

Inexact Restoration method for Derivative-Free Optimization with smooth constraints ^{*}

L. F. Bueno [†] A. Friedlander [‡] J. M. Martínez [§]
F. N. C. Sobral [¶]

November 21st, 2011.

Abstract

A new method is introduced for solving constrained optimization problems in which the derivatives of the constraints are available but the derivatives of the objective function are not. The method is based on the Inexact Restoration framework, by means of which each iteration is divided in two phases. In the first phase one considers only the constraints, in order to improve feasibility. In the second phase one minimizes a suitable objective function subject to a linear approximation of the constraints. The second phase must be solved using derivative-free methods. An algorithm introduced recently by Kolda, Lewis, and Torczon for linearly constrained derivative-free optimization is employed for this purpose. Under usual assumptions, convergence to stationary points is proved. A computer implementation is described and numerical experiments are presented.

^{*}This work was supported by PRONEX-CNPq/FAPERJ Grant E-26/171.164/2003 - APQ1, FAPESP Grants 03/09169-6, 06/53768-0, 07/06663-0 and 08/00468-4 and CNPq.

[†]Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, State University of Campinas, Campinas, SP, Brazil. E-mail: lfelipebueno@gmail.com

[‡]Professor, Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, State University of Campinas, Campinas, SP, Brazil. E-mail: friedlan@ime.unicamp.br

[§]Professor, Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, State University of Campinas, Campinas, SP, Brazil. E-mail: martinez@ime.unicamp.br

[¶]Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, State University of Campinas, Campinas, SP, Brazil. E-mail: fsobral@ime.unicamp.br

Key words: Inexact Restoration, Derivative-Free Optimization, Global Convergence, Numerical Experiments.

AMS Subject Classification: 65K05, 65Y20, 90C56.

1 Introduction

In this paper we address nonlinear programming problems in which the derivatives of the objective function are not available, whereas the derivatives of the constraints are [24]. Several methods take advantage of the simplicity of boxes and linear constraints [10, 18, 20, 21, 32] or need sufficient thickness of the feasible set [7, 8]. Here, we have in mind more general, perhaps highly nonlinear, constraints. We believe that approaches in which one evaluates function and constraints at the same points (for example, the Augmented Lagrangian approaches of [11, 16, 22, 23]) are not fully satisfactory because, sometimes, the presence of topologically complex constraints causes the necessity of performing many evaluations. In these cases, the intrinsically expensive objective function could be unnecessarily computed, increasing the overall computational cost.

In this context, methods that separately evaluate constraints and objective function seem to be useful. The key point is that one should not evaluate the objective function when the partial (possibly difficult) goal is to improve feasibility. Inexact Restoration (IR) methods are well suited for this purpose.

The main iteration of modern Inexact Restoration methods for smooth constrained optimization [9, 12, 13, 15, 26, 27] proceeds in two phases. In the first (Restoration) phase, feasibility is improved without evaluations of the objective function at all. In the second (Optimization) phase, one improves the objective function (or a Lagrangian, or an Augmented Lagrangian with moderate penalty parameter) on a tangent approximation to the constraints. The resulting trial point is accepted, or not, according to trust-region [26, 27], line search [12] or filter criteria [13, 15]. If the trial point is rejected, a new trial point is taken closer to the restored point obtained at the end of the first phase. Convergence to KKT points under regularity assumptions is usually obtained and superlinear convergence can be proved for local versions of the methods [9, 15]. A method that resembles Inexact Restoration ideas in the global optimization field was introduced in [28].

The Inexact Restoration approach is useful for the problems that we have

in mind for two reasons. On one hand, difficulties associated with the fulfillment of the constraints are transferred to the first phase of the iterations, in which the objective function (whose derivatives are not available) plays no role. Therefore, we can take advantage of well established smooth optimization tools in this phase. On the other hand, the Optimization phase needs derivative-free minimization with linear constraints, a problem for which there exist very efficient algorithms, in particular, the GSS method introduced by Kolda, Lewis, and Torczon [17, 19].

Although this paper deals with optimization problems in which constraint derivatives are available but function derivatives are not, it is worthwhile to mention that the Inexact Restoration philosophy applies to more general situations. For instance, consider the case in which function derivatives are available too, but its evaluation is very expensive. In this case it is better to deal with infeasibility in a way that independes of evaluating the objective function. Moreover, if some constraint evaluations are expensive and others are not, it is sensible to include the expensive constraints in the objective function in an Augmented Lagrangian context with constrained subproblems [1]. In this way, the Inexact Restoration method turns out to be appropriate for solving the constrained Augmented Lagrangian subproblems.

This paper is organized as follows. In Section 2 we describe the derivative-free Inexact Restoration algorithm. In Section 3 we analyze the algorithm's convergence. In Section 4 we discuss implementation details and the numerical experiments are shown in Section 5. Finally, Section 6 is devoted to conclusions and lines for future research.

Notation

The symbol $\|\cdot\|$ will denote the Euclidean norm on \mathbb{R}^n .

We denote $\mathbb{N} = \{0, 1, 2, \dots\}$.

2 Algorithm

The problem considered in this paper is:

$$\text{Minimize } f(x) \quad \text{subject to } h(x) = 0 \quad \text{and } x \in \Omega, \quad (1)$$

where $f : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$, $h : \Omega \rightarrow \mathbb{R}^m$, and Ω is a bounded polytope given by:

$$\Omega = \{x \in \mathbb{R}^n \mid a_i^T x \leq b_i, i = 1, \dots, p\}. \quad (2)$$

We assume that h is smooth and its first derivatives are available. The set Ω will be generally defined by unrelaxable constraints in the sense of [8].

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

$$\Phi(x, \theta) = \theta f(x) + (1 - \theta)\|h(x)\|. \quad (3)$$

The main algorithm considered in this paper is defined as follows.

Algorithm 1

Let $r \in [0, 1)$, $\beta > 0$, $\bar{\mu} \geq \gamma > 0$ and $1 < \alpha_l \leq \alpha_u$ be algorithmic parameters. Choose $x^0 \in \Omega$, $\theta_0 \in (0, 1)$. Set $k \leftarrow 0$.

Step 1 Restoration

Compute $y^k \in \Omega$ such that

$$\|h(y^k)\| \leq r\|h(x^k)\| \quad (4)$$

and

$$\|y^k - x^k\| \leq \beta\|h(x^k)\|. \quad (5)$$

Step 2 Penalty parameter

If

$$\Phi(y^k, \theta_k) - \Phi(x^k, \theta_k) \leq \frac{1}{2}(1 - r)(\|h(y^k)\| - \|h(x^k)\|) \quad (6)$$

set $\theta_{k+1} = \theta_k$ and go to Step 3.

Else, compute

$$\theta_{k+1} = \frac{(1 + r)(\|h(x^k)\| - \|h(y^k)\|)}{2[f(y^k) - f(x^k) + \|h(x^k)\| - \|h(y^k)\|]}. \quad (7)$$

Step 3 Optimization and Regularization

Choose $\mu \in [\gamma, \bar{\mu}]$.

Step 3.1 Tangent set minimization

Find an approximate solution $d \in \mathbb{R}^n$ of the following subproblem:

$$\begin{aligned} &\text{Minimize} && f(y^k + d) + \mu\|d\|^2 \\ &\text{subject to} && \nabla h(y^k)^T d = 0 \\ &&& y^k + d \in \Omega. \end{aligned} \quad (8)$$

Step 3.2 Descent conditions

Test the conditions

$$f(y^k + d) \leq f(y^k) - \gamma\|d\|^2 \quad (9)$$

and

$$\Phi(y^k + d, \theta_{k+1}) \leq \Phi(x^k, \theta_{k+1}) + \frac{1}{2}(1 - r)(\|h(y^k)\| - \|h(x^k)\|). \quad (10)$$

If both (9) and (10) are fulfilled, define $d^k = d$, $\mu_k = \mu$, $x^{k+1} = y^k + d^k$, update $k \leftarrow k + 1$, and go to Step 1. Else, update

$$\mu \in [\alpha_l \mu, \alpha_u \mu] \quad (11)$$

and go to Step 3.1.

Remarks

1. The Restoration phase, given by the conditions (4) and (5), has been defined in Algorithm 1 as in the Inexact Restoration papers [9, 26, 27]. A slightly more general form for the condition (5) was considered in [12]. However, the condition in [12] involves the computation of the objective function, whereas in the present paper, we wish to emphasize that no evaluations of f are necessary at the first phase of the method. The procedure for updating the penalty parameter is similar to the one employed in [12]. However, here we try to decrease more slowly the penalty parameter than in [12] and we state explicitly the biggest possible value of this parameter.
2. We have been ambiguous with respect to the precise description of Step 3.1. The approximate minimizer of (8) will be obtained, in practice, by means a well-established method for minimization with linear constraints that does not employ derivatives of f at all. The practical effectiveness of Algorithm 1 relies strongly on the efficiency of this method. In the way presented above, Algorithm 1 defines a sequence of alternated restoration and linearly constrained minimization steps. The objective function at the minimization steps is the objective function of the original problem plus a regularization term. This formulation is adequate for the employment of derivative-free minimization on the tangent subspace. We note that formulations based on line searches [12] are not appropriate since, in those cases, one needs to guarantee that descent is possible along the obtained direction without explicit gradient information.

3 Convergence

Let us state first some assumptions that will be used along this section. The first assumption is that the Restoration step is well defined for all $k = 0, 1, 2, \dots$. Clearly, this assumption ceases to be satisfied when, for example, the feasible set is empty. Sufficient regularity conditions on h that guarantee that Assumption A1 holds have been given in [12, 26].

Assumption A1

For all $k = 0, 1, 2, \dots$ it is possible to compute $y^k \in \Omega$ such that (4) and (5) are fulfilled.

The following assumption merely states that f satisfies a Lipschitz condition.

Assumption A2

There exists $L_f > 0$ such that

$$|f(x) - f(y)| \leq L_f \|x - y\| \quad (12)$$

for all $x, y \in \Omega$.

Lemma 3.1 *Assume that A1 and A2 are fulfilled. Then, for all $k \in \mathbb{N}$, Steps 1 and 2 of Algorithm 1 are well defined. Moreover, the sequence $\{\theta_k\}$ is non-increasing, the inequality*

$$\Phi(y^k, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) \leq \frac{1}{2}(1-r)(\|h(y^k)\| - \|h(x^k)\|) \quad (13)$$

is fulfilled for all k , and there exists $\bar{\theta} > 0$ such that

$$\theta_k \downarrow \bar{\theta}. \quad (14)$$

Proof. Step 1 is well defined by Assumption A1.

Using Assumption A1 and (4), we obtain that

$$\|h(x^k)\| - \|h(y^k)\| + \frac{1-r}{2}(\|h(y^k)\| - \|h(x^k)\|) = \frac{1+r}{2}(\|h(x^k)\| - \|h(y^k)\|) \geq 0. \quad (15)$$

If $\|h(y^k)\| - \|h(x^k)\| = 0$, then, by (4), we have that $\|h(y^k)\| = \|h(x^k)\| = 0$. Therefore, by (5), $y^k = x^k$. So, $\Phi(x^k, \theta_k) = \Phi(y^k, \theta_k)$. Thus, (6) holds in this case and, consequently, $\theta_{k+1} = \theta_k$.

Therefore, by (15), it remains to consider only the case in which $\|h(y^k)\| < \|h(x^k)\|$. So,

$$\|h(x^k)\| - \|h(y^k)\| + \frac{1-r}{2}(\|h(y^k)\| - \|h(x^k)\|) = \frac{1+r}{2}(\|h(x^k)\| - \|h(y^k)\|) > 0. \quad (16)$$

By direct calculations, the inequality (6) is equivalent to

$$\begin{aligned} & \theta_k [f(y^k) - f(x^k) + \|h(x^k)\| - \|h(y^k)\|] \\ & \leq \|h(x^k)\| - \|h(y^k)\| + \frac{1-r}{2}(\|h(y^k)\| - \|h(x^k)\|). \end{aligned} \quad (17)$$

Thus, by (16), the requirement (6) is fulfilled whenever $f(y^k) - f(x^k) + \|h(x^k)\| - \|h(y^k)\| \leq 0$. In this case, the algorithm also chooses $\theta_{k+1} = \theta_k$.

Therefore, we only need to consider the case in which

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

In this case, if (6) does not hold, then, by (17), we have that

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

for all $\theta \geq \theta_k$. Now, the choice (7) obviously implies that

$$\begin{aligned} & \theta_{k+1} [f(y^k) - f(x^k) + \|h(x^k)\| - \|h(y^k)\|] \\ & = \|h(x^k)\| - \|h(y^k)\| + \frac{1-r}{2}(\|h(y^k)\| - \|h(x^k)\|) \end{aligned} \quad (18)$$

Since both the numerator and the denominator of (18) are positive, it turns out that $\theta_{k+1} < \theta_k$ is well defined. Moreover, by (7) and (17),

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

and the sequence $\{\theta_k\}$ is non-increasing. It only remains to prove that this sequence 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 (7) and both numerator and denominator are positive. By (4), we have:

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

Thus, by (12), (4), and (5),

$$\frac{1}{\theta_{k+1}} \leq \frac{2}{1+r} \left[\frac{L_f \beta \|h(x^k)\|}{(1-r)\|h(x^k)\|} + 1 \right] = \frac{2}{1+r} \left[\frac{L_f \beta}{1-r} + 1 \right] \quad (19)$$

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

The formula (19) exposes the worst possible size of the penalty parameter. By (19) we see that, taking, say, $r \leq 1/2$, extreme penalty parameters may appear only if the product $L_f \beta$ is very big. The constant L_f measures the variations of f whereas β is related to the possibility of having an error bound associated with the constraints. It turns out that the penalty parameter θ_k may decrease excessively only if f exhibits large local variations and the constraints are ill-conditioned.

From now on, we employ an additional smoothness condition on h . Essentially, we are going to assume that ∇h satisfies a Lipschitz condition. This is a sufficient condition for the fulfillment of Assumption A3 below.

Assumption A3 There exists $L_h > 0$ such that, for all $d \in \mathbb{R}^n$, $y \in \Omega$ such that $y + d \in \Omega$ and $\nabla h(y)^T d = 0$, one has:

$$\|h(y + d)\| \leq \|h(y)\| + L_h \|d\|^2. \quad (20)$$

For the next result presented in this section, the only condition we are going to assume regarding the approximate solution d of (8) is the fulfillment of a single descent criterion with respect to the restored point y^k :

$$f(y^k + d) + \mu \|d\|^2 \leq f(y^k). \quad (21)$$

Clearly, as in [12], this condition is very easy to obtain, since even the null direction $d = 0$ satisfies (21).

Assumption A4

For all $k \in \mathbb{N}$ the vector $d \in \mathbb{R}^n$ computed at Step 3.1 is a feasible point of the subproblem (8) and satisfies (21).

Lemma 3.2 *Assume that A1, A2, A3, and A4 are fulfilled. Let $x^k \in \Omega$ be an iterate computed by Algorithm 1. Then, after a finite number of updates (11), conditions (9) and (10) are satisfied. (Thus, Step 3 of Algorithm 1 is well defined.) Moreover, there exists $\mu_{bound} > 0$ such that $\mu_k \leq \mu_{bound}$ for*

all $k \in \mathbb{N}$.

Proof. By (11) and the initialization of μ at Step 3, we have that $\mu \geq \gamma$ at all the subproblems (8). Therefore, by (21),

$$f(y^k + d) + \gamma \|d\|^2 \leq f(y^k + d) + \mu \|d\|^2 \leq f(y^k).$$

Therefore, condition (9) always takes place. Let us show now that condition (10) also holds for μ large enough. By Lemma 3.1, there exists $\bar{\theta} > 0$ such that

$$\theta_k \geq \bar{\theta} \text{ for all } k. \quad (22)$$

We are going to prove now that, if

$$\mu \geq \frac{1 - \bar{\theta}}{\bar{\theta}} L_h, \quad (23)$$

then inequality (10) is also fulfilled.

By (23) we have that

$$(1 - \bar{\theta})L_h - \bar{\theta}\mu \leq 0. \quad (24)$$

By Assumptions A3-A4, (13) and the definition of Φ , we have:

$$\begin{aligned} & \Phi(y^k + d, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) \\ &= \Phi(y^k + d, \theta_{k+1}) - \Phi(y^k, \theta_{k+1}) + \Phi(y^k, \theta_{k+1}) - \Phi(x^k, \theta_{k+1}) \\ &\leq \theta_{k+1}[f(y^k + d) - f(y^k)] + (1 - \theta_{k+1})(\|h(y^k + d)\| - \|h(y^k)\|) + \frac{1 - r}{2}(\|h(y^k)\| - \|h(x^k)\|) \\ &\leq -\theta_{k+1}\mu \|d\|^2 + (1 - \theta_{k+1})L_h \|d\|^2 + \frac{1 - r}{2}(\|h(y^k)\| - \|h(x^k)\|). \end{aligned} \quad (25)$$

But, by (22) and (24),

$$-\theta_{k+1}\mu \|d\|^2 + (1 - \theta_{k+1})L_h \|d\|^2 \leq -\bar{\theta}\mu \|d\|^2 + (1 - \bar{\theta})L_h \|d\|^2 \leq 0. \quad (26)$$

By (25) and (26), it follows that (10) holds whenever (23) takes place. Thus, after a finite number of updates (11), both (9) and (10) are satisfied. By the boundedness of the initial μ , the update rule (11), and (23), the whole sequence $\{\mu_k\}$ is bounded independently of k . \square

Theorem 3.1 *Assume that $\{x^k\}$ and $\{y^k\}$ are generated by Algorithm 1 and Assumptions A1-A4 hold. Then,*

$$\lim_{k \rightarrow \infty} \|h(x^k)\| = \lim_{k \rightarrow \infty} \|h(y^k)\| = \lim_{k \rightarrow \infty} \|d^k\| = 0. \quad (27)$$

Moreover, $\{x^k\}$ and $\{y^k\}$ admit the same limit points and every limit point is feasible.

Proof. By condition (10), for all $k \in \mathbb{N}$ one has that

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

Therefore, by (4),

$$\Phi(x^{k+1}, \theta_{k+1}) \leq \Phi(x^k, \theta_{k+1}) - \frac{(1-r)^2}{2} \|h(x^k)\|. \quad (28)$$

Let us define $\rho_k = (1 - \theta_k)/\theta_k$ for all $k \in \mathbb{N}$. By Lemma 3.1 there exists $\bar{\theta} > 0$ such that $\theta_k \geq \bar{\theta}$ for all $k \in \mathbb{N}$. This implies that $\rho_k \leq 1/\bar{\theta} - 1$ for all $k \in \mathbb{N}$. Since $\{\rho_k\}$ is bounded and nondecreasing it follows that

$$\sum_{k=0}^{\infty} (\rho_{k+1} - \rho_k) = \lim_{k \rightarrow \infty} \rho_{k+1} - \rho_0 < \infty. \quad (29)$$

By compactness, the sequence $\{\|h(x^k)\|\}$ is bounded. Therefore, by (29), there exists $c > 0$ such that

$$\sum_{k=0}^{\infty} (\rho_{k+1} - \rho_k) \|h(x^k)\| \leq c < \infty. \quad (30)$$

Now, by (28),

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

Since $\theta_{k+1} < 1$, this implies that, for all $k \in \mathbb{N}$,

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

Therefore, for all $k \in \mathbb{N}$,

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

Thus, for all $k \in \mathbb{N}$, we have:

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

Therefore, by (30),

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

Thus,

$$\frac{(1-r)^2}{2} \sum_{j=0}^{k-1} \|h(x^j)\| \leq -[f(x^k) + \rho_k \|h(x^k)\|] + f(x^0) + \rho_0 \|h(x^0)\| + c$$

Since $\{\rho_k\}$ is bounded, by the continuity of f and h and the compactness of Ω , it follows that the series $\sum_{k=0}^{\infty} \|h(x^k)\|$ is convergent. Therefore,

$$\lim_{k \rightarrow \infty} \|h(x^k)\| = 0.$$

Thus, by (5),

$$\lim_{k \rightarrow \infty} \|y^k - x^k\| = 0$$

and the sequences $\{x^k\}$ and $\{y^k\}$ admit the same limit points.

Now, by (9), for all $k \in \mathbb{N}$ we have:

$$f(x^{k+1}) - f(x^k) \leq f(x^{k+1}) - f(y^k) + f(y^k) - f(x^k) \leq -\gamma \|d^k\|^2 + f(y^k) - f(x^k).$$

Then, by (5) and (12),

$$f(x^{k+1}) - f(x^k) \leq -\gamma \|d^k\|^2 + L_f \beta \|h(x^k)\|$$

for all $k \in \mathbb{N}$. Therefore,

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

Since the series $\sum_{k=0}^{\infty} \|h(x^k)\|$ is convergent, there exists $\bar{c} > 0$ such that, for all $k \in \mathbb{N}$,

$$f(x^k) \leq f(x^0) + \bar{c} - \gamma \sum_{\ell=0}^{k-1} \|d^\ell\|^2.$$

Thus, since f is bounded below on Ω , the series $\sum_{k=0}^{\infty} \|d^k\|^2$ is convergent and $\|d^k\|$ tends to zero. \square

Observe that, up to now, the existence of derivatives of f have not been assumed at all. Well-definiteness of the algorithm and feasibility of limit points have been guaranteed employing a Lipschitz condition for f but without additional smoothness assumptions. This feature opens the possibility of using efficient fully derivative-free algorithms for linearly-constrained optimization for solving (8).

Assumption A5 imposes that the approximate solution d in (8) should be obtained by means of the algorithm GSS for derivative-free minimization with linear constraints [17, 19].

Assumption A5

For all $k \in \mathbb{N}$, the direction d at Step 3.1 of Algorithm 1, is obtained by means of Algorithm GSS of [19] using a stopping criterion parameter $\Delta_k > 0$ such that $\lim_{k \rightarrow \infty} \Delta_k = 0$.

Observe that the employment of GSS to solve the subproblem (8) is compatible with Assumption A4, since GSS is a descent method and the direction $d = 0$ is feasible.

For deciding to stop the execution, GSS employs a fixed tolerance $\eta > 0$ such that the only linear constraints that are considered in the testing procedure are those that are “almost active” at $y^k + d$ with tolerance η . This includes, of course, the active constraints at $y^k + d$. Recalling the definition of Ω (2), it is useful to define:

$$A(k, \eta) = \{i \in \{1, \dots, p\} \mid 0 \geq a_i^T(y^k + d) - b_i \geq -\eta\},$$

and

$$\Omega(k, \eta) = \{x \in \mathbb{R}^n \mid a_i^T x \leq b_i \forall i \in A(k, \eta)\}.$$

$A(k, \eta)$ is the set of indices of the linear constraints that are almost active at the point $y^k + d \in \Omega$ and $\Omega(k, \eta)$ is the polytope Ω excluding the almost active constraints. Clearly, $\Omega \subset \Omega(k, \eta)$ and, if $\eta = \infty$, one has that $\Omega = \Omega(k, \eta)$. Moreover, since $y^k \in \Omega$, for d small enough we have that

$$A(k, \eta) = \{i \in \{1, \dots, m\} \mid 0 \geq a_i^T y^k - b_i \geq -2\eta\}.$$

Finally, we define:

$$D_k = \{x \in \Omega(k, \eta) \mid \nabla h(y^k)^T(x - y^k) = 0\}.$$

Assumption A6

From now on we will assume that the derivatives of f , although not used

in our algorithm at all, exist and are Lipschitz continuous, with constant L_g , for all x in a neighborhood of Ω .

We emphasize that Assumption A6 is necessary to prove convergence to optimal points in the context of the employment of GSS as subproblem solver. By Theorem 6.3 of [19], there exists $c_k > 0$ such that, if we stop the GSS algorithm with tolerance Δ_k , we have that

$$\|P_{D_k}(x^{k+1} - \nabla f(x^{k+1}) - 2\mu_k(x^{k+1} - y^k)) - x^{k+1}\| \leq c_k \Delta_k. \quad (31)$$

Moreover, the constant c_k only depends on algorithmic parameters and problem-dependent magnitudes that are naturally bounded. Therefore, we may assume that $c_k \leq c > 0$ for all k .

This property is enough to prove the main convergence result for Algorithm 1 (with GSS).

Theorem 3.2 *Assume that the sequence $\{x^k\}$ is generated by Algorithm 1 and that Assumptions A1–A6 hold. Let x^* be a limit point of $\{x^k\}$. Then, x^* is feasible and satisfies the Approximate Gradient Projection Condition (AGP) in the sense of [2, 6, 29]. Moreover, if the Constant Positive Linear Dependence Condition (CPLD) holds at x^* , then x^* fulfills the KKT conditions.*

Proof. The feasibility of x^* has been proved in Theorem 3.1.

By (31) and the contraction property of projections, we have that:

$$\begin{aligned} & \|P_{D_k}(y^k - \nabla f(y^k)) - y^k\| \\ \leq & \|P_{D_k}(x^{k+1} - \nabla f(x^{k+1})) - x^{k+1}\| + \|P_{D_k}(x^{k+1} - \nabla f(x^{k+1})) - P_{D_k}(y^k - \nabla f(y^k)) + y^k - x^{k+1}\| \\ & \leq \|P_{D_k}(x^{k+1} - \nabla f(x^{k+1})) - x^{k+1}\| + (L_g + 2)\|x^{k+1} - y^k\| \\ \leq & \|P_{D_k}(x^{k+1} - \nabla f(x^{k+1})) - P_{D_k}(x^{k+1} - \nabla f(x^{k+1}) - 2\mu_k(x^{k+1} - y^k))\| \\ & + \|P_{D_k}(x^{k+1} - \nabla f(x^{k+1}) - 2\mu_k(x^{k+1} - y^k)) - x^{k+1}\| + (L_g + 2)\|d^k\| \\ \leq & \|2\mu_k(x^{k+1} - y^k)\| + \|P_{D_k}(x^{k+1} - \nabla f(x^{k+1}) - 2\mu_k(x^{k+1} - y^k)) - x^{k+1}\| + (L_g + 2)\|d^k\| \\ & \leq (2\mu_k + L_g + 2)\|d^k\| + c\Delta_k. \end{aligned}$$

Therefore, by Lemma 3.2 (boundedness of μ_k), Theorem 3.1 ($\|d^k\| \rightarrow 0$) and Assumption A5, we have that

$$\lim_{k \rightarrow \infty} \|P_{D_k}(y^k - \nabla f(y^k)) - y^k\| = 0.$$

This implies that the limit point x^* satisfies the AGP optimality condition. Therefore, the desired result follows from [2]. \square

Remarks

If the CPLD constraint qualification [5, 33] holds at x^* and a linear combination of a subset of gradients of active constraints (with non-negative coefficients corresponding to inequalities) is null, the same subset of gradients is linearly dependent for all x in a neighborhood of x^* . This constraint qualification is weaker than the Mangasarian-Fromovitz condition, which means that results ensuring KKT under CPLD are stronger than results in which KKT is guaranteed subject to the fulfillment of Mangasarian-Fromovitz. This attractive feature has already motivated the introduction of Augmented Lagrangian derivative-free methods in which optimality in the limit is associated with CPLD [11, 23].

It is worthwhile to mention that the conclusions of Theorem 3.2 can be obtained with constraint qualifications weaker than CPLD. This is due to the fact that sequential optimality conditions as AGP imply KKT even under the weak conditions defined, for example, in [3, 4].

4 Implementation

One of the most attractive characteristic of Inexact Restoration methods relies on the freedom to choose the methods for both Restoration and Optimization phases. This allows one to use suitable methods which exploit the structure of the problem. In Algorithm 1, although Assumption A5 requires the use of a specific algorithm for solving subproblem (8), any derivative-free method for linear constraints whose limit points satisfy (31) can be used.

4.1 Restoration phase

In the Restoration phase, we approximately solve the following feasibility problem:

$$\text{Minimize } \|y - x^k\| \text{ subject to } h(y) = 0 \text{ and } y \in \Omega. \quad (32)$$

As the derivatives of the constraints are available, any nonlinear programming algorithm can be used for solving, approximately, problem (32). We use Algencan [1], an Augmented Lagrangian algorithm, with its default parameters. The optimality tolerance employed was 0.1 and the feasibility

tolerance was dynamically reduced as a function of Δ_k in the following way:

$$\varepsilon_k^{\text{feas}} = \max\{10^{-8}/\sqrt{n}, \min\{\varepsilon_{k-1}^{\text{feas}}, \|h(x^k)\|\}\Delta_k\},$$

where $\varepsilon_0^{\text{feas}} = 0.01$.

In practice, we do not test conditions (4) and (5) at every iteration of the algorithm. The reason is that, when we restore according to (32) with the specifications above, (4) and (5) are usually satisfied for problem-dependent values of r and β . Therefore, testing these conditions with user-given parameters does not lead to practical efficiency.

4.2 Optimization phase

To approximately solve (8) in the Optimization phase, we use the GSS algorithm [17, 19], which was implemented in the software HOPSPACK [31].

The Optimization phase of Algorithm 1 may be expensive because it involves objective function evaluations. While conditions (9) and (10) are not satisfied, a linearly constrained derivative-free problem has to be solved for increasing values of μ . According to Lemma 3.2, sufficiently large values for μ produce directions that satisfy these conditions, at the cost of taking small step sizes. On the other hand, small values of μ generate large step sizes, but subproblem (8) may be solved several times until conditions (9) and (10) are fulfilled.

Inspired in Lemma 3.2 we use the following rule to set good values for μ . In the first iteration ($k = 0$), μ is initialized by the formula:

$$\mu = 1.01 \min\{\max\{\gamma, \mu'\}, 10^{40}\gamma\},$$

where

$$\mu' = \begin{cases} \frac{1 - \theta_1 \|h(x^0)\| - \|h(y^0)\|}{\theta_1 \|x^0 - y^0\|^2} & , \text{ if } y^0 \neq x^0 \\ \gamma & , \text{ if } y^0 = x^0 \end{cases}.$$

After solving subproblem (8), we compute

$$\mu'' = \begin{cases} \frac{1 - \theta_{k+1} \|h(y^k + d)\| - \|h(y^k)\|}{\theta_{k+1} \|d\|^2} & , \text{ if } d \neq 0 \\ \mu & , \text{ if } d = 0 \end{cases} \quad \text{and} \quad (33)$$

$$\mu' = 1.01 \min\{\max\{\gamma, \mu''\}, 10^{10}\mu, 10^{40}\gamma\},$$

where μ is the regularization parameter that has just been used in the objective function of (8) and d is the approximate solution found by GSS. If

either of the conditions (9) or (10) is not satisfied, then we update μ by the formula

$$\mu = \max\{\mu', 10\mu\} \quad (34)$$

and we solve (8) again, otherwise μ' is stored and used in the next iteration for the initialization of μ . At subsequent iterations ($k \geq 1$), μ is initialized by $\mu = \mu'$, where μ' was calculated in the previous iteration, and is updated by rules (33)-(34).

These implemented rules for initialization and updating of μ are compatible with the rules described in Algorithm 1, with the algorithmic parameters $\bar{\mu}$, α_l and α_u implicitly determined. Other implementation details used in this phase are given below:

1. We set $\gamma = 2^{-20}$.
2. The initial point for subproblem (8) is $d = 0$. If the approximate solution d does not satisfy one of the conditions (9) or (10), we use this solution as initial point for solving (8) again.

The advantages of using such initial points are that both are feasible solutions and, using the update given by (33), if the starting point is declared an approximate solution by GSS, then (9) and (10) are automatically satisfied.

3. We set $\Delta_0 = 0.5$ and

$$\Delta_{k+1} = \max\{10^{-16}, \min\{\delta_k, 0.1 \max\{\|h(y^k + d^k)\|, \|d^k\|\}\}\}, \quad (35)$$

where $\delta_k = 0.5/(1.1)^k$. Using (35) we require more effort in the GSS algorithm when both the infeasibility and the step size are decreasing or when a sufficient large number of iterations has been taken.

4. We declare that Algorithm 1 successfully converged if a point $y^k + d^k$ is found such that

$$\|h(y^k + d^k)\| \leq 10^{-8}, \quad \|d^k\| \leq 10^{-3} \quad \text{and} \quad \Delta_k \leq 10^{-3}.$$

5 Numerical Experiments

Algorithm 1 was implemented in C/C++ language as an extension of the software HOPSPACK. HOPSPACK [31] is an Augmented Lagrangian algorithm developed in C++ to deal with general derivative-free problems

described in [16]. At each step it uses GSS to minimize the Augmented Lagrangian function subject to linear and bound constraints. No derivatives are required at all. For using Algencan in the Restoration phase an interface between C++ and Fortran 77 was used. The code of Algencan and its interface with C++ are available at [34]. To generate the executable we used `gfortran-4.2`, `g++-4.2` and `cmake`. The last one was necessary for compiling HOPSPACK. The BLAS and LAPACK libraries (version 3.2.1) were also necessary.

The Hock and Schittkowski [14] test set is widely used to test derivative-free algorithms on moderate-size problems. A subset of 105 constrained problems, from a set of 116 general nonlinear problems, was selected to test our algorithm. The box-constrained problems (9 problems) and the “badly coded” problems (2 problems) were ignored. Inequality constraints were converted to equality constraints by the addition of nonnegative slack variables.

We compared the number of function evaluations used in our method with the Augmented Lagrangian method HOPSPACK [31], which implements the algorithm described in [16]. Table 1 shows the problems considered in this test. The column `Prob.` corresponds to the problem number from [14]. The columns `Var.`, `Ineq.` and `Eq.` are the number of variables, inequality constraints and equality constraints, respectively. The values between parentheses represent the number of linear constraints.

We used the performance and data profiles discussed in [30] for comparing the algorithms. The performance measure was the number of objective function evaluations. A problem was considered solved by an algorithm if the given solution \bar{x} was such that

$$\|h(\bar{x})\| \leq 10^{-8} \quad \text{and} \quad \frac{|f(\bar{x}) - f_L|}{\max\{1, |f(\bar{x})|, |f_L|\}} \leq 0.1, \quad (36)$$

where f_L is the lowest objective function value found among the compared algorithms. The results are displayed in Figure 1. Algorithmic executions were stopped after 20 minutes of CPU time.

Observe that, when we allow the algorithms to perform 1000 objective function evaluations, the IR method solves 20 problems that are not solved by HOPSPACK. This difference increases to 30 problems when 10^6 function evaluations are allowed. The performance profile in Figure 1(b) show that the Inexact Restoration proposal is able to save objective function evaluations, with respect to a well-establish solver, as expected.

For completeness, in Table 2 we show the complete numerical results for the IR algorithm and HOPSPACK. In the first 7 columns, `P` is the problem’s

Prob.	Var.	Ineq.	Eq.	Prob.	Var.	Ineq.	Eq.	Prob.	Var.	Ineq.	Eq.
6	2	0	1	43	4	3	0	80	5	0	3
7	2	0	1	44	4	6(6)	0	81	5	0	3
8	2	0	2	46	5	0	2	83	5	6	0
9	2	0	1(1)	47	5	0	3	84	5	6	0
10	2	1	0	48	5	0	2(2)	86	5	10(10)	0
11	2	1	0	49	5	0	2(2)	87	6	0	4
12	2	1	0	50	5	0	3(3)	88	2	1	0
13	2	1	0	51	5	0	3(3)	89	3	1	0
14	2	1	1(1)	52	5	0	3(3)	90	4	1	0
15	2	2	0	53	5	0	3(3)	91	5	1	0
16	2	2	0	54	6	0	1(1)	92	6	1	0
17	2	2	0	55	6	0	6(6)	93	6	2	0
18	2	2	0	56	7	0	4	95	6	4	0
19	2	2	0	57	2	1	0	96	6	4	0
20	2	3	0	58	2	3	0	97	6	4	0
21	2	1(1)	0	59	2	3	0	98	6	4	0
22	2	2(1)	0	60	3	0	1	99	7	0	2
23	2	5(1)	0	61	3	0	2	100	7	4	0
24	2	3(3)	0	62	3	0	1(1)	101	7	6	0
26	3	0	1	63	3	0	2(1)	102	7	6	0
27	3	0	1	64	3	1	0	103	7	6	0
28	3	0	1(1)	65	3	1	0	104	8	6	0
29	3	1	0	66	3	2	0	105	8	1 (1)	0
30	3	1	0	68	4	0	2	106	8	6 (3)	0
31	3	1	0	69	4	0	2	107	9	0	6
32	3	1	1(1)	70	4	1	0	108	9	13	0
33	3	2	0	71	4	1	1	109	9	4 (2)	6
34	3	2	0	72	4	2	0	111	10	0	3
35	3	1(1)	0	73	4	2(1)	1(1)	112	10	0	3(3)
36	3	1(1)	0	74	4	2(2)	3	113	10	8 (3)	0
37	3	2(2)	0	75	4	2(2)	3	114	10	8 (4)	3(1)
39	4	0	2	76	4	3(3)	0	116	13	15 (5)	0
40	4	0	3	77	5	0	2	117	15	5	0
41	4	0	1(1)	78	5	0	3	118	15	29(29)	0
42	4	0	2	79	5	0	3	119	16	0	8(8)

Table 1: Description of the test problems.

number and, for each algorithm, f is the functional value, $\|h\|$ is the norm of infeasibility and $\#FE$ is the number of objective function evaluations. The last 7 columns have analogous meanings as the first ones.

6 Final Remarks

We presented an Inexact Restoration approach for constrained derivative-free optimization. The derivatives of the constraints were supposed to be available. Global convergence to stationary points was proved and an implementation of the proposed algorithm was tested against the derivative-free Augmented Lagrangian algorithm HOPSPACK [31].

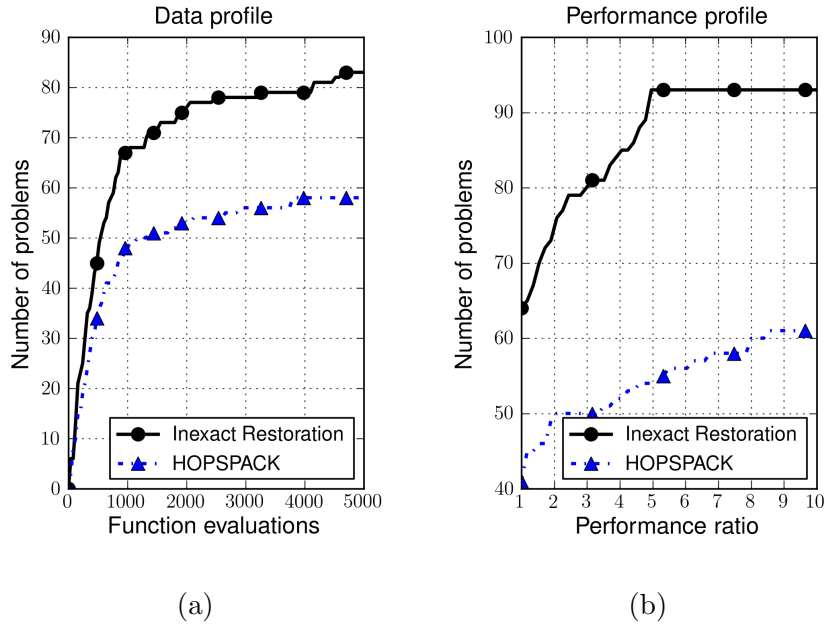


Figure 1: Data (a) and performance (b) profiles for the comparison between the Inexact Restoration algorithm (Algorithm 1) and HOPSPACK.

The availability of constraint derivatives is an important assumption since, in the Optimization phase of the algorithm, we solve a linearly constrained derivative-free optimization problem defined by the Jacobian of the constraints. On the other hand, objective function derivatives are not used at all. Moreover, the gradient of f plays a modest role in the convergence theory since it is only associated to the convergence properties of the internal GSS solver [19]. Clearly, the efficiency of this solver is essential to support the effectiveness of the overall algorithm. Nevertheless, the independence of important properties of the IR algorithm with respect to the smoothness of the objective function suggests that variations of the algorithm in which smoothness of f would not be assumed at all should be useful. Further research may be expected on this subject.

Inexact Restoration methods for constrained optimization are closely related to the block-generalized Brown-Brent methods for solving nonlinear systems of equations defined in [25]. This relation was emphasized in [9], in connection with local convergence proofs. In Brown-Brent methods one obtains local quadratic convergence without necessarily using derivatives of

the components of the system and employing a different number of evaluations for each component. If the derivatives of all the components, except one, are available, it is sensible to divide the system into two blocks, the first containing the derivable components and the last one with only one function, for which the gradient is not available. With such decomposition the Brown-Brent method is quadratically convergent, we take full advantage of the available derivatives, and we need to evaluate the complicate component only twice per iteration. This procedure suggests plausible ideas for considering a natural generalization of problem (1). We have in mind the situation in which only the derivatives of some constraints are available, whereas the remaining constraints must be addressed using derivative-free tools. This will be the subject of forthcoming research.

P	Inexact Restoration			HOPSPACK			P	Inexact Restoration			HOPSPACK		
	f	h	#FE	f	h	#FE		f	h	#FE	f	h	#FE
6	5.57E-10	4E-09	366	* 4.84E+00	* 5E-03	* 151	62	-2.63E+04	2E-16	848	-2.63E+04	1E-16	233
7	-1.73E+00	2E-09	153	* 6.93E-01	* 0E+00	* 325	63	9.62E+02	4E-09	171	9.63E+02	* 5E-04	* 317
8	-1.00E+00	2E-09	4	* -1.00E+00	* 9E-03	* 187	64	6.30E+03	3E-09	1025	6.30E+03	0E+00	6253
9	-5.00E-01	4E-15	117	* -5.00E-01	* 4E-15	* 26	65	9.54E-01	5E-09	1840	1.01E+00	0E+00	379
10	-1.00E+00	7E-09	260	* -8.65E-01	* 0E+00	* 357	66	5.18E-01	4E-09	406	5.33E-01	0E+00	566
11	-8.50E+00	8E-09	126	* -8.48E+00	0E+00	523	68	-9.20E-01	9E-09	10424	* -8.44E-01	* 2E-04	* 1316
12	-3.00E+01	7E-09	68E+00	* -3.00E+01	0E+00	342	69	-9.57E+02	2E-09	4130	* -9.57E+02	1E-04	* 2471
13	1.09E+00	6E-09	52326	1.00E+00	0E+00	803	70	* 2.69E-01	* 1E-09	* 5563	7.74E-03	0E+00	3766
14	1.39E+00	3E-09	20	* 1.39E+00	* 4E-06	* 202	71	1.70E+01	9E-09	4586	* 1.70E+01	* 5E-07	* 1939
15	3.06E+00	7E-14	82	3.06E+00	0E+00	451	72	7.28E+02	3E-09	58181	* 7.21E+02	* 2E-04	* 9606
16	2.58E-01	7E-10	546	2.50E-01	0E+00	600	73	2.99E+01	1E-09	211	3.02E+01	0E+00	223
17	1.00E+00	8E-10	118	1.00E+00	0E+00	612	74	5.13E+03	6E-10	392	* 5.14E+03	* 1E+00	* 46145
18	5.00E+00	9E-10	39217	* 1.07E+01	* 0E+00	* 263	75	5.17E+03	1E-09	139	* 5.23E+03	* 2E-01	* 22678
19	-6.96E+03	3E-09	138	* -7.47E+03	* 3E-01	* 1370	76	-4.68E+00	6E-10	505	-4.68E+00	0E+00	403
20	4.02E+01	8E-10	88	3.82E+01	0E+00	393	77	2.42E-01	2E-09	790	* 4.68E+00	* 2E-04	* 1904
21	-1.00E+02	7E-15	153	-1.00E+02	0E+00	32	78	-2.92E+00	8E-09	566	* -2.89E+00	* 4E-03	* 869
22	1.00E+00	9E-09	35	1.00E+00	0E+00	276	79	7.88E-02	6E-09	362	* 2.42E-01	1E-03	* 1054
23	2.00E+00	7E-09	37	2.00E+00	0E+00	466	80	5.39E-02	2E-09	658	* 1.00E+00	* 5E-03	* 557
24	-1.00E+00	1E-09	132	-1.00E+00	4E-16	27	81	5.39E-02	6E-09	770	* 1.00E+00	* 5E-03	* 557
26	1.58E-07	9E-09	11112	* 2.12E+01	* 0E+00	* 585	83	-3.07E+04	2E-09	450	-3.07E+04	0E+00	1708
27	4.00E+00	7E-09	4135	* 4.00E+00	* 3E-06	* 1358	84	-5.28E+01	4E-10	338	-5.28E+01	0E+00	2842
28	2.05E-23	9E-16	510	7.70E-08	3E-15	264	86	-3.23E+01	3E-10	437	-3.23E+01	0E+00	485
29	-2.26E+01	6E-10	647	-2.25E+01	0E+00	327	87	8.85E+03	1E-09	860	* 9.33E+03	* 7E-01	* 16244
30	1.00E+00	4E-09	790	1.00E+00	0E+00	55	88	1.37E+00	8E-12	105	1.37E+00	* 7E-07	* 1274
31	6.00E+00	8E-10	526	6.00E+00	0E+00	921	89	1.36E+00	4E-09	510	* 1.36E+00	* 3E-06	* 1945
32	1.00E+00	1E-09	86	1.00E+00	2E-16	51	90	1.37E+00	4E-09	1304	1.37E+00	0E+00	2655
33	* -4.00E+00	* 5E-09	* 54	* -4.59E+00	0E+00	381	91	1.36E+00	5E-09	1305	* 1.36E+00	* 8E-07	* 4180
34	-8.34E-01	2E-09	228	* -2.28E-01	* 0E+00	* 582	92	1.37E+00	3E-09	3235	* 1.36E+00	4E-06	* 4004
35	1.11E-01	7E-10	289	1.11E-01	0E+00	340	93	1.48E+02	2E-09	2006	1.37E+02	0E+00	129
36	-3.30E+03	1E-12	273	-3.30E+03	0E+00	60	95	1.86E-02	2E-09	741	1.71E-02	0E+00	156
37	-3.46E+03	1E-09	404	-3.46E+03	5E-14	102	96	1.56E-02	5E-10	668	1.71E-02	0E+00	156
39	-9.89E-01	4E-09	125	-1.00E+00	0E+00	830	97	4.07E+00	5E-10	5654	4.12E+00	0E+00	145
40	-2.50E-01	2E-09	133	* -2.51E-01	* 0E+00	* 897	98	3.14E+00	8E-10	1540	* 4.12E+00	0E+00	145
41	1.93E+00	1E-09	430	1.93E+00	2E-15	292	99	-8.31E+08	2E-09	4	* -7.46E+08	* 3E+03	* 729
42	1.39E+01	2E-09	831	1.40E+01	0E+00	779	100	* 5.18E+03	* 7E-10	* 1314	6.84E+02	0E+00	873
43	-4.40E+01	8E-10	1878	-4.40E+01	0E+00	1134	101	1.81E+03	1E-09	267	1.82E+03	0E+00	14563
44	-1.30E+01	2E-09	277	* -1.30E+01	0E+00	57	102	* 1.13E+03	* 9E-10	* 57697	9.21E+02	0E+00	15147
46	1.42E-06	1E-09	1485	* 3.34E+00	* 2E-16	* 777	103	* 1.55E+03	* 3E-12	* 22	5.44E+02	0E+00	14583
47	1.15E-08	9E-10	289	* 1.25E+01	* 0E+00	* 901	104	3.95E+00	6E-09	28794	3.95E+00	0E+00	9844
48	1.07E-24	2E-15	861	1.12E-06	2E-15	497	105	1.14E+03	1E-09	25324	1.14E+03	0E+00	10580
49	1.34E-07	2E-14	20308	1.43E-04	7E-15	1002	106	7.05E+03	9E-09	68430	* 1.14E+04	* 0E+00	* 30424
50	9.25E-27	6E-14	620	5.29E-07	6E-14	290	107	5.06E+03	4E-09	265	* 5.06E+03	* 3E-03	* 7232
51	1.94E-27	9E-16	507	1.25E-06	9E-16	142	108	-5.00E-01	1E-09	139823	-5.00E-01	0E+00	99
52	5.33E+00	3E-09	307	* 5.33E+00	* 8E-06	* 311	109	5.39E+03	6E-10	62792	* 5.00E-01	0E+00	99
53	4.09E+00	3E-09	308	4.09E+00	1E-14	216	110	-4.28E+01	2E-09	2470	* 5.50E+03	* 2E-02	* 57551
54	-1.54E-01	5E-10	447	* -1.54E-01	* -	* -	111	-4.78E+01	3E-09	10907	* -	* -	* -
55	6.67E+00	3E-09	18	* 6.00E+00	* 1E+00	* 0	113	2.43E+01	4E-10	4469	-4.78E+01	8E-16	730
56	-1.06E-06	3E-10	21267	-1.00E+00	9E-16	2075	114	-1.76E+03	2E-09	108605	2.54E+01	0E+00	1887
57	* 3.06E-02	* 1E-09	* 287	3.06E-02	0E+00	74	116	9.76E+01	5E-09	48155	* 5.00E+01	* 8E-02	* 9131
58	3.19E+00	1E-09	102	3.19E+00	0E+00	817	117	3.23E+01	5E-10	7951	* 5.40E+01	0E+00	* 7190
59	-7.80E+00	4E-09	929	* -6.75E+00	* 0E+00	* 340	118	6.65E+02	1E-09	1987	6.65E+02	2E-14	3762
60	3.26E-02	8E-10	596	* 5.47E-02	* 2E-03	* 465	119	2.45E+02	2E-09	1356	2.45E+02	5E-15	944
61	-1.44E+02	3E-09	182	-1.43E+02	0E+00	621							

Table 2: Numerical results for the Inexact Restoration algorithm and HOPSPACK in 105 test problems. The entries marked with * correspond to non-solved problems.

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] R. Andreani, G. Haeser and J. M. Martínez, On sequential optimality conditions for smooth constrained optimization. *Optimization* 60, pp. 627–641 (2011).
- [3] R. Andreani, G. Haeser, M. L. Schuverdt and P. J. S. Silva, A relaxed constant positive linear dependence constraint qualification and applications, *Mathematical Programming*, (2011). DOI: 10.1007/s10107-011-0456-0.
- [4] R. Andreani, G. Haeser, M. L. Schuverdt and P. J. S. Silva, Two new constraint qualifications and applications. *Optimization Online* (2011).
- [5] R. Andreani, J. M. Martínez and M. L. Schuverdt, On the relation between the Constant Positive Linear Dependence condition and quasinormality constraint qualification, *Journal of Optimization Theory and Applications* 125, pp. 473–485 (2005).
- [6] R. Andreani, J. M. Martínez and B. F. Svaiter, A new sequential optimality condition for constrained optimization and algorithmic consequences, *SIAM Journal on Optimization*, 20(6), pp. 3533–3554 (2011).
- [7] C. Audet and J. E. Dennis Jr., Mesh adaptive direct search algorithms for constrained optimization, *SIAM Journal on Optimization* 17, pp. 188–217 (2006).
- [8] C. Audet and J. E. Dennis Jr., A progressive barrier for derivative-free nonlinear programming, *SIAM Journal on Optimization* 20, pp. 445–472 (2009).
- [9] E. G. Birgin and J. M. Martínez, Local convergence of an Inexact-Restoration method and numerical experiments, *Journal of Optimization Theory and Applications* 127, pp. 229–247 (2005).
- [10] A. R. Conn, K. Scheinberg and L. N. Vicente, *Introduction to Derivative-Free Optimization*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2009.

- [11] M. A. Diniz-Ehrhardt, J. M. Martínez and L. G. Pedroso, Derivative-free methods for nonlinear programming with general lower-level constraints, *Computational & Applied Mathematics* 30, pp. 19–52 (2011).
- [12] A. Fischer and A. Friedlander, A new line search inexact restoration approach for nonlinear programming, *Computational Optimization and Applications* 46, pp. 333–346 (2010).
- [13] C. C. Gonzaga, E. Karas and M. Vanti, A globally convergent filter method for Nonlinear Programming, *SIAM Journal on Optimization* 14, pp. 646–669 (2003).
- [14] W. Hock and K. Schittkowski, Test examples for nonlinear programming codes, *Lecture Notes in Economics and Mathematical Systems* 187, Springer-Verlag, Berlin, Heidelberg, New York, 1981.
- [15] E. W. Karas, C. C. Gonzaga, and A. A. Ribeiro, Local convergence of filter methods for equality constrained non-linear programming, *Optimization* 59, pp. 1153–1171 (2010).
- [16] T. G. Kolda and J. D. Griffin, *Nonlinearly-Constrained Optimization Using Asynchronous Parallel Generating Set Search*. Technical Report, SAND2007–3257, Sandia National Laboratories, 2007.
- [17] T. G. Kolda, R. M. Lewis and V. Torczon, Optimization by direct search: new perspectives on some classical and modern methods, *SIAM Review* 45, pp. 385–482 (2003).
- [18] T. G. Kolda, R. M. Lewis and V. Torczon, *A generating set direct search augmented Lagrangian algorithm for optimization with a combination of general and linear constraints*. Technical Report, SAND2006–5315, Sandia National Laboratories, 2006.
- [19] T. G. Kolda, R. M. Lewis and V. Torczon, Stationarity results for generating set search for linearly constrained optimization, *SIAM Journal on Optimization* 17, pp. 943–968 (2006).
- [20] R. M. Lewis and V. Torczon, Pattern search algorithms for bound constrained minimization, *SIAM Journal on Optimization* 9, pp. 1082–1099 (1999).
- [21] R. M. Lewis and V. Torczon, Pattern search algorithms for linearly constrained minimization, *SIAM Journal on Optimization* 10, pp. 917–941 (2000).

- [22] R. M. Lewis and V. Torczon, A globally convergent Augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds, *SIAM Journal on Optimization* 12, pp. 1075–1089 (2002).
- [23] R. M. Lewis and V. Torczon, *A direct search approach to nonlinear programming problems using an Augmented Lagrangian method with explicit treatment of linear constraints*. Technical Report WM-CS-2010-01, College of William & Mary, Department of Computer Sciences, 2010.
- [24] S. Lucidi, M. Sciandrone and P. Tseng, Objective derivative-free methods for constrained optimization, *Mathematical Programming* 92, pp. 37–59 (2002).
- [25] J. M. Martínez, Generalization of the methods of Brent and Brown for solving nonlinear simultaneous equations, *SIAM Journal on Numerical Analysis* 16, pp. 434–448 (1979).
- [26] J. M. Martínez, Inexact restoration method with Lagrangian tangent decrease and new merit function for nonlinear programming, *Journal of Optimization Theory and Applications* 111, pp. 39–58 (2001).
- [27] J. M. Martínez and E. A. Pilotta, Inexact restoration algorithms for constrained optimization, *Journal of Optimization Theory and Applications* 104, pp. 135–163 (2000).
- [28] J. M. Martínez and F. N. C. Sobral, *Constrained Derivative-Free Optimization on Thin Domains*, Technical Report, Department of Applied Mathematics, State University of Campinas, 2011.
- [29] J. M. Martínez and B. F. Svaiter, A practical optimality condition without constraint qualifications for nonlinear programming, *Journal of Optimization Theory and Applications* 118, pp. 117–133 (2003).
- [30] J. J. Moré and S. M. Wild, Benchmarking derivative-free optimization algorithms, *SIAM Journal on Optimization* 20, pp. 172–191 (2009).
- [31] T. D. Plantenga, *HOPSPACK 2.0 User Manual*. Sandia National Laboratories, Albuquerque, NM and Livermore, CA, SAND2009-6265, 2009.

- [32] M. J. D. Powell, *The BOBYQA algorithm for bound constrained optimization without derivatives*, Cambridge NA Report NA2009/06, University of Cambridge, Cambridge, 2009.
- [33] L. Qi and Z. Wei, On the constant positive linear dependence condition and its application to SQP methods, *SIAM Journal on Optimization* 10, pp. 963–981 (2000).
- [34] <http://www.ime.usp.br/~egbirgin/tango/>.