(z^\ell\right)\right\| + M\|z^{\ell+1} - z^\ell\| \le \left(L_{\nabla c} + M\right)\left\|z^{\ell+1} - z^\ell\right\|. \end{aligned}$$

Thus,

$$\begin{aligned} \left\|P_\Omega\left(z^{\ell+1} - \nabla c\left(z^{\ell+1}\right)\right) - z^{\ell+1}\right\| &\le \left\|P_\Omega(z^{\ell+1} - [\nabla c\left(z^\ell\right) + B_k\left(z^{\ell+1} - z^\ell\right)]) - z^{\ell+1}\right\| \\ &\quad + (L_{\nabla c} + M)\|z^{\ell+1} - z^\ell\|. \end{aligned} \tag{2.36}$$

Now, again by the contraction property of projections and by Assumption 2.3,

$$\begin{aligned} &\left\|P_\Omega\left(z^{\ell+1} - [\nabla c\left(z^\ell\right) + B_k\left(z^{\ell+1} - z^\ell\right)]\right) - z^{\ell+1}\right\| \\ &\quad \le \|P_\Omega\left(z^{\ell+1} - \left[\nabla c\left(z^\ell\right) + B_k\left(z^{\ell+1} - z^\ell\right)\right]\right) - \\ &\qquad P_\Omega\left(z^{\ell+1} - \left[\nabla c\left(z^\ell\right) + B_k\left(z^{\ell+1} - z^\ell\right) + \sigma\left(z^{\ell+1} - z^\ell\right)\right]\right)\| \\ &\qquad + \left\|P_\Omega\left(z^{\ell+1} - \left[\nabla c\left(z^\ell\right) + B_k\left(z^{\ell+1} - z^\ell\right) + \sigma\left(z^{\ell+1} - z^\ell\right)\right]\right) - z^{\ell+1}\right\| \\ &\quad \le \sigma\left\|z^{\ell+1} - z^\ell\right\| + \kappa\left\|z^{\ell+1} - z^\ell\right\|. \end{aligned} \tag{2.37}$$

Therefore, (2.36) and (2.35) imply that

$$\left\|P_\Omega\left(z^{\ell+1} - \nabla c\left(z^{\ell+1}\right)\right) - z^{\ell+1}\right\| \le \left(C_\sigma + L_{\nabla c} + M + \kappa\right)\left\|z^{\ell+1} - z^\ell\right\|. \tag{2.38}$$

By (2.25) we have that

$$c\left(z^{\ell+1}\right) \le c\left(x^k\right) - \gamma \sum_{j=0}^{\ell} \left\|z^{j+1} - z^j\right\|^2.$$

So, if $\|z^{j+1} - z^j\| > \frac{\epsilon_c}{C_\sigma + L_{\nabla c} + M + \kappa}$ for all $j \leq \ell$, we have

$$c\left(z^{\ell+1}\right) \leq c\left(x^k\right) - \gamma\ell \frac{\epsilon_c^2}{(C_\sigma + L_{\nabla c} + M + \kappa)^2}.$$

In this case, using the definition of c_{target} and ϵ_c in (2.23), Algorithm 2.2 finishes finding $y^k \in \Omega$ that satisfies (2.33) after at most N_R iterations. By (2.38), if $\|z^{j+1} - z^j\| \leq \frac{\epsilon_c}{C_\sigma + L_{\nabla c} + M + \kappa}$, then condition (2.34) holds, and so no more than N_R iterations are made until the stopping criterion is reached.

Since Algorithm 2.2 performs one evaluation of ∇c per iteration and one evaluation of c each time that condition (2.25) is checked, the desired result is proved. ∎

The objective of the following results is to show that the distance between x^k and y^k is bounded by a multiple of the infeasibility measure.

LEMMA 2.5. *For a fixed k, consider the iterates z^ℓ generated by Algorithm 2.2. Suppose that Assumptions 2.1 and 2.2 hold. Then, for all k and ℓ, we have that*

$$\|z^{\ell+1} - z^\ell\| \leq \frac{(1+\kappa)L_h}{\mu_{min}} \|h\left(x^k\right)\|.$$

Proof. The result is obvious if $z^{\ell+1} = z^\ell$. Otherwise, consider the function $\varphi(t)$ given in Assumption 2.2. Once again, since $\nabla c(z^\ell)^T v < 0$, the unconstrained minimizer of this parabola is $\bar{t} > 0$. Using the Cauchy–Schwarz inequality and the fact that $\|v\| = 1$, we have that

$$\bar{t} = -\frac{\nabla c\left(z^\ell\right)^T v}{v^T(B_\ell + \sigma_k I)v} \leq -\frac{\nabla c\left(z^\ell\right)^T v}{v^T \sigma I v} \leq \frac{\|\nabla c\left(z^\ell\right)\|}{\mu_{min}}. \tag{2.39}$$

By the convexity of Ω and the form of $\varphi(t)$ we have that $t_* \leq \bar{t}$. But, by Assumption 2.2, (2.39), and (2.6),

$$\left\|z^{\ell+1} - z^\ell\right\| \leq (1+\kappa)t_* \leq \frac{(1+\kappa)\|\nabla c\left(z^\ell\right)\|}{\mu_{min}} \leq \frac{(1+\kappa)L_h}{\mu_{min}} \|h\left(z^\ell\right)\|. \tag{2.40}$$

Since $c(z^\ell) \leq c(x^k)$ we have that $\|h(z^\ell)\| \leq \|h(x^k)\|$, so the thesis follows from (2.40). ∎

LEMMA 2.6. *For a fixed k, consider the iterates z^ℓ generated by Algorithm 2.2. Let N_R be such as in Lemma 2.4 and define $\beta \equiv \frac{(1+\kappa)N_R L_h}{\mu_{min}}$. Then, for all y^k computed at Step 1 of Algorithm 2.1, we have that*

$$\|y^k - x^k\| \leq \beta \|h\left(x^k\right)\|. \tag{2.41}$$

Proof. Let N_{Rk} be the number of iterations performed by Algorithm 2.2 at iteration k of Algorithm 2.1. By Lemma 2.4 we have that $N_{Rk} \leq N_R$. Then, by Lemma 2.5,

$$\|y^k - x^k\| = \left\|\sum_{l=1}^{N_{Rk}} \left(z^\ell - z^{\ell-1}\right)\right\| \leq N_R \frac{(1+\kappa)L_h}{\mu_{min}} \|h\left(x^k\right)\|.$$

∎

3. Complexity and convergence. Throughout this section we consider sequences $\{x^k\}$ and $\{y^k\}$ generated by Algorithm 2.1. These sequences are defined for all $k \in \mathbb{N}$, except if stopping occurs at some y^k with the diagnostic of failure of restoration. The main results of this section say that, given an arbitrary $\varepsilon > 0$, the number of iterations such that the norm of infeasibility is bigger than ε is $O(\varepsilon^{-1})$ and that the number of iterations such that the norm of a vector that represents optimality is bigger than ε is $O(\varepsilon^{-2})$. As a consequence, we will obtain global convergence of the algorithm and suitable stopping criteria.

The first technical lemma states that the penalty parameters $\{\theta_k\}$ are bounded away from zero.

LEMMA 3.1. *Suppose that Assumptions* 2.1, 2.2, *and* 2.3 *hold and that β is defined such as in Lemma* 2.6. *Given x^k and y^k satisfying* (2.16), *Step* 2 *of Algorithm* 2.1 *is well defined. Moreover, for all k, $\theta_{k+1} \le \theta_k$, the inequality*

$$\Phi\left(y^k, \theta_{k+1}\right) - \Phi\left(x^k, \theta_{k+1}\right) \le \frac{1}{2}(1-r)(\|h\left(y^k\right)\| - \|h(x^k)\|) \tag{3.1}$$

is fulfilled, and defining $\bar{\theta} \equiv \min\{\theta_0, \frac{1+r}{2}(\frac{L_f\beta}{1-r}+1)^{-1}\}$ we have that

$$\theta_k \ge \bar{\theta}. \tag{3.2}$$

Proof. We will first prove that Step 2 is well defined and that $0 < \theta_{k+1} \le \theta_k$. If $\|h(y^k)\| - \|h(x^k)\| = 0$ and (2.16) holds, then $\|h(y^k)\| = \|h(x^k)\| = 0$ and, by Step 1 of Algorithm 2.2, $y^k = x^k$ and so $\Phi(x^k, \theta_k) = \Phi(y^k, \theta_k)$. Thus, (2.17) holds in this case and, consequently, $\theta_{k+1} = \theta_k > 0$.

Therefore, it remains to consider only the case in which $\|h(y^k)\| < \|h(x^k)\|$. In this case, we obtain that

$$\|h\left(x^k\right)\| - \|h\left(y^k\right)\| + \frac{1-r}{2}(\|h\left(y^k\right)\| - \|h\left(x^k\right)\|) = \frac{1+r}{2}(\|h\left(x^k\right)\| - \|h\left(y^k\right)\|) > 0. \tag{3.3}$$

By direct calculations, the inequality (2.17) is equivalent to

$$\begin{aligned}\theta_k\left[f\left(y^k\right) - f\left(x^k\right) + \|h\left(x^k\right)\| - \|h\left(y^k\right)\|\right] \le{}& \|h\left(x^k\right)\| - \|h\left(y^k\right)\| \\ &+ \frac{1-r}{2}(\|h\left(y^k\right)\| - \|h\left(x^k\right)\|).\end{aligned} \tag{3.4}$$

Thus, by (3.3) and the fact that $\theta_k > 0$, the requirement (2.17) is fulfilled whenever $f(y^k) - f(x^k) + \|h(x^k)\| - \|h(y^k)\| \le 0$. In this case, the algorithm also chooses $\theta_{k+1} = \theta_k > 0$.

Therefore, we only need to consider the case in which

$$f\left(y^k\right) - f\left(x^k\right) + \|h\left(x^k\right)\| - \|h\left(y^k\right)\| > 0.$$

In this case, both the numerator and the denominator of (2.18) are positive and so $\theta_{k+1} > 0$. Moreover, if (2.17) does not hold, then, by (3.4), we have that

$$\Phi(y^k, \theta) - \Phi(x^k, \theta) > \frac{1-r}{2}(\|h\left(y^k\right)\| - \|h(x^k)\|)$$

for all $\theta \ge \theta_k$. Now, since the choice (2.18) obviously implies that

$$\begin{aligned}&\theta_{k+1}\left[f\left(y^k\right) - f\left(x^k\right) + \|h\left(x^k\right)\| - \|h\left(y^k\right)\|\right] \\ &\quad = \|h\left(x^k\right)\| - \|h\left(y^k\right)\| + \frac{1-r}{2}(\|h\left(y^k\right)\| - \|h\left(x^k\right)\|),\end{aligned} \tag{3.5}$$

we conclude that $0 < \theta_{k+1} \le \theta_k$ in all cases. Thus the sequence $\{\theta_k\}$ is positive and nonincreasing. Furthermore, by (2.17), (2.18), and (3.5), we have that

$$\Phi(y^k, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) \le \frac{1-r}{2}(\|h\left(y^k\right)\| - \|h(x^k)\|).$$

It remains to prove that the sequence $\{\theta_k\}$ is bounded away from zero. For this purpose, it suffices to show that θ_{k+1} is greater than a fixed positive number when it is defined by (2.18). In this case, we have that

$$\frac{1}{\theta_{k+1}} = \frac{2\left[f\left(y^k\right) - f\left(x^k\right) + \|h\left(x^k\right)\| - \|h\left(y^k\right)\|\right]}{(1+r)[\|h\left(x^k\right)\| - \|h\left(y^k\right)\|]}$$

$$\le \frac{2}{1+r}\left[\frac{|f\left(y^k\right) - f\left(x^k\right)|}{\|h\left(x^k\right)\| - \|h\left(y^k\right)\|} + 1\right].$$

Thus, by (2.7), (2.16), and Lemma 2.6,

$$\frac{1}{\theta_{k+1}} \le \frac{2}{1+r}\left[\frac{L_f\|y^k - x^k\|}{(1-r)\|h\left(x^k\right)\|} + 1\right] \le \frac{2}{1+r}\left[\frac{L_f\beta}{1-r} + 1\right].$$

This implies that the sequence $\{1/\theta_{k+1}\}$ is bounded. Therefore, the monotone sequence $\{\theta_k\}$ is bounded away from zero, as we wanted to prove. ∎

Assumption 3.2. Assumptions 2.1, 2.2, and 2.3 hold and, for every iteration k, the approximate solution of the quadratic programming problem (2.19) satisfies

$$\nabla f\left(y^k\right)^T\left(x - y^k\right) + \frac{1}{2}(x - y^k)^T H_k (x - y^k) + \mu\|x - y^k\|^2 \le 0. \tag{3.6}$$

Similarly to Assumption 2.1, Assumption 3.2 merely states that the objective function of the subproblem (2.19) must decrease regarding its value at y^k.

Lemma 3.3 below states that the criteria for stopping an optimality phase are necessarily satisfied after $O(1)$ iterations.

LEMMA 3.3. *Suppose that Assumption* 3.2 *holds. Then, defining $\bar\theta$ such as in Lemma* 3.1 *and*

$$N_{reg} \equiv \left\lfloor \log_2\left(\frac{M}{2} + L_{\nabla f} + \max\left\{\gamma, \frac{1-\bar\theta}{\bar\theta}L_{\nabla h}\right\}\right) - \log_2(\mu_{min})\right\rfloor + 1,$$

after at most N_{reg} updates (2.22), *conditions* (2.20) *and* (2.21) *are satisfied. Moreover, for*

$$\bar\mu \equiv \max\left\{\mu_{max}, 5M + 10L_{\nabla f} + 10\max\left\{\gamma, \frac{1-\bar\theta}{\bar\theta}L_{\nabla h}\right\}\right\},$$

we have that $\mu_k \le \bar\mu$ for all k.

Proof. Let $\bar\theta$ be defined as in Lemma 3.1. Define

$$\alpha \equiv \max\left\{\gamma, \frac{1-\bar\theta}{\bar\theta}L_{\nabla h}\right\}. \tag{3.7}$$

If $\mu \geq \frac{M}{2} + L_{\nabla f} + \alpha$, then, by (2.12) and (3.6),

$$\begin{aligned} f(x) &\leq f(y^k) + \nabla f(y^k)^T(x - y^k) + L_{\nabla f}\|x - y^k\|^2 \\ &\leq f(y^k) + \nabla f(y^k)^T(x - y^k) + \frac{1}{2}(x - y^k)^T H_k(x - y^k) \\ &\quad + \left(\frac{M}{2} + L_{\nabla f} + \alpha\right)\|x - y^k\|^2 - \alpha\|x - y^k\|^2 \\ &\leq f(y^k) + \nabla f\left(y^k\right)^T(x - y^k) + \frac{1}{2}(x - y^k)^T H_k(x - y^k) + \mu\|x - y^k\|^2 \\ &\quad - \alpha\|x - y^k\|^2 \\ &\leq f(y^k) - \alpha\|x - y^k\|^2. \end{aligned}$$

Since $\gamma \leq \alpha$, (2.20) holds. Moreover, by the definition of α,

$$f(x) - f(y^k) \leq -\frac{1 - \bar{\theta}}{\bar{\theta}} L_{\nabla h}\|x - y^k\|^2. \tag{3.8}$$

Let us show now that condition (2.21) also holds for $\mu \geq \frac{M}{2} + L_{\nabla f} + \alpha$. By the definition of Φ and (3.1) we have

$$\begin{aligned} \Phi(x, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) &= \Phi(x, \theta_{k+1}) - \Phi(y^k, \theta_{k+1}) + \Phi(y^k, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) \\ &\leq \theta_{k+1}\left[f(x) - f\left(y^k\right)\right] + (1 - \theta_{k+1})(\|h(x)\| - \|h\left(y^k\right)\|) \\ &\quad + \frac{1 - r}{2}(\|h\left(y^k\right)\| - \|h\left(x^k\right)\|). \end{aligned}$$

Then, by (3.8), (2.13), and the fact that $\nabla h\left(y^k\right)^T\left(x - y^k\right) = 0$ due to the constraints of the subproblem (2.19), we have

$$\begin{aligned} \Phi(x, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) &\leq -\theta_{k+1}\frac{1 - \bar{\theta}}{\bar{\theta}} L_{\nabla h}\|x - y^k\|^2 + (1 - \theta_{k+1})L_{\nabla h}\|x - y^k\|^2 \\ &\quad + \frac{1 - r}{2}(\|h\left(y^k\right)\| - \|h\left(x^k\right)\|). \end{aligned}$$

So, since $\theta_k \geq \bar{\theta}$,

$$\begin{aligned} \Phi(x, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) &\leq -\bar{\theta}\frac{1 - \bar{\theta}}{\bar{\theta}} L_{\nabla h}\|x - y^k\|^2 + (1 - \bar{\theta})L_{\nabla h}\|x - y^k\|^2 \\ &\quad + \frac{1 - r}{2}(\|h\left(y^k\right)\| - \|h\left(x^k\right)\|) \\ &\leq \frac{1 - r}{2}(\|h\left(y^k\right)\| - \|h\left(x^k\right)\|). \end{aligned}$$

Thus, after at most N_{reg} updates (2.22), both (2.20) and (2.21) are satisfied.

Moreover, by the boundedness of the initial μ and the update rule (2.22), the whole sequence $\{\mu_k\}$ is bounded by $\bar{\mu}$. ∎

In Theorem 3.4 we prove that the sum of the infeasibilities of all the iterates is bounded and that the same happens with the sum of all the squared increments computed in the optimality phases.

THEOREM 3.4. *Suppose that Assumption* 3.2 *holds. Let β and $\bar{\theta}$ be such as in Lemma* 2.6 *and Lemma* 3.1, *respectively. Then, defining*

$$\bar{h} \equiv \frac{2}{(1 - r)^2}\left(\left(\frac{(1 - r)^2}{2} + \rho_0 + \frac{1}{\bar{\theta}}\right)C_h + 2C_f\right)$$

and

$$C_d \equiv \frac{2C_f + \beta L_f \bar{h}}{\gamma} \tag{3.9}$$

for all $k \in \mathbb{N}$ and $j \leq k$, if the sequences $\{x^j\}$ and $\{y^j\}$ are generated by Algorithm 2.1, *we have that*

$$\sum_{j=1}^{k} \|h(x^j)\| \leq \bar{h}, \tag{3.10}$$

$$\sum_{j=1}^{k} \|h(y^j)\| \leq r\bar{h}, \tag{3.11}$$

$$f\left(x^k\right) \leq f(x^0) + L_f \beta \bar{h} - \gamma \sum_{j=0}^{k-1} \|x^{j+1} - y^j\|^2, \tag{3.12}$$

and

$$\sum_{j=0}^{k} \|x^{j+1} - y^j\|^2 \leq C_d. \tag{3.13}$$

Proof. By condition (2.21), for all j one has that

$$\Phi(x^{j+1}, \theta_{j+1}) \leq \Phi(x^j, \theta_{j+1}) + \frac{1-r}{2}(\|h(y^j)\| - \|h(x^j)\|).$$

Therefore, by (2.16),

$$\Phi(x^{j+1}, \theta_{j+1}) \leq \Phi(x^j, \theta_{j+1}) - \frac{(1-r)^2}{2}\|h(x^j)\|. \tag{3.14}$$

Let us define $\rho_j \equiv (1-\theta_j)/\theta_j$ for all j. By Lemma 3.1, we have that $\theta_j \geq \bar{\theta}$ for all j. This implies that $\rho_j \leq \frac{1}{\bar{\theta}} - 1$ for all j. Since $\{\rho_j\}$ is bounded and $\rho_0 > 0$, it follows that, for all k,

$$\sum_{j=0}^{k-1} (\rho_{j+1} - \rho_j) = \rho_k - \rho_0 < \frac{1}{\bar{\theta}}. \tag{3.15}$$

By (2.4), the fact that $\{\rho_j\}$ is nondecreasing, and (3.15), we have that

$$\sum_{j=0}^{k-1} (\rho_{j+1} - \rho_j)\|h(x^j)\| \leq \frac{C_h}{\bar{\theta}}. \tag{3.16}$$

Now, by (3.14), for all $j \leq k-1$,

$$f\left(x^{j+1}\right) + \frac{1-\theta_{j+1}}{\theta_{j+1}}\|h\left(x^{j+1}\right)\| \leq f(x^j) + \frac{1-\theta_{j+1}}{\theta_{j+1}}\|h(x^j)\| - \frac{(1-r)^2}{2\theta_{j+1}}\|h(x^j)\|.$$

Since $\theta_{j+1} < 1$, this implies that, for all $j \leq k-1$,

$$f\left(x^{j+1}\right) + \rho_{j+1}\|h\left(x^{j+1}\right)\| \leq f(x^j) + \rho_{j+1}\|h(x^j)\| - \frac{(1-r)^2}{2}\|h(x^j)\|.$$

Therefore, for all $j \leq k-1$,

$$f\left(x^{j+1}\right)+\rho_{j+1}\|h\left(x^{j+1}\right)\| \leq f(x^j)+\rho_j\|h(x^j)\|+(\rho_{j+1}-\rho_j)\|h(x^j)\|-\frac{(1-r)^2}{2}\|h(x^j)\|.$$

Thus, we have that

$$f(x^k)+\rho_k\|h\left(x^k\right)\| \leq f(x^0)+\rho_0\|h(x^0)\|+\sum_{j=0}^{k-1}(\rho_{j+1}-\rho_j)\|h(x^j)\|-\frac{(1-r)^2}{2}\sum_{j=0}^{k-1}\|h(x^j)\|.$$

Therefore, by (3.16),

$$f(x^k) + \rho_k\|h\left(x^k\right)\| \leq f(x^0) + \rho_0\|h(x^0)\| + \frac{C_h}{\bar{\theta}} - \frac{(1-r)^2}{2}\sum_{j=0}^{k-1}\|h(x^j)\|.$$

Thus,

$$\frac{(1-r)^2}{2}\sum_{j=0}^{k}\|h(x^j)\| \leq \frac{(1-r)^2}{2}\|h\left(x^k\right)\|-[f(x^k)+\rho_k\|h\left(x^k\right)\|]+f(x^0)+\rho_0\|h(x^0)\|+\frac{C_h}{\bar{\theta}}.$$

Since $\rho_k\|h(x^k)\| \geq 0$, by (2.3) and (2.4), it follows that

$$\sum_{j=0}^{k}\|h(x^j)\| \leq \frac{2}{(1-r)^2}\left(\left(\frac{(1-r)^2}{2} + \rho_0 + \frac{1}{\bar{\theta}}\right) C_h + 2C_f\right).$$

Thus, (3.10) is proved. Then, (3.11) follows from (2.16) and (3.10).

Now, by (2.20), for all $j \leq k-1$ we have

$$f\left(x^{j+1}\right)-f(x^j) \leq f\left(x^{j+1}\right)-f(y^j)+f(y^j)-f(x^j) \leq -\gamma\|x^{j+1}-y^j\|^2+f(y^j)-f(x^j).$$

Then, by (2.7) and (2.41),

$$f\left(x^{j+1}\right) - f(x^j) \leq -\gamma\|x^{j+1} - y^j\|^2 + \beta L_f\|h(x^j)\|$$

for all $j \leq k-1$. Therefore,

$$f\left(x^k\right) \leq f(x^0) - \gamma\sum_{j=0}^{k-1}\|x^{j+1} - y^j\|^2 + \beta L_f\sum_{j=0}^{k-1}\|h(x^j)\|.$$

Therefore, (3.12) follows from (3.10) and (3.13) follows from (2.3) and (3.10). ∎

From now on, we define, for all $k \in \mathbb{N}$,

$$D_{k+1} \equiv \left\{x \in \Omega \mid \nabla h\left(y^k\right)^T (x - y^k) = 0\right\}. \tag{3.17}$$

Assumption 3.5. Assumption 3.2 holds and, for every iteration k, the approximate solution of the quadratic programming problem (2.19) satisfies

$$\left\|P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(y^k\right)-H_k(x^{k+1}-y^k)-2\mu_k(x^{k+1}-y^k)\right)-x^{k+1}\right\| \leq \kappa\left\|x^{k+1}-y^k\right\|. \tag{3.18}$$

As well as Assumption 2.3, Assumption 3.5 can be satisfied by using any standard quadratic programming algorithm that generates a sequence of iterates converging to the solution of (2.19).

In the following lemma we prove that the sum of the optimality measures for all the iterates generated by Algorithm 2.1 is bounded.

LEMMA 3.6. *Suppose that Assumption* 3.5 *holds. Then, for every iteration k, if x^{k+1} is generated by Algorithm* 2.1*, we have*

$$\left\|P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(x^{k+1}\right)\right)-x^{k+1}\right\| \leq (L_{\nabla f}+2\bar{\mu}+M+\kappa)\|x^{k+1}-y^k\| \tag{3.19}$$

and

$$\left\|P_{D_{k+1}}\left(y^k-\nabla f\left(y^k\right)\right)-y^k\right\| \leq (2+2L_{\nabla f}+2\bar{\mu}+M+\kappa)\|x^{k+1}-y^k\|, \tag{3.20}$$

where $\bar{\mu}$ is defined in Lemma 3.3.

Moreover, for every iteration k we have that

$$\sum_{j=0}^{k}\left\|P_{D_{j+1}}\left(x^{j+1}-\nabla f\left(x^{j+1}\right)\right)-x^{j+1}\right\|^2 \leq (L_{\nabla f}+2\bar{\mu}+M+\kappa)^2 C_d \tag{3.21}$$

and

$$\sum_{j=0}^{k}\left\|P_{D_{j+1}}(y^j-\nabla f(y^j))-y^j\right\|^2 \leq (2+2L_{\nabla f}+2\bar{\mu}+M+\kappa)^2 C_d, \tag{3.22}$$

where C_d is given in (3.9).

Proof. By (3.18), the contraction property of projections, Lemma 3.3 (boundedness of μ_k), and (2.9) we have that

$$\begin{aligned}
&\left\|P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(x^{k+1}\right)\right)-x^{k+1}\right\| \\
&\quad \leq \left\|P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(x^{k+1}\right)\right)-P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(y^k\right)-H_k\left(x^{k+1}-y^k\right)\right.\right. \\
&\qquad \left.\left.-2\mu_k\left(x^{k+1}-y^k\right)\right)\right\| \\
&\qquad + \left\|P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(y^k\right)-H_k\left(x^{k+1}-y^k\right)-2\mu_k\left(x^{k+1}-y^k\right)\right)-x^{k+1}\right\| \\
&\quad \leq \left\|\nabla f\left(x^{k+1}\right)-\nabla f\left(y^k\right)\right\|+(2\mu_k+M)\|x^{k+1}-y^k\|+\kappa\|x^{k+1}-y^k\| \\
&\quad \leq (L_{\nabla f}+2\bar{\mu}+M+\kappa)\|x^{k+1}-y^k\|.
\end{aligned}$$

So, (3.19) is proved.

Let us now prove (3.20). By contraction of projections and (2.9) we have that

$$\begin{aligned}
&\left\|P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(x^{k+1}\right)\right)-x^{k+1}-\left[P_{D_{k+1}}\left(y^k-\nabla f\left(y^k\right)\right)-y^k\right]\right\| \\
&\quad \leq \left\|P_{D_{k+1}}\left(x^{k+1}-\nabla f\left(x^{k+1}\right)\right)-P_{D_{k+1}}\left(y^k-\nabla f\left(y^k\right)\right)\right\|+\|x^{k+1}-y^k\| \\
&\quad \leq \|\nabla f\left(x^{k+1}\right)-\nabla f\left(y^k\right)\|+2\|x^{k+1}-y^k\| \leq (L_{\nabla f}+2)\|x^{k+1}-y^k\|.
\end{aligned}$$

Then, (3.20) follows from (3.19).

Finally, (3.21) and (3.22) follow from (3.13), (3.19), and (3.20). ∎

Lemma 3.7 states that the number of iterates such that the infeasibility is bigger than a given ϵ_{feas} is, at most, proportional to $1/\epsilon_{feas}$.

LEMMA 3.7. *Suppose that Assumption* 3.5 *holds and* $\epsilon_{feas} > 0$. *Let* N_{infeas} *be the number of iterations of Algorithm* 2.1 *such that*

$$\|h\left(x^k\right)\| > \epsilon_{feas}.$$

Then, defining $\bar{h}$ *as in Theorem* 3.4*,*

$$N_{infeas} \leq \left\lfloor \frac{\bar{h}}{\epsilon_{feas}} \right\rfloor. \tag{3.23}$$

Moreover, the number of iterations such that $\|h(y^k)\| > \epsilon_{feas}$ *is not bigger than* $\lfloor \frac{r\bar{h}}{\epsilon_{feas}} \rfloor$.

Proof. By (3.10) we have that

$$\sum_{j=1}^{k} \|h(x^j)\| \leq \bar{h}.$$

Therefore, the number of iterations at which $\|h(x^j)\| > \epsilon_{feas}$ cannot be bigger than $\bar{h}/\epsilon_{feas}$. Since N_{infeas} is an integer, we have (3.23). Analogously, by (3.11), the number of iterations of Algorithm 2.1 such that $\|h\left(y^k\right)\| > \epsilon_{feas}$ cannot be bigger than $\frac{r\bar{h}}{\epsilon_{feas}}$. ∎

In the following lemma we prove that the number of iterations for which the optimality measure is bigger than an arbitrarily given ϵ_{opt} is, at most, proportional to $1/\epsilon_{opt}^2$.

LEMMA 3.8. *Suppose that Assumption* 3.5 *holds and* $\epsilon_{opt} > 0$. *Let* $\bar{\mu}$ *and* C_d *be as in Lemma* 3.3 *and Theorem* 3.4*, respectively, and let* N_{opt} *be the number of iterations* j *of Algorithm* 2.1 *such that*

$$\left\|P_{D_{j+1}}\left(x^{j+1} - \nabla f\left(x^{j+1}\right)\right) - x^{j+1}\right\| > \epsilon_{opt}. \tag{3.24}$$

Then,

$$N_{opt} \leq \left\lfloor (L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d \epsilon_{opt}^{-2} \right\rfloor. \tag{3.25}$$

Moreover, if N_{opty} *is the number of iterations such that*

$$\|P_{D_{j+1}}(y^j - \nabla f(y^j)) - y^j\| > \epsilon_{opt}, \tag{3.26}$$

we have that

$$N_{opty} \leq \left\lfloor (2 + 2L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d \epsilon_{opt}^{-2} \right\rfloor. \tag{3.27}$$

Proof. Assume that (3.24) holds for more than N_{opt} iterations. Then, there exists k such that among the k first iterations there are N_{opt} iterations that satisfy (3.24). Therefore,

$$\sum_{j=0}^{k} \left\|P_{D_{j+1}}\left(x^{j+1} - \nabla f\left(x^{j+1}\right)\right) - x^{j+1}\right\|^2 > N_{opt}\epsilon_{opt}^2. \tag{3.28}$$

Therefore, by (3.21),

$$N_{opt}\epsilon_{opt}^2 < (L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d.$$

Since N_{opt} is an integer, this implies (3.25). Analogously, (3.27) follows from (3.22).∎

The following theorem is the complexity result that says that the number of iterates with infeasibility bigger than ϵ_{feas} or projected gradient of f bigger than ϵ_{opt} cannot be bigger than a quantity proportional to $1/\epsilon_{feas} + 1/\epsilon_{opt}^2$.

THEOREM 3.9. *Suppose that Assumption* 3.5 *holds,* $\epsilon_{feas} > 0$*, and* $\epsilon_{opt} > 0$*. Let* $N_R, N_{regfeas}, N_{reg}, \bar{\mu}, \bar{h}$*, and* C_d *be defined as in Lemma* 2.4*, Lemma* 3.3*, and Theorem* 3.4*. Then the following hold:*

- *The number of iterations such that*

$$\|h(x^{j+1})\| > \epsilon_{feas} \text{ or } \left\|P_{D_{j+1}}\left(x^{j+1} - \nabla f\left(x^{j+1}\right)\right) - x^{j+1}\right\| > \epsilon_{opt} \tag{3.29}$$

  is bounded by

$$N_{it} \equiv \left\lfloor \bar{h}\epsilon_{feas}^{-1} \right\rfloor + \left\lfloor (L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d \epsilon_{opt}^{-2} \right\rfloor + 1.$$

  Moreover, the number of evaluations of f*,* ∇f*,* h*, and* ∇h *are bounded by* $N_{it}(1 + N_{reg}) + 1$*,* $N_{it} + 1$*,* $N_{it}(N_R N_{regfeas} + N_{reg}) + 1$*, and* $N_{it}N_R + 1$*, respectively.*

- *The number of iterations such that*

$$\|h(y^j)\| > \epsilon_{feas} \text{ or } \|P_{D_{j+1}}(y^j - \nabla f(y^j)) - y^j\| > \epsilon_{opt} \tag{3.30}$$

  is bounded by

$$N_{ity} \equiv \left\lfloor r\bar{h}\epsilon_{feas}^{-1} \right\rfloor + \left\lfloor (2 + 2L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d \epsilon_{opt}^{-2} \right\rfloor + 1.$$

  Moreover, the number of evaluations of f*,* ∇f*,* h*, and* ∇h *are bounded by* $N_{ity}(1 + N_{reg}) + 1$*,* N_{ity}*,* $N_{ity}(N_R N_{regfeas} + N_{reg}) + 1$*, and* $N_{ity}N_R + 1$*, respectively.*

Proof. Assume that k_1 iterations of Algorithm 2.1 are executed satisfying (3.29) and

$$k_1 > \bar{h}\epsilon_{feas}^{-1} + (L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d \epsilon_{opt}^{-2}.$$

Therefore, for all $j = 0, 1, \ldots, k_1$ the iterate x^{j+1} was defined and at least one of the following two statements held:

$$\|h(x^{j+1})\| > \epsilon_{feas} \tag{3.31}$$

or

$$\left\|P_{D_{j+1}}\left(x^{j+1} - \nabla f\left(x^{j+1}\right)\right) - x^{j+1}\right\| > \epsilon_{opt}. \tag{3.32}$$

By Lemma 3.7, (3.31) can occur at most in $\bar{h}\epsilon_{feas}^{-1}$ iterations. Moreover, by Lemma 3.8, we have that (3.32) can occur at most in $(L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d \epsilon_{opt}^{-2}$ iterations. Therefore, the number of iterations at which at least one of the statements (3.31) or (3.32) hold cannot exceeds $\lfloor \bar{h}\epsilon_{feas}^{-1} \rfloor + \lfloor (L_{\nabla f} + 2\bar{\mu} + M + \kappa)^2 C_d \epsilon_{opt}^{-2} \rfloor + 1$. The bound on the number of evaluations of h and ∇h follows from the number of iterations and Lemma 2.4, whereas the bound on the number of evaluations of f follows from the number of iterations and Lemma 3.3.

The second part of the thesis is proved in an entirely analogous way. ∎

Regarding the required precision, Theorem 3.9 shows that the complexity of the algorithm, with respect to evaluations of f, ∇f, h, and ∇h, is in $O(\epsilon_{feas}^{-1} + \epsilon_{opt}^{-2})$. In order to analyze the complexity obtained by explicitly involving the characteristics of the problem, let us define $C_0 \equiv \max\{C_f, C_h, 1\}$, $L_0 \equiv \max\{L_f, L_h, 1\}$, and $L_1 \equiv \max\{L_{\nabla f}, L_{\nabla h}, 1\}$. In this way we obtain that

- the number of evaluations of f is
$$O\left(C_0 L_0^4 \left(L_0 + L_1\right)^2 \log\left(L_0 + L_1\right) \left(\varepsilon_{feas}^{-1} + L_0^{12} L_1^2 \left(L_0 + L_1\right)^6 \varepsilon_{opt}^{-2}\right)\right),$$
- the number of evaluations of ∇f is
$$O\left(C_0 L_0^4 \left(L_0 + L_1\right)^2 \left(\varepsilon_{feas}^{-1} + L_0^{12} L_1^2 \left(L_0 + L_1\right)^6 \varepsilon_{opt}^{-2}\right)\right),$$
- the number of evaluations of h is
$$O\left(C_0 L_0^6 \left(L_0 + L_1\right)^4 \log\left(L_0 + L_1\right) \left(\varepsilon_{feas}^{-1} + L_0^{12} L_1^2 \left(L_0 + L_1\right)^6 \varepsilon_{opt}^{-2}\right)\right),$$
- the number of evaluations of ∇h is
$$O\left(C_0 L_0^6 \left(L_0 + L_1\right)^4 \left(\varepsilon_{feas}^{-1} + L_0^{12} L_1^2 \left(L_0 + L_1\right)^6 \varepsilon_{opt}^{-2}\right)\right).$$

The convergence theorem below is an immediate consequence of the complexity results.

THEOREM 3.10. *Suppose that Assumption* 3.5 *holds and that Algorithm* 2.1 *does not stop by restoration failure. Then,*

$$\lim_{k\to\infty} \|h(x^k)\| = 0, \lim_{k\to\infty} \|P_{D_k}(x^k - \nabla f(x^k)) - x^k\| = 0, \tag{3.33}$$

$$\lim_{k\to\infty} \|h\left(y^k\right)\| = 0, \text{ and } \lim_{k\to\infty} \|P_{D_{k+1}}(y^k - \nabla f\left(y^k\right)) - y^k\| = 0. \tag{3.34}$$

Proof. Assume that the algorithm generates infinitely many iterations and that $\|h(x^k)\|$ does not converge to zero. This implies that there exists $\varepsilon > 0$ and an infinite set of indices K such that $\|h(x^k)\| > \varepsilon$ for all $k \in K$. Therefore (3.29) occurs infinitely many times with $\epsilon_{feas} = \varepsilon$. This is impossible by Theorem 3.9. In a similar way, we prove that $\|P_{D_k}(x^k - \nabla f(x^k)) - x^k\|$ converges to zero. Analogously, (3.34) is proved using (3.30). ∎

From Theorems 3.9 and 3.10 we see that a reasonable stopping criterion for the practical application of Algorithm 2.1 is

$$\|h\left(y^k\right)\| \le \epsilon_{feas} \text{ and } \|P_{D_{k+1}}(y^k - \nabla f\left(y^k\right)) - y^k\| \le \epsilon_{opt}. \tag{3.35}$$

The condition (3.35) is the natural stopping criterion associated to the AGP optimality condition introduced in [43] and analyzed in [2] and [3].

The last theorem establishes that every limit point generated by Algorithm 2.1 satisfies the AGP optimality condition and, under AGP-regularity, satisfies the KKT optimality conditions.

THEOREM 3.11. *Suppose that Assumption* 3.5 *holds and that Algorithm* 2.1 *does not stop by restoration failure. Let $y^* \in \Omega$ be a limit point of the sequence $\{y^k\}$ generated by the algorithm. Then, y^* is feasible ($\|h(y^*)\| = 0$) and satisfies the AGP optimality condition. Finally, if y^* satisfies the AGP-regularity condition, the KKT conditions hold at y^*.*

Proof. The first statement is a consequence of continuity and (3.34). KKT fulfillment under AGP-regularity follows from [4]. ∎

4. Future research. IR methods have proved to be very efficient for solving problems in which there exist natural ways to restore feasibility, induced by the structure of constraints [1, 5, 11, 18, 19, 20, 31, 34, 39, 44]. Different general-purpose implementations have been given in [9, 13, 36]. The fact that, according to the results of the present paper, our version of the IR algorithm enjoys pleasant complexity properties is quite stimulating for the search of more competitive general-purpose computer implementations. As frequently occurs, theoretical developments inspire algorithmic ideas. Clearly, the properties (2.16), (2.41), (2.20), and (2.21) play a main role in the proofs of complexity and convergence. This suggests that new developments could be possible focusing on those properties with the aim of their fulfillment by means of a single quadratic programming subproblem, opening the possibility of obtaining sequential quadratic programming algorithms with satisfactory complexity properties.

Acknowledgment. The authors are indebted to the anonymous referees, whose comments helped us to improve the clarity of the paper.

REFERENCES

[1] R. Andreani, S. L. C. Castro, J. L. Chela, A. Friedlander, and S. A. Santos, *An inexact-restoration method for nonlinear bilevel programming problems*, Comput. Optim. Appl., 43 (2009), pp. 307–328, https://doi.org/10.1007/s10589-007-9147-4.

[2] R. Andreani, G. Haeser, and J. M. Martínez, *On sequential optimality conditions for smooth constrained optimization*, Optimization, 60 (2011), pp. 627–641, https://doi.org/10.1080/02331930903578700.

[3] R. Andreani, J. M. Martínez, A. Ramos, and P. J. S. Silva, *A cone-continuity constraint qualification and algorithmic consequences*, SIAM J. Optim., 26 (2016), pp. 96–110, https://doi.org/10.1137/15M1008488.

[4] R. Andreani, J. M. Martínez, A. Ramos, and P. J. S. Silva, *Strict constraint qualifications and sequential optimality conditions for constrained optimization*, Math. Oper. Res., 43 (2018), pp. 693–717, https://doi.org/10.1287/moor.2017.0879.

[5] R. Andreani, V. Ramírez, S. A. Santos, and L. D. Secchin, *Bilevel optimization with a multiobjective problem in the lower level*, Numer. Algorithms, 81 (2019), pp. 915–946, https://doi.org/10.1007/s11075-018-0576-1.

[6] N. Banihashemi and C. Y. Kaya, *Inexact restoration for Euler discretization of box constrained optimal controls*, J. Optim. Theory Appl., 156 (2013), pp. 726–760, https://doi.org/10.1007/s10957-012-0140-4.

[7] E. Bergou, Y. Diouane, and S. Gratton, *A line-search algorithm inspired by the adaptive cubic regularization framework and a complexity analysis*, J. Optim. Theory Appl., 178 (2018), pp. 885–913, http://doi.org/10.1007/s10957-018-1341-2.

[8] E. Bergou, Y. Diouane, and S. Gratton, *On the use of the energy norm in trust-region and adaptive cubic regularization subproblems*, Comput. Optim. Appl., 68 (2017), pp. 533–554, https://doi.org/10.1007/s10589-017-9929-2.

[9] E. G. Birgin, L. F. Bueno, and J. M. Martínez, *Assessing the reliability of general-purpose inexact restoration methods*, J. Comput. Appl. Math., 282 (2015), pp. 1–16, https://doi.org/10.1016/j.cam.2014.12.031.

[10] E. G. Birgin, J. L. Gardenghi, J. M. Martínez, S. A. Santos, and P. L. Toint, *Evaluation complexity for nonlinear constrained optimization using unscaled KKT conditions and high-order models*, SIAM J. Optim., 26 (2016), pp. 951–967, https://doi.org/10.1137/15M1031631.

[11] E. G. Birgin, N. Krejić, and J. M. Martínez, *On the employment of inexact restoration for the minimization of functions whose evaluation is subject to errors*, Math. Comp., 87 (2018), pp. 1307–1326, https://doi.org/10.1090/mcom/3246.

[12] E. G. Birgin, R. Lobato, and J. M. Martínez, *Constrained optimization with integer and continuous variables using inexact restoration and projected gradients*, Bull. Comput. Appl. Math., 4 (2016), pp. 55–70, https://bcamjournal.com/index.php/path/article/download/30/25.

[13] E. G. Birgin and J. M. Martínez, *Local convergence of an inexact-restoration method and numerical experiments*, J. Optim. Theory Appl., 127 (2005), pp. 229–247, https://doi.org/10.1007/s10957-005-6537-6.

[14] E. G. Birgin and J. M. Martínez, *The use of quadratic regularization with a cubic descent condition for unconstrained optimization*, SIAM J. Optim., 27 (2017), pp. 1049–1074, https://doi.org/10.1137/16M110280X.

[15] E. G. Birgin and J. M. Martínez, *On regularization and active-set methods with complexity for constrained optimization*, SIAM J. Optim., 28 (2018), pp. 1367–1395, https://doi.org/10.1137/17M1127107.

[16] E. G. Birgin, J. M. Martínez, L. Martínez, and G. B. Rocha, *Sparse projected-gradient method as a linear-scaling low-memory alternative to diagonalization in self-consistent field electronic structure calculations*, J. Chem. Theory Comput., 9 (2013), pp. 1043–1051, https://doi.org/10.1021/ct3009683.

[17] E. G. Birgin, J. M. Martínez, and M. Raydan, *Nonmonotone spectral projected gradient methods on convex sets*, SIAM J. Optim., 10 (2000), pp. 1196–1211, https://doi.org/10.1137/S1052623497330963.

[18] L. F. Bueno, A. Friedlander, J. M. Martínez, and F. N. Sobral, *Inexact-restoration method for derivative-free optimization with smooth constraints*, SIAM J. Optim., 23 (2013), pp. 1189–1213, https://doi.org/10.1137/110856253.

[19] L. F. Bueno, G. Haeser, and J. M. Martínez, *A flexible inexact-restoration method for constrained optimization*, J. Optim. Theory Appl., 165 (2015), pp. 188–208, https://doi.org/10.1007/s10957-014-0572-0.

[20] L. F. Bueno, G. Haeser, and J. M. Martínez, *An inexact restoration approach to optimization problems with multiobjective constraints under weighted-sum scalarization*, Optim. Lett., 10 (2016), pp. 1315–1325, https://doi.org/10.1007/s11590-015-0928-x.

[21] L. F. Bueno, G. Haeser, and L. Santos, *Towards an efficient augmented Lagrangian method for convex quadratic programming*, Comput. Optim. Appl., (2019), https://doi.org/10.1007/s10589-019-00161-2.

[22] C. Cartis, N. I. M. Gould, and P. L. Toint, *On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization*, SIAM J. Optim., 20 (2010), pp. 2833–2852, https://doi.org/10.1137/090774100.

[23] C. Cartis, N. I. M. Gould, and P. L. Toint, *Adaptive cubic regularization methods for unconstrained optimization. Part* I*: Motivation, convergence and numerical results*, Math. Program., 127 (2011), pp. 245–295, https://doi.org/10.1007/s10107-009-0286-5.

[24] C. Cartis, N. I. M. Gould, and P. L. Toint, *Adaptive cubic regularization methods for unconstrained optimization. Part* II*: Worst-case function and derivative complexity*, Math. Program., 130 (2011), pp. 295–319, https://doi.org/10.1007/s10107-009-0337-y.

[25] C. Cartis, N. I. M. Gould, and P. L. Toint, *On the evaluation complexity of composite function minimization with applications to nonconvex nonlinear programming*, SIAM J. Optim., 21 (2011), pp. 1721–1739, https://doi.org/10.1137/11082381X.

[26] C. Cartis, N. I. M. Gould, and P. L. Toint, *Complexity bounds for second-order optimality in unconstrained optimization*, J. Complexity, 28 (2012), pp. 28–93, https://doi.org/10.1016/j.jco.2011.06.001.

[27] C. Cartis, N. I. M. Gould, and P. L. Toint, *On the complexity of finding first-order critical points in constrained nonlinear optimization*, Math. Program., 144 (2014), pp. 93–106, https://doi.org/10.1007/s10107-012-0617-9.

[28] C. Cartis, N. I. M. Gould, and P. L. Toint, *Universal regularization methods: Varying the power, the smoothness and the accuracy*, SIAM J. Optim., 29 (2019), pp. 595–615, https://doi.org/10.1137/16M1106316.

[29] F. E. Curtis, D. P. Robinson, and M. Samadi, *A trust-region algorithm with a worst-case iteration complexity of* $O(\varepsilon^{-3/2})$, Math. Program., 162 (2017), pp. 1–32, https://doi.org/10.1007/s10107-016-1026-2.

[30] Z. Dostál, *Optimal Quadratic Programming Algorithms: With Applications to Variational Inequalities*, Springer Optim. Appl. 23, Springer, New York, 2009, https://doi.org/10.1007/b138610.

[31] P. S. Ferreira, E. W. Karas, M. Sachine, and F. N. Sobral, *Global convergence of a derivative-free inexact restoration filter algorithm for nonlinear programming*, Optimization, 66 (2017), pp. 271–292, https://doi.org/10.1080/02331934.2016.1263629.

[32] A. Fischer and A. Friedlander, *A new line search inexact restoration approach for nonlinear programming*, Comput. Optim. Appl., 46 (2010), pp. 333–346, https://doi.org/10.1007/s10589-009-9267-0.

[33] R. Fletcher, *Augmented lagrangians, box constrained QP and extensions*, IMA J. Numer. Anal., 37 (2017), pp. 1635–1656, https://doi.org/10.1093/imanum/drx002.

[34] J. B. Francisco, J. M. Martínez, L. Martínez, and F. I. Pisnitchenko, *Inexact restoration method for minimization problems arising in electronic structure calculations*, Comput. Optim. Appl., 50 (2011), pp. 555–590, https://doi.org/10.1007/s10589-010-9318-6.

[35] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Emerald Group, Bradford, UK, 1982.

[36] M. A. Gomes-Ruggiero, J. M. Martínez, and S. A. Santos, *Spectral projected gradient method with inexact restoration for minimization with nonconvex constraints*, SIAM J. Sci. Comput., 31 (2009), pp. 1628–1652, https://doi.org/10.1137/070707828.

[37] C. C. Gonzaga, E. W. Karas, and M. Vanti, *A globally convergent filter method for nonlinear programming*, SIAM J. Optim., 14 (2004), pp. 646–669, https://doi.org/10.1137/S1052623401399320.

[38] J. Y. Grapiglia, G. N.and Yuan and Y. X. Yuan, *On the convergence and worst-case complexity of trust-region and regularization methods for unconstrained optimization*, Math. Program., 152 (2015), pp. 491–520, https://doi.org/10.1007/s10107-014-0794-9.

[39] E. W. Karas, E. A. Pilotta, and A. A. Ribeiro, *Numerical comparison of merit function with filter criterion in inexact restoration algorithms using hard-spheres problems*, Comput. Optim. Appl., 44 (2009), pp. 427–441, https://doi.org/10.1007/s10589-007-9162-5.

[40] J. M. Martínez, *Inexact restoration method with lagrangian tangent decrease and new merit function for nonlinear programming*, J. Optim. Theory Appl., 111 (2001), pp. 39–58, https://doi.org/10.1023/A:1017567113614.

[41] J. M. Martínez, *On high-order model regularization for constrained optimization*, SIAM J. Optim., 27 (2017), pp. 2447–2458, https://doi.org/10.1137/17M1115472.

[42] J. M. Martínez and E. A. Pilotta, *Inexact restoration algorithms for constrained optimization*, J. Optim. Theory Appl., 104 (2000), pp. 135–163, https://doi.org/10.1023/A:1004632923654.

[43] J. M. Martínez and B. F. Svaiter, *A practical optimality condition without constraint qualifications for nonlinear programming*, J. Optim. Theory Appl., 118 (2003), pp. 117–133, https://doi.org/10.1023/A:1024791525441.

[44] L. M. Mito, *O problema de cobertura via geometria algébrica convexa*, Master's thesis, Department of Applied Mathematics, University of São Paulo, 2018.

[45] Y. Nesterov and B. T. Polyak, *Cubic regularization of Newton's method and its global performance*, Math. Program., 108 (2006), pp. 177–205, https://doi.org/10.1007/s10107-006-0706-8.

[46] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, 2000, https://doi.org/10.1007/978-0-387-40065-5.