Local Convergence of an Inexact-Restoration Method and Numerical Experiments *

E. G. Birgin † J. M. Martínez ‡

April 29, 2005.

Abstract

Local convergence of an inexact-restoration method for nonlinear programming is proved. Numerical experiments are performed with the objective of evaluating the behavior of the purely local method against a globally convergent nonlinear-programming algorithm.

Key words: Inexact-restoration methods, nonlinear programming, local convergence, numerical experiments.

1 Introduction

We address the nonlinear programming problem:

$$\min f(x)$$

$$\text{s.t. } h(x) = 0, \; x \in \Omega, \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^m$ are smooth functions and $\Omega \subset \mathbb{R}^n$ is a (generally simple) closed and convex set.

Inexact restoration (IR) methods (see [1, 2, 3]) are modern versions of the classical feasible methods ([4, 5, 6, 7, 8, 9, 10, 11, 12]) for nonlinear programming. The main iteration of an IR algorithm consists of two phases: in the restoration phase, infeasibility is reduced and in the optimality phase a Lagrangian function is approximately minimized on an appropriate linear approximation of the constraints. Global convergence is obtained in [1] by means of a trust-region strategy where the trust balls are centered, not in the current

*This work was supported by PRONEX-CNPq/FAPERJ Grant E-26/171.164/2003 - APQ1, FAPESP Grants 03/09169-6 and 01/04597-4, and CNPq. The authors are indebted to Juliano B. Francisco and Yalcin Kaya for their careful reading of the first draft of this paper.

†Associate Professor, Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, São Paulo, SP, Brazil.

‡Professor, Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, Campinas, SP, Brazil.

point, as in several sequential quadratic programming algorithms (see, for example, [13]) but in the inexactly restored point. The merit function used in [1] is a sharp Lagrangian as defined in [14], Example 11.58.

Merit functions are useful tools in all branches of optimization. However, it has been observed that in many practical situations the performance of optimization algorithms that do not impose merit-function decrease is better than the performance of algorithms whose global convergence is based on merit functions. The reason is that merit-function decrease imposes a restrictive path towards the limit point whereas, sometimes, the purely local algorithm climbs over merit-function valleys in a very efficient way.

In unconstrained optimization, nonmonotone strategies, where decrease of the merit function is not required at every iteration ([15]), became a popular tool in the last decade.

In nonlinear programming, the more consistent strategy for globalizing algorithms without the use of merit functions seems to be the filter technique introduced by Fletcher and Leyffer ([16]). Gonzaga, Karas and Vanti ([17]) applied the filter strategy to an algorithm that resembles Inexact Restoration. Previous attempts of eliminating merit functions as globalization tools for semifeasible methods go back to [18].

It is not difficult to modify poor algorithms in order to obtain theoretically globally convergent methods. This can be made using both monotone or nonmonotone strategies. In general, the modification of a poor local method leads to a poor global method. A good globally convergent method is usually good even before the global modification and, sometimes, the purely local version is better than the global one. One of the key features that allow one to predict the practical behavior of an optimization algorithm is the presence of a local convergence theorem with order of convergence. In general, the existence of such a theorem indicates that the model used at each iteration to mimic the original problem is adequate. (Other evidences of this adequacy exist but are less susceptible of mathematical formalization.) This was our motivation for developing a local convergence theory for the Inexact Restoration algorithm. Since our main objective is to explain and test the behavior of methods for solving practical problems, the numerical experiments that complete this paper are directed to evaluate the efficiency and robustness of the purely local algorithm, against globally convergent ones.

The local algorithm and its convergence theory is presented in Section 2. In Section 3 we describe the implementation. Numerical experiments are shown in Section 4 and conclusions are given in Section 5.

2 Local Convergence of Inexact Restoration

In this section we assume that $\Omega \subset I\!R^n$ is closed and convex. We also assume that $f : I\!R^n \to I\!R$ and $h : I\!R^n \to I\!R^m$ admit continuous first derivatives on an open set that contains Ω. The optimization problem to be considered is:

$$\min f(x) \text{ s.t. } h(x) = 0, \quad x \in \Omega. \tag{2}$$

For all $x \in \Omega$, $\lambda \in I\!R^m$, we define the Lagrangian function $L(x, \lambda)$ as

$$L(x, \lambda) = f(x) + \langle h(x), \lambda \rangle.$$

We denote $\nabla h(x) = (\nabla h_1(x), \ldots, \nabla h_m(x))$ and $h'(x) = \nabla h(x)^T$. Therefore,

$$\nabla L(x, \lambda) = \nabla f(x) + \nabla h(x)\lambda.$$

The symbol $\|\cdot\|$ will always denote the Euclidian norm along this paper. Let P be the projection operator on Ω with respect to $\|\cdot\|$. We say that $(x_*, \lambda_*) \in \Omega \times I\!R^m$ is a *critical pair* of the optimization problem (2) if

$$h(x_*) = 0 \tag{3}$$

and

$$P(x_* - \nabla L(x_*, \lambda_*)) - x_* = 0. \tag{4}$$

Under suitable constraint qualifications every local minimizer of (2) defines, with its Lagrange multipliers, a critical pair. (See, for example, [19].) In this section we will analyze a locally convergent algorithm for finding critical pairs, without any mention to the origin of the nonlinear system (3)–(4). We will address the resolution of this nonsmooth nonlinear system of equations using a variation of the inexact restoration (IR) algorithm introduced in [1]. We denote

$$G(x, \lambda) = P(x - \nabla L(x, \lambda)) - x \;\; \forall \;\; x \in \Omega, \; \lambda \in I\!R^m.$$

Therefore, $\|h(x)\|$ is a measure of the feasibility of $x \in \Omega$ and $\|G(x, \lambda)\|$ measures the "optimality" of the pair (x, λ). Given the current iterate $x \in \Omega$, the idea of IR is to find, first, a "more feasible" point $y \in \Omega$, and then, to find a "more optimal" point z such that $z \in \Omega$ and

$$h'(y)(z - y) = 0. \tag{5}$$

(Condition (5) will be relaxed in (8).)

The inexact restoration iteration depends on five algorithmic parameters $\theta \in [0, 1), \eta \in [0, 1)$ and $K_1, K_2, K_3 > 0$. The first two indicate the amount of improvement that we require in the feasibility phase and the optimality phase, respectively. The role of K_1 and K_3 is to maintain the new iterate reasonably close to the current one. (See [1, 3] for details.) The constant K_2 gives a tolerance for the linear infeasibility of the optimality-phase minimizer.

Given $x \in \Omega$ and $\lambda \in I\!R^m$, we say that an IR iteration starting from (x, λ) can be completed (or is well defined) if we can compute $y, z \in \Omega$, $\mu \in I\!R^m$ such that:

$$\|h(y)\| \le \theta \|h(x)\|, \tag{6}$$

$$\|y - x\| \le K_1 \|h(x)\|, \tag{7}$$

$$\|h'(y)(z - y)\| \le K_2 \|G(y, \lambda)\|^2, \tag{8}$$

$$\|P(z - \nabla L(z, \lambda) - \nabla h(y)(\mu - \lambda)) - z\| \le \eta \|G(y, \lambda)\| \tag{9}$$

and

$$\|z - y\| + \|\mu - \lambda\| \le K_3 \|G(y, \lambda)\|. \tag{10}$$

The motivation for the condition (9) comes from considering that, in the optimality phase, one generally minimizes the Lagrangian $L(z, \lambda)$ subject to $z \in \Omega$ and $h'(y)(z-y) = 0$. Writing the optimality conditions for this subproblem and defining $(\mu-\lambda)$ as the vector of Lagrange multipliers corresponding to these conditions, we obtain

$$P(z - \nabla L(z, \lambda) - \nabla h(y)(\mu - \lambda)) - z = 0.$$

So, inequality (9) is an inexact version of this condition. The stability conditions (7) and (10) express the necessity of staying close to the current point if this point is close to feasibility or optimality respectively.

Given the pair $(x, \lambda) \in \Omega \times I\!R^m$, if the IR iteration can be completed giving (z, μ), we denote

$$N_{[\theta,\eta,K_1,K_2,K_3]}(x, \lambda) = (z, \mu).$$

For simplicity, we will always denote

$$N(x, \lambda) = N_{[\theta,\eta,K_1,K_2,K_3]}(x, \lambda) = (z, \mu).$$

Throughout this section we will assume that ∇f and ∇h are Lipschitz-continuous. To simplify the notation, and without loss of generality, we assume that, for the same Lipschitz constant γ and for all $x, w \in \Omega$, $i = 1, \dots, m$,

$$\|\nabla f(x) - \nabla f(w)\| \leq \gamma\|x - w\|, \quad \|\nabla h_i(x) - \nabla h_i(w)\| \leq \gamma\|x - w\|, \tag{11}$$

$$\|\nabla h(x) - \nabla h(w)\| \leq \gamma\|x - w\|, \tag{12}$$

and

$$\|h(w) - h(x) - h'(x)(w - x)\| \leq \gamma\|w - x\|^2. \tag{13}$$

We define the following constants, that will be used along this section:

$$c = \max\{K_1, K_2, K_3\},$$

$$c_1 = 2c + c\gamma,$$

$$c_2 = c\gamma,$$

$$c_3 = c + 2c^2 + c^2\gamma,$$

$$c_4 = c^2\gamma + c.$$

Theorem 2.1. Assume that the IR iteration starting from (x, λ) can be completed and $(z, \mu) = N(x, \lambda)$. Then,

$$\|h(z)\| \leq \theta\|h(x)\| + c_4[\|G(x, \lambda)\| + (c_1 + c_2\|\lambda\|)\|h(x)\|]^2, \tag{14}$$

$$\begin{gathered}\|G(z, \mu)\| \leq \\ \eta[(c_1 + c_2\|\lambda\|)\|h(x)\| + \|G(x, \lambda)\|] + c_4[\|G(x, \lambda)\| + (c_1 + c_2\|\lambda\|)\|h(x)\|]^2,\end{gathered} \tag{15}$$

$$\|z - x\| \leq (c_3 + c_4\|\lambda\|)\|h(x)\| + c\|G(x, \lambda)\| \tag{16}$$

and

$$\|\mu - \lambda\| \leq (c_3 + c_4\|\lambda\|)\|h(x)\| + c\|G(x, \lambda)\|. \tag{17}$$

Proof. By (13),

$$\|h(z) - h(y)\| \leq \|h'(y)(z - y)\| + \gamma\|z - y\|^2.$$

So, by (6), (8) and (10),

$$\|h(z)\| \leq \theta\|h(x)\| + (\gamma c^2 + c)\|G(y, \lambda)\|^2. \tag{18}$$

Now, by (7) and (11)–(13),

$$\|G(y, \lambda) - G(x, \lambda)\| = \|P(y - \nabla L(y, \lambda)) - y - (P(x - \nabla L(x, \lambda)) - x)\|$$

$$\leq \|y - x\| + \|P(y - \nabla L(y, \lambda)) - P(x - \nabla L(x, \lambda))\|$$

$$\leq \|y - x\| + \|y - x + \nabla L(x, \lambda) - \nabla L(y, \lambda)\|$$

$$\leq 2\|y - x\| + \|\nabla f(y) - \nabla f(x)\| + \|[\nabla h(x) - \nabla h(y)]\lambda\|$$

$$\leq 2c\|h(x)\| + \gamma\|y - x\| + \gamma\|y - x\|\|\lambda\|$$

$$\leq (2c + c\gamma + c\gamma\|\lambda\|)\|h(x)\| = (c_1 + c_2\|\lambda\|)\|h(x)\|.$$

Therefore,

$$\|G(y, \lambda)\| \leq \|G(x, \lambda)\| + (c_1 + c_2\|\lambda\|)\|h(x)\|. \tag{19}$$

So, by (18) and (19),

$$\|h(z)\| \leq \theta\|h(x)\| + (\gamma c^2 + c)[\|G(x, \lambda)\| + (c_1 + c_2\|\lambda\|)\|h(x)\|]^2. \tag{20}$$

So, (14) is proved.

Now,

$$\|P(z - \nabla L(z, \mu)) - z\| = \|P(z - \nabla f(z) - \nabla h(z)\mu) - z\|$$

$$= \|P[z - \nabla f(z) - \nabla h(z)(\mu - \lambda) + \nabla h(z)(\mu - \lambda) - \nabla h(z)\mu] - z\|$$

$$= \|P[z - \nabla f(z) - \nabla h(z)\lambda - \nabla h(z)(\mu - \lambda)] - z\|$$

$$= \|P[z - \nabla L(z, \lambda) - \nabla h(y)(\mu - \lambda) + (\nabla h(y) - \nabla h(z))(\mu - \lambda)] - z\|.$$

Using the property

$$\|P(v + w) - z\| \leq \|P(v + w) - P(v)\| + \|P(v) - z\| \leq \|w\| + \|P(v) - z\|$$

with $v = z - \nabla L(z, \lambda) - \nabla h(y)(\mu - \lambda)$ and $w = (\nabla h(y) - \nabla h(z))(\mu - \lambda)$, by (9), (11)–(13) and (10), we get:

$$\|P(z - \nabla L(z, \mu)) - z\|$$

$$\leq \|P[z - \nabla L(z, \lambda) - \nabla h(y)(\mu - \lambda)] - z\| + \|\nabla h(y) - \nabla h(z)\|\|\mu - \lambda\|$$

$$\leq \eta\|G(y, \lambda)\| + \gamma\|y - z\|\|\mu - \lambda\|$$

$$\leq \eta\|G(y, \lambda)\| + \gamma(\|y - z\| + \|\mu - \lambda\|)^2$$

$$\leq \eta\|G(y,\lambda)\| + \gamma c^2\|G(y,\lambda)\|^2.$$

So, by (19),

$$\|G(z,\mu)\| \leq \\ \eta[\|G(x,\lambda)\| + (c_1 + c_2\|\lambda\|)\|h(x)\|] + \gamma c^2[\|G(x,\lambda)\| + (c_1 + c_2\|\lambda\|)\|h(x)\|]^2. \tag{21}$$

Therefore, (15) is also proved.

Now, by (7), (10) and (19),

$$\|z - x\| \leq \|y - x\| + \|z - y\| \leq c\|h(x)\| + c\|G(y,\lambda)\|$$

$$\leq c\|h(x)\| + c[\|G(x,\lambda)\| + (2c^2 + c^2\gamma + c^2\gamma\|\lambda\|)\|h(x)\|]$$

$$= (c + 2c^2 + c^2\gamma + c^2\gamma\|\lambda\|)\|h(x)\| + c\|G(x,\lambda)\|.$$

So, (16) is proved.

Moreover, by (10) and (19),

$$\|\mu - \lambda\| \leq c\|G(y,\lambda)\|$$

$$\leq c\|G(x,\lambda)\| + (2c^2 + c^2\gamma + c^2\gamma\|\lambda\|)\|h(x)\|.$$

Thus, (17) is also proved. □

From now on we assume that $(\bar{x},\bar{\lambda}) \in \Omega \times I\!R^m$ is a critical pair. That is,

$$h(\bar{x}) = 0$$

and

$$G(\bar{x},\bar{\lambda}) = 0.$$

We also define

$$M = 2\|\bar{\lambda}\| + 1,$$

$$c_5 = c_1 + c_2 M,$$

and $H \in I\!R^{2\times 2}$ by

$$H = \begin{pmatrix} \theta & 0 \\ c_5 & \eta \end{pmatrix}.$$

The eigenvalues of H are θ and η. Since both are strictly smaller than 1, given an arbitrarily small $\varepsilon > 0$, there exists a vector norm $\|\cdot\|_H$ on $I\!R^2$ such that

$$\|H\|_H = \rho \leq \max\{\theta,\eta\} + \varepsilon < 1. \tag{22}$$

Moreover, this norm is monotone in the sense that

$$0 \leq v \leq w \Rightarrow \|v\|_H \leq \|w\|_H.$$

From now on, we fix a "contraction" parameter r such that

$$\rho < r < 1. \tag{23}$$

Theorem 2.2. There exist $\varepsilon_1 > 0,\ \delta_1 > 0, \beta > 0$ such that, if r is given by (23),

$$\|x - \bar{x}\| \le \varepsilon_1,\ \|\lambda - \bar{\lambda}\| \le \delta_1,$$

and the IR iteration starting from (x, λ) is well defined, with

$$(z, \mu) = N(x, \lambda),$$

then

$$\|\lambda\| \le M,$$

$$\left\| \begin{pmatrix} \|h(z)\| \\ \|G(z,\mu)\| \end{pmatrix} \right\|_H \le r \left\| \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H, \tag{24}$$

$$\|z - x\| \le \beta \left\| \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H, \tag{25}$$

and

$$\|\mu - \lambda\| \le \beta \left\| \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H. \tag{26}$$

Proof. Take $\delta_0 = \|\bar{\lambda}\| + 1$. Then,

$$\|\lambda - \bar{\lambda}\| \le \delta_0 \Rightarrow \|\lambda\| \le \|\bar{\lambda}\| + \delta_0 \Rightarrow \|\lambda\| \le M.$$

By (14) and (15), if $\|\lambda - \bar{\lambda}\| \le \delta_0$ and the iteration is well defined,

$$\|h(z)\| \le \theta \|h(x)\| + c_4 [\|G(x,\lambda)\| + (c_1 + c_2 M)\|h(x)\|]^2,$$

and

$$\|G(z,\mu)\| \le (c_1 + c_2 M)\|h(x)\| + \eta \|G(x,\lambda)\| + c_4 [\|G(x,\lambda)\| + (c_1 + c_2 M)\|h(x)\|]^2.$$

So, since the norm $\|\cdot\|_H$ is monotone,

$$\begin{aligned} &\left\| \begin{pmatrix} \|h(z)\| \\ \|G(z,\mu)\| \end{pmatrix} \right\|_H \\ &\le \left\| H \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H + c_4 \left\| \begin{pmatrix} [\|G(x,\lambda)\| + (c_1 + c_2 M)\|h(x)\|]^2 \\ [\|G(x,\lambda)\| + (c_1 + c_2 M)\|h(x)\|]^2 \end{pmatrix} \right\|_H \\ &\le \rho \left\| \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H + c_4 [\|G(x,\lambda)\| + (c_1 + c_2 M)\|h(x)\|]^2 \times \left\| \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\|_H. \end{aligned}$$

Now, by the equivalence of norms in $I\!R^2$ there exists $\bar{\alpha} > 0$ such that, for all $a, b > 0$,

$$(c_1 + c_2 M)a + b \le \bar{\alpha} \ \left\| \begin{pmatrix} a \\ b \end{pmatrix} \right\|_H,$$

so,

$$\left\| \begin{pmatrix} \|h(z)\| \\ \|G(z,\mu)\| \end{pmatrix} \right\|_H \le \rho \left\| \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H + c_4 \bar{\alpha}^2 \left\| \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H^2 \left\| \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\|_H.$$

Since $\|h(x)\|$ and $\|G(x,\lambda)\|$ are continuous and vanish at $\bar{x},\bar{\lambda}$, taking δ_1 and ε_1 small enough, with $\delta_1 \le \delta_0$ we obtain (24).

Now, let us prove (25) and (26).

By (16) and (17), if $\|x-\bar{x}\| \le \varepsilon_1$, $\|\lambda-\bar{\lambda}\| \le \delta_1$ and the iteration is well defined,

$$\|z-x\| \le (c_3+c_4M)\|h(x)\| + c\|G(x,\lambda)\| \tag{27}$$

and

$$\|\mu-\lambda\| \le (c_3+c_4M)\|h(x)\| + c\|G(x,\lambda)\|. \tag{28}$$

But, by the equivalence of norms in $I\!R^2$, there exists $\beta > 0$ such that, for all $a, b > 0$,

$$(c_3+c_4M)a + cb \le \beta \left\| \begin{pmatrix} a \\ b \end{pmatrix} \right\|_H.$$

Therefore, taking $a = \|h(x)\|$ and $b = \|G(x,\lambda)\|$, (25) and (26) follow from (27) and (28). □

From now on, for all $x \in \Omega$ such that $\|x-\bar{x}\| \le \varepsilon_1$ and $\|\lambda-\bar{\lambda}\| \le \delta_1$, we define:

$$R(x,\lambda) = \left\| \begin{pmatrix} \|h(x)\| \\ \|G(x,\lambda)\| \end{pmatrix} \right\|_H.$$

In the next theorem we prove that, if (x_0,λ_0) is close enough to the critical pair $(\bar{x},\bar{\lambda})$, the sequence generated by $(x_{k+1},\lambda_{k+1}) = N(x_k,\lambda_k)$ converges to a critical pair. Uniqueness of the critical pair is not assumed. Convergence at a linear rate can take place to a different critical pair than $(\bar{x},\bar{\lambda})$.

Theorem 2.3. Let $(\bar{x},\bar{\lambda})$ be a critical pair. Let ρ and r be given by (22) and (23). Assume that $\varepsilon_2 \in (0,\varepsilon_1]$ and $\delta_2 \in (0,\delta_1]$ are such that the IR iteration starting from (x,λ) can be completed whenever $\|x-\bar{x}\| \le \varepsilon_2$ and $\|\lambda-\bar{\lambda}\| \le \delta_2$.

For all $k = 0,1,2,\ldots$, if $\|x_k-\bar{x}\| \le \varepsilon_2$ and $\|\lambda_k-\bar{\lambda}\| \le \delta_2$, we define

$$(x_{k+1},\lambda_{k+1}) = N(x_k,\lambda_k).$$

Then, there exist $\varepsilon_3 \in (0,\varepsilon_2]$, $\delta_3 \in (0,\delta_2]$ such that, taking $\|x_0-\bar{x}\| \le \varepsilon_3$ and $\|\lambda_0-\bar{\lambda}\| \le \varepsilon_3$ we have:

(a) The whole sequence $\{x_k\}$, $k = 0,1,2,\ldots$ is well defined and

$$\|x_k-\bar{x}\| \le \varepsilon_2, \quad \|\lambda_k-\bar{\lambda}\| \le \delta_2 \tag{29}$$

for all $k = 0,1,2,\ldots$;

(b) $R(x_{k+1},\lambda_{k+1}) \le r\, R(x_k,\lambda_k)$ and $R(x_k,\lambda_k) \le r^k R(x_0,\lambda_0)$ for all $k = 0,1,2,\ldots$

(c) The sequence $\{(x_k,\lambda_k)\}$ is convergent to a critical pair (x_*,λ_*).

(d) For all $k = 0,1,2,\ldots$,

$$\|x_k-x_*\| \le \frac{\beta r^k}{1-r} R(x_0,\lambda_0) \quad \text{and} \quad \|\lambda_k-\lambda_*\| \le \frac{\beta r^k}{1-r} R(x_0,\lambda_0), \tag{30}$$

where $\beta > 0$ is the constant defined in the thesis of Theorem 2.2.

Proof. Define

$$\Phi(\varepsilon, \delta) = \max\{R(x, \lambda) \mid \|x - \bar{x}\| \leq \varepsilon, \|\lambda - \bar{\lambda}\| \leq \delta\}.$$

By the continuity of $R(x, \lambda)$ and the fact that $R(\bar{x}, \bar{\lambda}) = 0$,

$$\lim_{\varepsilon \to 0, \delta \to 0} \Phi(\varepsilon, \delta) = 0.$$

Let $\varepsilon_3 \leq \varepsilon_2/2$ and $\delta_3 \leq \delta_2/2$ such that

$$\Phi(\varepsilon_3, \delta_3) \leq R(x_0, \lambda_0)$$

and

$$\frac{\beta \Phi(\varepsilon_3, \delta_3)}{1 - r} \leq \frac{\min\{\varepsilon_2, \delta_2\}}{2}.$$

Let $x_0 \in \Omega, \lambda_0 \in I\!R^m$ be such that $\|x_0 - \bar{x}\| \leq \varepsilon_3, \|\lambda_0 - \bar{\lambda}\| \leq \delta_3$. Then,

$$\varepsilon_3 + \frac{\beta R(x_0, \lambda_0)}{1 - r} \leq \varepsilon_2 \tag{31}$$

and

$$\delta_3 + \frac{\beta R(x_0, \lambda_0)}{1 - r} \leq \delta_2. \tag{32}$$

Let us prove by induction on k that x_k, λ_k are well defined,

$$R(x_k, \lambda_k) \leq r^k R(x_0, \lambda_0), \tag{33}$$

$$\|x_k - \bar{x}\| \leq \varepsilon_3 + \beta R(x_0, \lambda_0) \sum_{j=0}^{k-1} r^j, \tag{34}$$

$$\|\lambda_k - \bar{\lambda}\| \leq \delta_3 + \beta R(x_0, \lambda_0) \sum_{j=0}^{k-1} r^j. \tag{35}$$

For $k = 0$, (33), (34) and (35) are obviously true. Assume, as inductive hypothesis, that (33), (34) and (35) hold for some k. Then, by (31) and (32), since $\sum_{j=0}^{k-1} r^j \leq \sum_{j=0}^{\infty} r^j = \frac{1}{1-r}$ we have that

$$\|x_k - \bar{x}\| \leq \varepsilon_2 \leq \varepsilon_1, \|\lambda_k - \bar{\lambda}\| \leq \delta_2 \leq \delta_1.$$

Therefore, by the hypothesis of the theorem, x_{k+1} and λ_{k+1} are well defined. Then, by (24),

$$R(x_{k+1}, \lambda_{k+1}) \leq r R(x_k, \lambda_k).$$

So, by the inductive hypothesis (33),

$$R(x_{k+1}, \lambda_{k+1}) \leq r^{k+1} R(x_0, \lambda_0).$$

Now, by Theorem 2.2 and the inductive hypothesis,

$$\begin{aligned}\|x_{k+1}-\bar{x}\| &\le \|x_k-\bar{x}\|+\|x_{k+1}-x_k\| \\ &\le \varepsilon_3+\beta R(x_0,\lambda_0)\sum_{j=0}^{k-1} r^j+\beta R(x_k,\lambda_k) \\ &\le \varepsilon_3+\beta R(x_0,\lambda_0)\sum_{j=0}^{k-1} r^j+\beta r^k R(x_0,\lambda_0) \\ &\le \varepsilon_3+\beta R(x_0,\lambda_0)\sum_{j=0}^{k} r^j.\end{aligned}$$

Therefore, (34) holds replacing k by $k+1$. Analogously, we prove that (35) holds replacing k by $k+1$.

So far, the inductive proof is finished. Thus, the sequence is well defined,

$$\|x_k-\bar{x}\| \le \varepsilon_3+\beta R(x_0,\lambda_0)\sum_{j=0}^{k-1} r^j \le \varepsilon_3+\frac{\beta R(x_0,\lambda_0)}{1-r} \le \varepsilon_2 \tag{36}$$

and

$$\|\lambda_k-\bar{\lambda}\| \le \delta_3+\beta R(x_0,\lambda_0)\sum_{j=0}^{k-1} r^j \le \delta_3+\frac{\beta R(x_0,\lambda_0)}{1-r} \le \delta_2 \tag{37}$$

for all $k=0,1,2,\ldots$ Thus, (a) and (b) are proved.

Now, by Theorem 2.2 and (b), for all $k=0,1,2,\ldots$ we have that

$$\|x_{k+1}-x_k\| \le \beta R(x_k,\lambda_k) \le \beta r^k R(x_0,\lambda_0)$$

and

$$\|\lambda_{k+1}-\lambda_k\| \le \beta R(x_k,\lambda_k) \le \beta r^k R(x_0,\lambda_0).$$

This means that for all $k,j=0,1,2,\ldots,$

$$\|x_{k+j}-x_k\| \le \beta(r^k+\ldots+r^{k+j-1})R(x_0,\lambda_0) \le \frac{\beta r^k}{1-r}R(x_0,\lambda_0)$$

and

$$\|\lambda_{k+j}-\lambda_k\| \le \beta(r^k+\ldots+r^{k+j-1})R(x_0,\lambda_0) \le \frac{\beta r^k}{1-r}R(x_0,\lambda_0).$$

Therefore, $\{x_k\}$ and $\{\lambda_k\}$ are Cauchy sequences, thus convergent to $x_* \in \Omega$ and $\lambda_* \in I\!R^m$ respectively. Taking limits, we have the error estimates

$$\|x_k-x_*\| \le \frac{\beta r^k}{1-r}R(x_0,\lambda_0)$$

and

$$\|\lambda_k-\lambda_*\| \le \frac{\beta r^k}{1-r}R(x_0,\lambda_0).$$

From $R(x_k, \lambda_k) \leq r^k R(x_0, \lambda_0)$ and by the continuity of R we obtain that $R(x_*, \lambda_*) = 0$. Therefore, the theorem is proved. □

Remark 2.1. We used the fact that $\|\cdot\|$ is the Euclidian norm in the theorems above because the properties of the projection operator P are part of the proving arguments. In the particular case in which $\Omega = I\!R^n$ the projection P is the identity. In this case, it is easy to see that the results hold for an arbitrary norm.

Theorem 2.4. In addition to the hypotheses of Theorem 2.3, assume that the parameters θ and η depend on k and tend to zero. Then, $R(x_k, \lambda_k)$ tends to zero Q-superlinearly and (x_k, λ_k) tends to (x_*, λ_*) R-superlinearly.

Proof. The fact that $R(x_k, \lambda_k)$ tends to zero Q-superlinearly follows from Part (b) of Theorem 2.3. By (25) and (26) $\|x_{k+1} - x_k\| + \|\lambda_{k+1} - \lambda_k\|$ is bounded by a sequence that tends superlinearly to zero. This implies that $(\|x_k - x_*\|, \|\lambda_{k+1} - \lambda_*\|)$ tends R-superlinearly to zero. □

Theorem 2.5. In addition to the hypotheses of Theorem 2.3, assume that $\theta = \eta = 0$. Then, $R(x_k, \lambda_k)$ converges Q-quadratically to zero and the convergence of (x_k, λ_k) to (x_*, λ_*) is R-quadratic.

Proof. From (14) and (15), $R(x_k, \lambda_k)$ tends to zero Q-quadratically. By (25) and (26) $\|x_{k+1} - x_k\| + \|\lambda_{k+1} - \lambda_k\|$ is bounded by a sequence that tends quadratically to zero. This implies that $(x_{k+1} - x_*, \lambda_{k+1} - \lambda_*)$ tends R-quadratically to zero. □

Remark 2.2. Although somewhat cumbersome, it is not difficult to prove that, under some classical assumptions, the hypothesis of Theorem 2.3 holds. The more simple case is when $\bar{x}$ is an interior point of Ω. In this case the critical pair $\bar{x}, \bar{\lambda}$ is a solution of the nonlinear system

$$h(x) = 0$$

and

$$\nabla f(x) + \nabla h(x)\lambda = 0.$$

If the Jacobian of this nonlinear system is nonsingular at $(\bar{x}, \bar{\lambda})$, Brent's generalized method ([20, 21]) defines an admissible iteration for constants that only depend on $(\bar{x}, \bar{\lambda})$. The basic properties of this method guarantee that the iteration is well defined in a neighborhood of the critical pair and that the conditions (6)–(10) are satisfied.

The case in which $\bar{x}$ is not interior can be reduced to the interior case after some manipulations assuming nonsingularity of a reduced nonlinear system.

3 Implementation

From now on, we define

$$\Omega = \{x \in I\!R^n \mid \ell \leq x \leq u\}.$$

Algorithm 3.1 is an implementation of (6)–(10). Suppose that the initial pair (x_0, λ_0) is given, $x_0 \in \Omega$, as well as the algorithmic parameters $\theta \in [0,1), \eta \in [0,1)$, $K_1, K_2, \tilde{K}_3 > 0$ and $\varepsilon \geq 0$. The algorithm describes the steps to obtain (x_{k+1}, λ_{k+1}) starting from (x_k, λ_k).

Algorithm 3.1.

Step 1. *Feasibility Phase.* Solve, approximately, the minimization problem

$$\begin{array}{ll} \min_y & \|h(y)\|^2 \\ \text{s.t.} & \|y - x_k\|_\infty \leq K_1 \|h(x_k)\|, \\ & y \in \Omega. \end{array} \tag{38}$$

The approximate solution y_k is asked to satisfy

$$\|h(y_k)\| \leq \max\{\varepsilon,\ \theta \|h(x_k)\|\}. \tag{39}$$

If we are not able to find such an approximate solution, stop the execution declaring *"Failure at the Feasibility Phase"*.

Step 2. *Test Solution.* If $\|h(y_k)\|_\infty \leq \varepsilon$ and $\|G(y_k, \lambda_k)\|_\infty \leq \varepsilon$, terminate the execution of the algorithm. The pair (y_k, λ_k) is an approximate solution of the problem (exact if $\varepsilon = 0$).

Step 3. *Optimality Phase.* Obtain an approximate solution of

$$\begin{array}{ll} \min_z & L(z, \lambda_k) \\ \text{s.t.} & h'(y_k)(z - y_k) = 0, \\ & \|z - y_k\|_\infty \leq \tilde{K}_3 \max\{1, \|y_k\|_\infty\}, \\ & z \in \Omega. \end{array} \tag{40}$$

Let $(\lambda_{k+1} - \lambda_k) \in I\!R^m$ be the vector of Lagrange multipliers associated to the approximate solution x_{k+1} of (40). This approximate solution is asked to satisfy

$$\|h'(y_k)(x_{k+1} - y_k)\| \leq \max\{\varepsilon,\ K_2 \|G(y_k, \lambda_k)\|^2\} \tag{41}$$

and

$$\|\tilde{P}[x_{k+1} - \nabla L(x_{k+1}, \lambda_k) - \nabla h(y_k)(\lambda_{k+1} - \lambda_k)] - x_{k+1}\| \leq \max\{\varepsilon,\ \eta \|G(y_k, \lambda_k)\|\}, \tag{42}$$

where $\tilde{P}$ is de Euclidian projection operator onto the box

$$\Omega \cap \{z \in I\!R^n \mid \|z - y_k\|_\infty \leq \tilde{K}_3 \max\{1,\ \|y_k\|_\infty\}\}.$$

If we are not able to satisfy these requirements we declare *"Failure at the Optimality Phase"*.

Observe that approximate solutions that satisfy (41)–(42) can be always found by ordinary algorithms for linearly constrained minimization since the feasible set is nonempty and compact. Therefore, the diagnostic of Failure at the Optimality Phase can only represent lack of success of the algorithm used to solve the linearly constrained optimization problem (40). As we will see, this failure never occurred in our experiments.

The situation is somewhat different in the feasibility phase, because in this case it is possible to incorporate the theoretically required steplength control in the definition of the optimization problem (38). In this case, failure might be a characteristic of the problem. For example, *"Failure at the Feasibility Phase"* necessarily occurs if x_k is a global minimizer of (38) where $h(x_k) \neq 0$.

It is well known by practitioners that, in the process of solving nonlinear systems, locally convergent methods can be improved by the simple device of maintaining the distance between consecutive iterates under control. See [22]. This is the role of the constraint

$$\|z - y_k\|_\infty \leq \tilde{K}_3 \max\{1, \|y_k\|_\infty\} \tag{43}$$

in (40).

In a neighborhood of a solution, the step-control constraint tends to be inactive. In this case, we usually have that

$$\|P[x_{k+1} - \nabla L(x_{k+1}, \lambda_k) - \nabla h(y_k)(\lambda_{k+1} - \lambda_k)] - x_{k+1}\| \leq \max\{\varepsilon,\ \eta\|G(y_k, \lambda_k)\|\}. \tag{44}$$

In our implementation we used GENCAN, an algorithm introduced in [23] for solving (38) and ALGENCAN, a straightforward Augmented Lagrangian algorithm based on GENCAN for solving (40). Very likely, these are not the best choices from the point of view of efficiency, but they serve for the purpose the main questions that we want to be answered by the numerical experiments, which are related with robustness. Nevertheless, we would like to mention that in recent works ([24, 25, 26]) excellent numerical behavior of Augmented Lagrangian algorithms applied to linearly constrained minimization has been reported.

The way in which the requirement (39) was implemented was as follows: we ran GENCAN with a maximum of 1000 iterations and the stopping criterion $\|h(y)\| \leq \varepsilon$. If, at the final point, (39) is not satisfied, we declare *"Failure at the Feasibility Phase"*.

At the Optimality Phase, we ran ALGENCAN with a maximum of 25 iterations and the stopping criteria

$$\|h'(y_k)(z - y_k)\| \leq \varepsilon,$$

and

$$\|\tilde{P}[z - \nabla L(z, \lambda_k) - \nabla h(y_k)(\mu - \lambda_k)] - z\| \leq \varepsilon.$$

If, at the final point, (41) and (42) are not satisfied, we declare *"Failure at the Optimality Phase"*.

4 Numerical Experiments

The question that we want to answer by means of numerical experiments may be formulated as follows: How bad is the local Inexact Restoration method when compared to globally convergent nonlinear-programming algorithms? The key point is, of course, robustness. The comparison between local and global methods in nonlinear optimization is, sometimes, surprising. As far as in 1979, Moré and Cosnard ([22]) published a numerical study where Brent's method for solving nonlinear systems ([27, 28]) appeared to be better than globally convergent nonlinear solvers when a suitable control for the steplength was used. The analogy between the local Inexact Restoration method and the generalized Brown-Brent methods ([20]) as well as the natural way in which steplength controls appear in our implementation increases the motivation for the numerical study.

In [1] a globally convergent form of Algorithm 3.1 was presented. Global versions of Inexact Restoration algorithms using filters were introduced by Gonzaga, Karas and Vanti ([17]). The main question is: from the practical point of view, how necessary are globally convergent modifications? Is the local method (with step control) competitive with well established global algorithms for Nonlinear Programming from the point of view of robustness?

We selected all the nonlinearly constrained problems with quadratic or nonlinear objective function from the CUTE collection ([29]). The initial points were the ones provided with the problem definition and the initial estimate for the multiplier vector was $\lambda_0 = 0$.

The algorithmic parameters selected were: $\theta = \eta = 0.99$, $K_1 = K_2 = 10^6$, $\tilde{K}_3 = 0.1$, and $\varepsilon = 10^{-4}$.

The globally convergent method selected for numerical comparison was LANCELOT ([30]) with second derivatives, a maximum of 10000 iterations and remaining default options. In both methods we declared *Convergence* when the current iterate (x, λ) satisfies

$$\|h(x)\|_\infty \le \varepsilon \text{ and } \|P(x - \nabla L(x, \lambda)) - x\|_\infty \le \varepsilon.$$

In Algorithm 3.1 we used a maximum of 100 iterations.

The comparison is given in Table 1. In this table, we report the essential quantities that are necessary to compare the robustness of both methods.

Both in LANCELOT as in Algorithm 3.1, IER=0 means that feasibility and optimality up to the desired precision was achieved and IER=1 means that the maximum allowed number of iterations was exhausted. In Algorithm 3.1, IER=2 means failure of the feasibility phase. In LANCELOT, IER=3 means that the algorithm stopped because of a very small step and IER=8 is a failure for a non-reported reason. In spite of the diagnostic IER=0, feasibility was not achieved by LANCELOT in problem SREADIN3.

In HS99, Algorithm 3.1 exhausted the allowed number of iterations. However, it achieved feasibility and the objective function value coincided with the one reported in CUTE (-831079892). Therefore, the obtained point is an accurate approximate solution of the problem but the required precision on the Lagrange condition was not achieved possibly because of the effect of bad scaling on floating point calculations.

Table 1 shows that, surprisingly, Algorithm 3.1 is, at least, as robust as LANCELOT in the set of problems considered.

Let us now address the following question: To what extent, in practice, the implemented algorithm *is* the local algorithm defined by (6)–(10)? In order to answer this question, we define, for each problem and each iteration, the quantities:

$$\theta_k = \|h(y_k)\|/\|h(x_k)\|, \tag{45}$$

$$[K_1]_k = \|y_k - x_k\|/\|h(x_k)\|, \tag{46}$$

$$[K_2]_k = \|h'(y_k)(x_{k+1} - y_k)\|/\|G(y_k, \lambda_k)\|^2, \tag{47}$$

$$\eta_k = \|P(x_{k+1} - \nabla L(x_{k+1}, \lambda_k) - \nabla h(y_k)(\lambda_{k+1} - \lambda_k)) - x_{k+1}\|/\|G(y_k, \lambda_k)\| \tag{48}$$

and

$$[K_3]_k = (\|x_{k+1} - y_k\| + \|\lambda_{k+1} - \lambda_k\|)/\|G(y_k, \lambda_k)\|. \tag{49}$$

In (45)–(49) we use the convention $0/0 = 0$. If, for an infinite sequence generated by Algorithm 3.1, there exist $\bar{\theta}, \bar{\eta}, \bar{K}$ such that $\theta_k \leq \bar{\theta} < 1$, $\eta_k \leq \bar{\eta} < 1$, $[K_1]_k, [K_2]_k, [K_3]_k \leq \bar{K} < \infty$ and if the Lipschitz assumption holds, then the sequence exactly represents the IR iteration. Of course, we never compute an infinite sequence, but the values of the quantities (45)–(49) in a finite computed sequence may give an idea of how much the practical algorithm with step control differs from the theoretical convergent algorithm.

With this in mind, we compute $\theta_k, \eta_k, [K_step1]_k, [K_2]_k, [K_3]_k$ at all the iterations of each problem of our experiments. It is also interesting to discover the number of cases in which the step control constraint (43) is active at the approximate solution x_{k+1} of (40). Some statistics related to these figures are reported in Table 2. Under column #AC, we report the number of iterations in which (43) is active, the last iteration in which it was active, and the total number of iterations performed. The other columns are self-explanatory. The problems for which IRLOC did not stop satisfying the convergence criterion appear at the end of the table.

As expected, except in the cases of Failure at the Feasibility Phase, the value of θ_k was always smaller than 1. This means that the theoretical requirement for an approximate solution of the feasibility phase was always satisfied.

Large values of $[K_2]_k$ (say, greater than 100) appear in only 6 cases. They are associated with subproblems where the linearly constrained solver failed to obtain linear feasibility at the required precision. Nevertheless, is interesting to observe that in all these cases global convergence of the main algorithm took place.

On the other hand, in 14 problems values of $[K_3]_k$ greater than 100 appear and three of these problems are global failures. Large values of $[K_3]_k$ represent situations where the stability condition (10) is hard to satisfy. However, one must be cautious in the analysis of these quantities because they are quite susceptible to scaling of the original problem.

The first column of Table 2 shows that, as expected, the constraint that controls steplength at the optimality phase is not active at the final iterations of the cases in which the method converges. The comparison of this column with the fourth column of the table reveals that, in many cases, although the steplength constraint was active, the theoretical condition $\eta_k < 1$ was fulfilled.

Problem (n, m)	IRLOC					LANCELOT			
	IT	$f(x)$	$\|h(x)\|_\infty$	$\|\nabla L(x)\|_\infty$	IER	$f(x)$	$\|h(x)\|_\infty$	$\|\nabla L(x)\|_\infty$	IER
ALSOTAME (2, 1)	12	8.2085D−02	1.6D−12	1.2D−06	0	8.2085D−02	3.5D−06	1.7D−07	0
BT11 (5, 3)	14	8.2474D−01	6.3D−05	5.0D−05	0	8.2471D−01	7.2D−05	2.2D−05	0
BT6 (5, 2)	11	2.7705D−01	1.6D−05	9.3D−07	0	2.7705D−01	1.2D−05	3.4D−05	0
CLNLBEAM (33, 20)	40	3.4503D+02	5.0D−05	3.1D−06	0	3.4503D+02	5.7D−05	5.3D−05	0
DNIEPER (61, 24)	4	1.8744D+04	6.9D−06	3.1D−06	0	1.8743D+04	5.7D−05	9.6D−06	0
DTOC2 (298, 196)	9	4.8681D−01	1.8D−05	1.8D−06	0	4.8631D−01	1.2D−05	4.4D−05	0
DTOC4 (29, 18)	3	3.7509D+00	6.2D−05	8.7D−05	0	3.7508D+00	4.6D−06	3.8D−05	0
DTOC6 (201, 100)	16	7.2790D+02	1.4D−05	2.8D−05	0	7.2795D+02	4.0D−06	6.4D−05	0
HS100LNP (7, 2)	7	6.8063D+02	8.1D−06	5.0D−07	0	6.8063D+02	3.1D−06	1.1D−05	0
HS107 (9, 6)	4	5.0548D+03	3.7D−05	8.8D−05	0	5.0545D+03	4.9D−05	1.5D−08	0
HS111 (10, 3)	16	-4.7761D+01	5.8D−05	1.2D−05	0	-4.7761D+01	2.6D−05	4.1D−05	0
HS111LNP (10, 3)	16	-4.7761D+01	6.7D−05	4.6D−05	0	-4.7761D+01	2.6D−05	3.6D−05	0
HS26 (3, 1)	26	5.3806D−09	8.1D−07	3.1D−06	0	2.3577D−07	1.6D−06	3.9D−05	0
HS40 (4, 3)	4	-2.4999D−01	1.3D−05	1.8D−05	0	-2.5002D−01	2.1D−05	2.9D−05	0
HS46 (5, 2)	10	4.4356D−07	1.3D−08	7.0D−05	0	7.4195D−07	1.5D−05	6.4D−05	0
HS47 (5, 3)	17	1.8142D−08	6.4D−05	2.6D−05	0	2.3121D−07	2.5D−05	6.3D−05	0
HS56 (7, 4)	12	-3.4560D+00	1.4D−05	1.8D−05	0	-3.4559D+00	5.7D−05	4.4D−06	0
HS60 (3, 1)	6	3.2569D−02	7.7D−05	9.1D−06	0	3.2568D−02	9.3D−07	3.2D−05	0
HS7 (2, 1)	12	-1.7321D+00	9.8D−07	2.1D−06	0	-1.7320D+00	4.0D−06	6.2D−06	0
HS77 (5, 2)	11	2.4150D−01	7.4D−06	3.7D−06	0	2.4151D−01	1.8D−06	1.8D−05	0
HS78 (5, 3)	4	-2.9197D+00	3.9D−05	4.1D−09	0	-2.9197D+00	4.3D−06	4.8D−05	0
HS79 (5, 3)	4	7.8779D−02	8.3D−05	8.3D−06	0	7.8777D−02	2.6D−05	1.6D−05	0
HS80 (5, 3)	4	5.3947D−02	3.4D−05	1.7D−06	0	5.3950D−02	5.6D−06	2.4D−05	0
HS81 (5, 3)	4	5.3950D−02	2.2D−05	8.9D−06	0	5.3950D−02	8.2D−06	6.3D−05	0
HS99 (7, 2)	100	-8.3108D+08	7.4D−06	1.0D−01	1	-8.3125D+08	1.9D+01	1.28+00	1
HS99EXP (31, 21)	2	-1.4195D+11	2.7D+05	7.4D+05	2	-1.6249D+21		2.47+11	1
LAKES (90, 78)	2	7.2323D+11	4.7D+02	5.5D+05	2	3.7607D+10			1
LCH (150, 1)	13	-4.2301D+00	8.1D−05	2.3D−06	0	-4.1990D+00	1.4D−05	4.3D−07	0
LEWISPOL (6, 9)	11	1.1268D+00	3.4D−05	1.2D−08	0	1.1267D+00	3.6D−05	2.5D−06	0
LUBRIF (151, 100)	2	0.0000D+00	3.2D+00	0.0D+00	2	0.0000D+00	3.2D+00	8.3D−02	8
ORTHRDM2 (4003, 2000)	4	1.5553D+02	9.1D−08	5.1D−07	0	1.5553D+02	1.6D−07	3.9D−05	0
ORTHRDS2 (203, 100)	4	3.0432D+01	8.9D−05	7.1D−07	0	3.0433D+01	7.2D−05	1.2D−05	0
ORTHREGD (23, 10)	4	3.4121D+00	8.4D−06	3.1D−06	0	3.4121D+00	2.6D−08	2.0D−05	0
ORTHREGE (36, 20)	33	5.9596D+00	2.0D−05	3.2D−06	0	3.3908D+00	7.3D−06	8.4D−05	0
ORTHREGF (152, 49)	7	1.3150D+00	2.0D−05	2.2D−06	0	1.3149D+00	4.8D−05	4.3D−05	0
ORTHRGDM (4003, 2000)	4	6.0562D+02	4.6D−07	1.0D−06	0	6.0562D+02	2.6D−05	3.5D−09	0
ORTHRGDS (103, 50)	94	2.0735D+07	1.3D+02	1.0D+20	2	1.0815D+02	8.8D−03	5.0D−05	3
READING1 (102, 50)	17	-1.6067D−01	3.2D−05	4.7D−05	0	-1.6067D−01	2.0D−05	8.4D−06	0
READING3 (102, 51)	17	-1.5296D−01	8.6D−06	4.1D−05	0	-1.5296D−01	5.1D−05	5.2D−05	0
READING5 (51, 50)	2	2.2532D−08	5.9D−06	2.9D−05	0	8.1371D−06	2.2D−05	5.5D−05	0
READING6 (102, 50)	100	-1.4462D+02	6.0D−03	4.2D−02	1	-9.9392D+15	2.0D+07	1.17+09	1
READING9 (402, 200)	9	-4.4308D−02	1.4D−05	1.2D−05	0	-6.0705D+39	3.0D+18	2.06+06	1
ROBOT (14, 2)	3	6.5932D+00	1.5D−05	3.2D−10	0	6.5930D+00	2.6D−05	4.1D−08	0
SREADIN3 (12, 6)	16	-1.9147D−01	1.2D−05	4.4D−05	0	-1.9147D−01	3.3D−04	2.0D−06	0
SSNLBEAM (33, 20)	31	3.3779D+02	4.6D−05	3.2D−06	0	3.3776D+02	4.9D−05	2.0D−05	0
TRAINH (48, 22)	3	1.2004D+01	5.0D−05	6.1D−06	0	1.2004D+01	6.5D−05	1.0D−05	0
ZAMB2-10 (270, 96)	6	-1.5824D+00	8.4D−05	7.2D−05	0	-1.5812D+00	3.5D−05	6.8D−05	0
ZAMB2-11 (270, 96)	4	-1.1161D+00	5.8D−06	8.1D−05	0	-1.1157D+00	5.4D−05	8.5D−05	0
ZAMB2-8 (138, 48)	6	-1.5294D−01	1.4D−05	2.6D−06	0	-1.5293D−01	9.5D−06	3.4D−05	0
ZAMB2-9 (138, 48)	5	-3.5457D−01	1.9D−05	2.0D−05	0	-3.5422D−01	8.7D−05	2.3D−05	0
ZAMB2 (1326, 480)	7	-4.1422D+00	1.0D−05	5.8D−05	0	-4.1421D+00	2.6D−05	8.4D−05	0

Table 1: IRLOC vs. LANCELOT: measuring robustness.

Problem (n, m)	#AC			$\max_k\{\theta_k\}$	$\max_k\{[K_1]_k\}$	$\#\{\eta_k \geq 1\}$	$\max_k\{[K_2]_k\}$	$\max_k\{[K_3]_k\}$
ALSOTAME (2, 1)	9	9	12	3.119D−07	1.000D+00	3	9.552D−03	2.223D+00
BT11 (5, 3)	11	11	14	9.060D−04	7.628D−01	0	2.216D−01	9.213D−01
BT6 (5, 2)	8	8	11	1.919D−02	2.286D−01	0	9.967D+00	2.830D+00
CLNLBEAM (33, 20)	37	37	40	9.358D−01	3.404D+01	18	4.949D−01	1.346D+03
DNIEPER (61, 24)	1	1	4	8.308D−05	3.349D−01	0	2.003D−05	1.796D+00
DTOC2 (298, 196)	6	6	9	9.947D−02	1.997D+00	0	2.566D+02	3.995D+01
DTOC4 (29, 18)	0	0	3	2.616D−01	2.369D+00	0	4.549D−02	3.247D+00
DTOC6 (201, 100)	13	13	16	1.118D−02	9.468D+00	1	6.869D−01	6.306D+00
HS100LNP (7, 2)	3	3	7	2.405D−05	5.879D−02	0	9.358D−01	1.473D−01
HS107 (9, 6)	1	1	4	1.398D−02	1.268D+00	1	5.251D−05	6.889D+03
HS111 (10, 3)	14	14	16	1.964D−01	1.798D+00	0	5.621D+02	9.478D+02
HS111LNP (10, 3)	14	14	16	2.115D−01	1.702D+00	0	2.736D+02	8.157D+02
HS26 (3, 1)	24	24	26	6.276D−03	2.006D−01	0	3.452D−02	3.334D+00
HS40 (4, 3)	1	1	4	5.479D−06	8.047D−01	0	1.538D−01	1.335D+00
HS46 (5, 2)	7	7	10	7.670D−03	3.109D−01	1	1.394D+00	7.526D+00
HS47 (5, 3)	15	15	17	1.822D−03	9.181D−01	0	3.719D−02	8.772D+00
HS56 (7, 4)	9	9	12	9.412D−02	3.437D−01	4	1.105D−03	1.530D+00
HS60 (3, 1)	3	3	6	3.301D−03	8.632D−02	0	6.310D−01	3.964D−01
HS7 (2, 1)	9	9	12	6.074D−03	2.879D−01	6	2.487D−02	4.237D−01
HS77 (5, 2)	8	8	11	3.203D−03	2.486D−01	0	1.263D+00	2.304D+00
HS78 (5, 3)	0	0	4	5.431D−02	1.616D−01	0	1.886D+01	3.102D−01
HS79 (5, 3)	1	1	4	1.955D−04	3.913D−01	1	5.717D−01	1.482D+00
HS80 (5, 3)	1	1	4	2.335D−05	1.621D−01	0	6.219D+00	3.818D+00
HS81 (5, 3)	1	1	4	2.325D−05	1.621D−01	0	4.641D−02	1.188D+00
LCH (150, 1)	7	8	13	6.363D−03	4.968D−01	1	6.190D−02	5.472D+00
LEWISPOL (6, 9)	8	8	11	3.031D−01	1.095D+00	0	3.812D+02	1.043D+00
ORTHRDM2 (4003, 2000)	0	0	4	2.965D−03	7.061D−03	0	5.207D−03	8.479D−02
ORTHRDS2 (203, 100)	0	0	4	2.007D−01	6.696D−03	0	6.011D−01	2.500D+00
ORTHREGD (23, 10)	0	0	4	1.271D−05	5.650D−03	0	1.493D−01	4.109D−01
ORTHREGE (36, 20)	26	27	33	1.149D−01	9.302D−01	12	2.028D+02	4.771D+00
ORTHREGF (152, 49)	3	3	7	1.675D−03	8.462D−01	1	3.352D+00	9.577D−01
ORTHRGDM (4003, 2000)	0	0	4	1.971D−05	7.135D−03	0	1.303D−05	1.803D−01
READING1 (102, 50)	14	14	17	5.903D−03	4.228D−01	3	3.416D+00	8.125D+01
READING3 (102, 51)	15	15	17	5.131D−03	9.268D−02	6	8.963D+01	9.950D+02
READING5 (51, 50)	0	0	2	1.137D−05	9.092D−01	0	4.259D−04	7.696D−01
READING9 (402, 200)	8	8	9	9.653D−01	1.020D+01	7	9.941D+00	6.791D+02
ROBOT (14, 2)	0	0	3	4.468D−07	5.444D−01	0	4.124D+02	1.375D+00
SREADIN3 (12, 6)	13	13	16	1.408D−02	1.678D−01	5	3.275D−01	1.246D+01
SSNLBEAM (33, 20)	28	28	31	9.246D−01	4.122D+01	20	2.957D−01	1.625D+02
TRAINH (48, 22)	0	0	3	2.958D−05	5.099D+00	0	3.091D+02	4.884D+01
ZAMB2-10 (270, 96)	4	4	6	3.087D−03	4.063D+00	0	2.825D+03	1.355D+05
ZAMB2-11 (270, 96)	2	2	4	3.623D−05	5.798D+00	0	1.535D+01	1.727D+04
ZAMB2-8 (138, 48)	0	0	6	1.454D−01	3.828D+00	1	8.186D+02	2.267D+04
ZAMB2-9 (138, 48)	2	2	5	4.587D−01	3.052D+00	0	3.516D+00	1.190D+03
ZAMB2 (1326, 480)	4	4	7	9.401D−03	8.472D+00	0	2.595D+02	8.516D+04
HS99 (7, 2)	1	1	100	1.848D−09	1.079D−06	97	3.031D−06	1.633D+04
HS99EXP (31, 21)	1	1	2	9.999D−01	6.581D−01	1	9.210D−12	7.137D−02
LAKES (90, 78)	1	1	2	9.954D−01	8.319D+00	0	3.597D−12	2.343D−03
LUBRIF (151, 100)	0	0	2	1.000D+00	4.158D−01	0	0.000D+00	0.000D+00
ORTHRGDS (103, 50)	93	93	94	3.174D−01	6.021D−02	82	6.825D−09	1.378D+06
READING6 (102, 50)	100	100	100	8.802D−03	1.912D+00	36	6.966D−02	1.251D+02

Table 2: IRLOC statistics.

5 Conclusions

Inexact-restoration algorithms for nonlinear programming are based on inexact achievement of feasibility at each iteration followed by inexact minimization of the Lagrangian on a linear approximation of the constraints. Different methods can be used at both phases of the IR algorithm. In this paper we proved a local convergence result for an inexact-restoration algorithm. Essentially, the theorem says that if the IR iteration is well defined in a neighborhood of the solution, then linear convergence takes places to some solution of the KKT system. Under additional assumptions the convergence is superlinear or quadratic.

Based on the fact that, in Newtonian methods for nonlinear systems of equations, practical convergence can be dramatically improved by means of simple steplength control modifications, we proposed a modification of the optimality phase of IR that implicitly maintains the steplength under control. The proposed modification resembles a trust-region constraint added to the natural constraints of the feasibility phase. However, this trust-region constraint is fixed, and not reduced according to merit-function decrease as in [1].

Numerical experiments showed that the IR algorithm with this simple modification is at least as robust as a well established globally convergent nonlinear programming method in a set of problems taken from the CUTE collection. A more careful observation of the tests showed that, in spite of the steplength control, the iterations satisfied the theoretical requirements of the local algorithm at most iterations.

The conclusion of this work is not that one should abandon the project of defining algorithms with the best possible convergence theories, including global convergence, but to put in evidence what kind of practical effects one should expect from globally convergent methods (with or without merit functions). It seems that one should be tolerant with local methods that use a lot of information about the true problem, as Inexact Restoration does, and that the main effect of global modifications should be to maintain the steplength under control. Probably, this reinforces the importance of working with filter strategies and with algorithms that do not force merit function decrease at every iteration.

The results of this paper can be straightforwardly extended to the resolution of KKT systems of type:

$$h(x) = 0, \ x \in \Omega,$$

$$P(x - F(x) - \nabla h(x)\lambda) - x = 0.$$

Essentially, the modifications in the proof required to consider these systems consist in the judicious replacement of $\nabla f(x)$ by $F(x)$ in the proper places. This opens the path for the application of Inexact Restoration to variational inequalities, equilibrium problems and other extensions of constrained optimization.

References

[1] MARTÍNEZ, J.M., *Inexact Restoration Method with Lagrangian Tangent Decrease and New Merit Function for Nonlinear Programming*, Journal of Optimization Theory

and Applications, Vol. 111, pp. 39–58, 2001.

[2] MARTÍNEZ, J.M., and PILOTTA, E.A., *Inexact Restoration Algorithms for Constrained Optimization*, Journal of Optimization Theory and Applications, Vol. 104, pp. 135–163, 2000.

[3] MARTÍNEZ, J.M., and PILOTTA, E.A., *Inexact Restoration Methods for Nonlinear Programming: Advances and Perspectives*, Optimization and Control with Applications, Edited by L.Q. Qi, K.L. Teo and X.Q. Yang, Kluwer Academic Publishers, 2005.

[4] ABADIE, J., and CARPENTIER, J., *Generalization of the Wolfe Reduced-Gradient Method to the Case of Nonlinear Constraints*, Optimization, Edited by R. Fletcher, Academic Press, New York, NY, pp. 37–47, 1968.

[5] DRUD, A., *CONOPT: A GRG Code for Large Sparse Dynamic Nonlinear Optimization Problems*, Mathematical Programming, Vol. 31, pp. 153–191, 1985.

[6] LASDON, L.S., *Reduced Gradient Methods*, Nonlinear Optimization 1981, Edited by M.J.D. Powell, Academic Press, New York, NY, pp. 235–242, 1982.

[7] MIELE, A., HUANG, H.Y., and HEIDEMAN, J.C., *Sequential Gradient-Restoration Algorithm for the Minimization of Constrained Functions: Ordinary and Conjugate Gradient Version*, Journal of Optimization Theory and Applications, Vol. 4, pp. 213–246, 1969.

[8] MIELE, A., LEVY, A.V., and CRAGG, E.E., *Modifications and Extensions of the Conjugate-Gradient Restoration Algorithm for Mathematical Programming Problems*, Journal of Optimization Theory and Applications, Vol. 7, pp. 450–472, 1971.

[9] MIELE, A., SIMS, E.M., and BASAPUR, V.K., *Sequential Gradient-Restoration Algorithm for Mathematical Programming Problems with Inequality Constraints, Part 1, Theory*, Rice University, Aero-Astronautics Report No. 168, 1983.

[10] ROSEN, J.B., *The Gradient Projection Method for Nonlinear Programming, Part 1, Linear Constraints*, SIAM Journal on Applied Mathematics, Vol. 8, pp. 181–217, 1960.

[11] ROSEN, J.B., *The Gradient Projection Method for Nonlinear Programming, Part 2, Nonlinear Constraints*, SIAM Journal on Applied Mathematics, Vol. 9, pp. 514–532, 1961.

[12] ROSEN, J.B., and KREUSER, J., *A Gradient Projection Algorithm for Nonlinear Constraints*, Numerical Methods for Nonlinear Optimization, Edited by F.A. Lootsma, Academic Press, London, UK, pp, 297–300, 1972.

[13] GOMES, F.M., MACIEL, M.C., and MARTÍNEZ, J.M., *Nonlinear Programming Algorithms Using Trust Regions and Augmented Lagrangians with Nonmonotone Penalty Parameters*, Mathematical Programming, Vol. 84, pp. 161–200, 1999.

[14] ROCKAFELLAR, R.T., and WETS, R.J.-B., *Variational Analysis*, Springer-Verlag, Berlin, Germany, 1998.

[15] GRIPPO, L., LAMPARIELLO, F., and LUCIDI, S., *A Class of Nonmonotone Stabilization Methods in Unconstrained Optimization*, Numerische Mathematik, Vol. 59, pp. 779–805, 1991.

[16] FLETCHER, R., and LEYFFER, S., *Nonlinear Programming without a Penalty Function*, Mathematical Programming, Vol. 91, pp. 239–269, 2002.

[17] GONZAGA, C.C., KARAS, E., and VANTI, M., *A Globally Convergent Filter Method for Nonlinear Programming*, SIAM Journal on Optimization, Vol. 14, pp. 646–669, 2003.

[18] BIELSCHOWSKY, R.H., *Nonlinear Programming Algorithms with Dynamic Definition of Near-Feasibility: Theory and Implementations*, University of Campinas, PhD Thesis, 1996.

[19] BERTSEKAS, D.P., *Nonlinear Programming*, 2nd Edition, Athena Scientific, Belmont, Massachusetts, 1999.

[20] MARTÍNEZ, J.M., *Generalization of the Methods of Brent and Brown for Solving Nonlinear Simultaneous Equations*, SIAM Journal on Numerical Analysis, Vol. 16, pp. 434–448, 1979.

[21] MARTÍNEZ, J.M., *Solving nonlinear simultaneous equations with a generalization of Brent's method*, BIT, Vol. 20, pp. 501–510, 1980.

[22] MORÉ, J.J., and COSNARD, M.Y., *Numerical Solution of Nonlinear Equations*, ACM Transactions on Mathematical Software, Vol. 5, pp. 64–85, 1979.

[23] BIRGIN, E.G., and MARTÍNEZ, J.M., *Large-Scale Active-Set Box-Constrained Optimization Method with Spectral Projected Gradients*, Computational Optimization and Applications, Vol. 23, pp. 101–125, 2002.

[24] DOSTÁL, Z., FRIEDLANDER, A., and SANTOS, S.A., *Augmented Lagrangians with Adaptive Precision Control for Quadratic Programming with Equality Constraints*, Computational Optimization and Applications, Vol. 14, pp. 37–53, 1999.

[25] DOSTÁL, Z., FRIEDLANDER, A., and SANTOS, S.A., *Augmented Lagrangians with Adaptive Precision Control for Quadratic Programming with Simple Bounds and Equality Constraints*, SIAM Journal on Optimization, Vol. 13, pp. 1120–1140, 2003.

[26] DOSTÁL, Z., GOMES, F.A.M., and SANTOS, S.A., *Solution of Contact Problems by FETI Domain Decomposition with Natural Coarse Space Projection*, Computational Method in Applied Mechanics and Engineering, Vol. 190, pp. 1611–1627, 2002.

[27] BRENT, R.P., *Some Efficient Algorithms for Solving Systems of Nonlinear Equations*, SIAM Journal on Numerical Analysis, Vol. 10, pp. 327–344, 1973.

[28] MORÉ, J.J., and COSNARD, M.Y., *Algorithm 554 - BRENTM, A FORTRAN Subroutine for the Numerical Solution of Systems of Nonlinear Equations*, ACM Transactions on Mathematical Software, Vol. 6, pp. 240–251, 1980.

[29] BONGARTZ, I., CONN, A.R., GOULD, N.I.M., and TOINT, Ph.L., *CUTE: Constrained and Unconstrained Testing Environment*, ACM Transactions on Mathematical Software, Vol. 21, pp. 123–160, 1995.

[30] CONN, A.R., GOULD, N.I.M., and TOINT, Ph.L., *A Globally Convergent Augmented Lagrangian Algorithm for Optimization with General Constraints and Simple Bounds*, SIAM Journal on Numerical Analysis, Vol. 28, pp. 545–572, 1991.