Cutting Planes and a Biased Newton Direction for Minimizing Quasiconvex Functions*

N. Echebest[1] *M. T. Guardarucci*[1] *H. Scolnik*[2]
M.C. Vacchino[1]

[1]Departamento de Matemática
Universidad de La Plata
Buenos Aires, Argentina
opti@mate.unlp.edu.ar

[2]Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires, Argentina
scolnik@fd.com.ar

Abstract
A biased Newton direction is introduced for minimizing quasiconvex functions with bounded level sets. It is a generalization of the usual Newton's direction for strictly convex quadratic functions. This new direction can be derived from the intersection of approximating hyperplanes to the epigraph at points on the boundary of the same level set. Based on that direction, an unconstrained minimization algorithm is presented. It is proved to have global and local-quadratic convergence under standard hypotheses. These theoretical results may lead to different methods based on computing search directions using only first order information at points on the level sets. Most of all if the computational cost can be reduced by relaxing some of the conditions according for instance to the results presented in the Appendix. Some tests are presented to show the qualitative behavior of the new direction and with the purpose to stimulate further research on these kind of algorithms.

Keywords: quasiconvex functions, level sets, discretization methods.

1 Introduction

We propose a new descent direction for solving the problem $min_{x \in \Omega} f(x)$, $f \in C^2(\Omega)$, where $\Omega \subset \Re^n$ is an open set, and f is such that its level sets are convex and bounded. The aim of this paper is to derive a descent direction by gathering non

*Work supported by the Universities of Buenos Aires and La Plata, Argentina.

local geometric information of the function at suitable chosen points on the boundary of the same level set. The gradients at those points lead to a finite difference system of equations whose solution gives a search direction which is Newton's when f is a convex quadratic function. Such direction comes from a linear interpolation of gradients at those points. Since the discretization steps are distances between points on the level set corresponding to the current iteration, this method is unlikely trapped by poor local models. Geometrically speaking, at the current iteration such a direction is obtained by computing the intersection of the hyperplane tangent to the epigraph of the function with approximating hyperplanes at the points chosen on the level set. In Gaudioso and Monaco [11](1994) the authors show that, for strictly convex quadratic functions, the Newton's direction may be obtained via an appropiate definition of a set of shifted supporting hyperplanes to the epigraph of the function. Analogously we presented similar results and geometrical interpretations in [7](1993) and [8] (1994) which led to the current paper.

We present a minimization algorithm based on this new descent direction and prove that the generated sequence is globally convergent. If in addition we assume Lipschitz continuity of the second derivatives, then we also prove that the algorithm is locally quadratically convergent.

With the purpose of studying the behavior of an algorithm based on this new direction we implemented it using a routine for determining points sufficiently close to the level sets instead of exact ones as used for deriving the theoretical results. The effects of this special relaxation, which aims at reducing the computational cost, is studied in the Appendix showing that it is possible to preserve the main theoretical properties of the algorithm.

The paper is organized as follows: in Section 2 the algorithm defining the descent direction is described. In Section 3 the minimization algorithm is defined and global convergence is proved. In Section 4 we prove that our method is locally quadratically convergent to stationary points where the Hessian is positive definite. In Section 5 special results for pseudo-convex functions are given. In Section 6 some numerical experiences are described with the purpose of showing the qualitative behavior of the new direction.

In order to improve readability some auxiliary results needed are proved in the Appendix. We also present there a practical implementation of the algorithm introduced in Section 2, which determines approximate points on the level sets while keeping essential properties of Algorithm 3.1.

2 The search direction

In this section we shall obtain the descent direction at the current point x_c, using the gradients of $f(x)$ calculated at specially chosen points on the level surface.

We assume $f \in C^2(\Omega)$, where Ω is a convex set in $\Re^n$. Further f is quasiconvex in Ω and its level sets are compact. For the sake of completeness we include the following definition:

Definition 2.1: The function f is said to be quasiconvex if, for each x_1 and $x_2 \in \Omega$, the following is true:

$$f(\lambda x_1 + (1-\lambda)x_2) \leq maximun\{f(x_1), f(x_2)\} \; for \; each \; \lambda \in (0,1)$$

It is known that a quasiconvex function can be characterized by the convexity of its level sets ([1]).

We denote by $L(x_c) = \{x : f(x) \leq f(x_c)\}$ the level set at x_c, and the corresponding level surface $C(x_c) = \{x : f(x) = f(x_c)\}$

The l_2 norm will be used throughout this paper.

The definition of the descent direction will take into account a property shared by convex and quasiconvex functions, that is the convexity of their level sets.

It is known that the minimizer of a convex quadratic function can be obtained as the solution of a system of linear equations involving only the first derivatives. In Friedlander, Martinez and Scolnik [10](1979) this result was extended to the rank deficient case and more recently is presented in the paper by Gaudioso and Monaco [11].

Given $f(x) = \frac{1}{2}x^THx + b^Tx + c, \;\; H = H^T$ positive definite, we have

Theorem 2.1: Given $y^0 \in \Re^n$ arbitrary and $p^0, \ldots, p^{n-1}$ linearly independent directions in $\Re^n$, and $\gamma^i = Hp^i$, $i = 0, \ldots, n-1$, then $x^* = argminf(x)$ is obtained from the solution of the system

$$\{\langle \gamma^i, x - y^0 \rangle = -\langle p^i, \nabla f(y^0) \rangle, i = 0, \ldots, n-1\}. \tag{2.1}$$

Proof.(see [11], Proposition 1.1) □

When considering points $y^0, y^1, \ldots, y^n$ determining linearly independent directions $p^{i-1} = y^i - y^0$, for i=1,..., n, since

$Hp^{i-1} = H(y^i - y^0) = \nabla f(y^i) - \nabla f(y^0)$, the system (2.1) can be written as

$$\{(\nabla f(y^i) - \nabla f(y^0))^T(x - y^0) = -\nabla f(y^0)^T(y^i - y^0), i = 1, \ldots, n\}. \tag{2.2}$$

In particular, if the points defining the directions belong to $C(y^0) = \{y : f(y) = f(y^0)\}$, from the fact that f is quadratic, the following equalities hold :

$$\nabla f(y^i)^T(y^i - y^0) = \nabla f(y^0)^T(y^0 - y^i), \;\; i = 1, \ldots, n \tag{2.3}$$

$$\nabla f(y^i)^T(x^* - y^i) = \nabla f(y^0)^T(x^* - y^0), \;\; i = 1, \ldots, n, \tag{2.4}$$

where $x^* = argminf(x)$.

In this case, we obtain

Lemma 2.1: If $x^* = argminf(x)$, $y^i \in C(y^0)$, i=1,...,n and the directions $p^{i-1} = y^i - y^0$ are linearly independent, then x^* is the solution of the system

$$\{\nabla f(y^i)^T(x - y^i) = \nabla f(y^0)^T(x - y^0), i = 1, \ldots, n\} \tag{2.5}$$

Moreover, the systems (2.2) and (2.5) are equivalent.

Proof. That x^* is solution of (2.5) follows straightforwardly from (2.4). Using (2.3) a simple calculation shows that both systems are equivalent. ∎

The above results have an interesting geometrical interpretation ([7],[11]). That is, x^* coincides with the abscissa of the intersection of the supporting hyperplanes to the epigraph of f at the points $(y^i, f(y^0))$, which arising from the points chosen on $C(y^0)$. This result agrees with the geometric observation that, due to the symmetry, the supporting hyperplanes at points on the same level surface of a strictly convex quadratic function meet at the minimizer.

It is worthwhile to analyze the solution of the system (2.2) and (2.5) when the points $y^1, \ldots, y^j$ with $j \le n$ on $C(y^0)$ are sequentially obtained from y^0, so that

$$y^i = y^{i-1} + h_{i-1}p^{i-1} \tag{2.6}$$

where $p^0, \ldots, p^{j-1}$ are H conjugate directions.

From the very well-known properties of the conjugate directions ([9], [13]), it follows easily that:

$$\langle \nabla f(y^j) - \nabla f(y^i), y^i - y^0 \rangle = 0 \quad for\ 1 \le i < j, \tag{2.7}$$

$$\langle y^j - y^i, \nabla f(y^i) - \nabla f(y^0) \rangle = 0 \quad for\ 1 \le i < j. \tag{2.8}$$

Further, using (2.8) and (2.3), we have

$$\langle \nabla f(y^i) - \nabla f(y^0), y^j - y^0 \rangle = 2\langle -\nabla f(y^0), y^i - y^0 \rangle \tag{2.9}$$

for $1 \le i \le j$.

Then it is easy to prove that $x^+ = (y^0 + y^j)/2$ is a solution of the system

$$\{\langle \nabla f(y^i), x - y^i \rangle = \langle \nabla f(y^0), x - y^0 \rangle, i = 1, \ldots, j\} \tag{2.10}$$

using (2.9) and the substitution $x^+ - y^0 = (y^j - y^0)/2$ in the equivalent expression

$$\{\langle \nabla f(y^i) - \nabla f(y^0), x - y^0 \rangle = \langle -\nabla f(y^0), y^i - y^0 \rangle,\ i = 1, \ldots, j\} \tag{2.11}$$

Now, let us denote by $[y^1 - y^0, \ldots, y^j - y^0]$ the subspace spanned by the vectors $\{y^1 - y^0, \ldots, y^j - y^0\}$ and by $V(y^0, y^1 - y^0, y^2 - y^0, \ldots, y^j - y^0)$ the affine subspace which contains y^0.

Lemma 2.2: Assuming the conjugacy of the directions and the definition of the points given by (2.6), the solution $x^+ = (y^0 + y^j)/2$ of the system (2.10) is the minimizer of the given quadratic function over $V(y^0, y^1 - y^0, \ldots, y^j - y^0)$.

Moreover, when $\nabla f(y^0)$ belongs to $[y^1 - y^0, \ldots, y^j - y^0]$ and the subspace $[\nabla f(y^1) - \nabla f(y^0), \ldots, \nabla f(y^j) - \nabla f(y^0)]$ coincides with $[y^1 - y^0, \ldots, y^j - y^0]$, then x^+ is equal to x^*.

Proof. The first part follows from $x^+ = y^0 + (y^j - y^0)/2$ and the fact that $\nabla f(x^+) = (\nabla f(y^0) + \nabla f(y^j))/2$ is orthogonal to $[y^1 - y^0, \ldots, y^j - y^0]$. This can be deduced from the fact that for every $i = 1, \ldots, j$ $\langle \nabla f(y^j) + \nabla f(y^0), y^i - y^0 \rangle = 0$, using (2.7) and (2.3).

The second part follows from the relation $-\nabla f(y^0) = \sum_{i=1}{}^j \mu_i (\nabla f(y^i) - \nabla f(y^0))$, since by hypothesis the subspaces $[y^1 - y^0, \ldots, y^j - y^0]$ and $[\nabla f(y^1) - \nabla f(y^0), \ldots, \nabla f(y^j) - \nabla f(y^0)]$ are equal. Then, from $x^* - y^0 = H^{-1}(-\nabla f(y^0))$, it follows $x^* - y^0 = \sum_{i=1}{}^j \mu_i (y^i - y^0)$. Hence, $x^* \in V(y^0, y^1 - y^0, \ldots, y^j - y^0)$ which implies that is equal to x^+.■

From the previous results, it follows that if $\nabla f(y^0) \in [y^1 - y^0, \ldots, y^j - y^0]$ and the procedure defined by (2.6) leads to a point y^j for which the gradients $\{\nabla f(y^0), \ldots, \nabla f(y^j)\}$ are linearly dependent, then $\nabla f(y^j) = -\nabla f(y^0)$. Thus, in this case x^*, is the intersection of the supporting hyperplanes to the epigraph at the points $(y^i, f(y^0))$, coincides with the midpoint $x^+ = (y^j + y^0)/2$. Moreover, since a convex combination of the gradients $\{\nabla f(y^i)\}_{i=0}^j$ exists equal to zero, x^* satisfies

$$x^* = argmin(max\{\nabla f(y^i)^T(x - y^i)\ , i = 0, 1, \ldots, j\}).$$

The geometrical interpretation and the conclusions given for quadratic functions above were the basis for extending the procedure described in (2.6) to quasiconvex functions with bounded level sets. Like in the previous particular case we shall choose suitable points on the level set of each iterate and the descent direction will be obtained by solving a linear system. When that systems is underdetermined we adopt the closest solution to the current iterate.

Denoting $y^0 = x_c$, let us define from it points $y^1, \ldots, y^j$ $j \leq n$ on $C(x_c)$. We shall prove, in this more general framework, that the mid-point $(y^j + y^0)/2$ is a solution of the system

$$\{(g^i)^T (x - y^i) = \nabla f(y^0)^T (x - y^0), \ \ i = 1, \ldots, j\}, \tag{2.12}$$

where y^i, and $g^i = \alpha_i \nabla f(y^i)$ for a suitable scalar α_i are formally defined below in Algorithm 2.1.

For each point y^i determined by the procedure in $L(x_c)$, we consider a quadratic approximating function $Q_i(x)$ which interpolates the values of f at the points y^0 and y^i. We define the gradient g^0 in y^0 coincident with $\nabla f(y^0)$, and the gradient g^i in y^i

as the scaled $\alpha_i \nabla f(y^i)$. The scalar α_i allows to satisfy the requirement for quadratic functions $(g^i)^T(y^i - y^0) = (g^0)^T(y^0 - y^i)$, when $Q_i(y^0) = Q_i(y^i)$

The ith equation of (2.12) comes from considering the intersection of the supporting hyperplanes of Q_i in $(y^0, f(y^0))$ and $(y^i, f(y^0))$ respectivelly. Such intersection contains the minimizer of Q_i. Thus, the solution $(y^0 + y^j)/2$ to the proposed system (2.12) is in the intersection of the approximating hyperplanes defined before.

2.1 A strategy for determining the points on the level surface

In order to validate the algorithm for obtaining the points $y^i \in C(x_c)$, $i = 1, \ldots, j$, the following lemmas are required.

Lemma 2.3: Given $y \in C(x_c)$, $z \in L(x_c)$, such that $y \neq z$, then the following inequalities hold:

$$\nabla f(y)^T(z - y) \leq 0 \tag{2.13}$$

$$(\nabla f(z) - \nabla f(y))^T(z - y) \geq 0, \quad if\ z \in C(x_c) \tag{2.14}$$

Proof. (2.13) and (2.14) follow from the properties of $f(x)$ (Crouzeix and Ferland [5], Greenberg and Pierskalla [14]). ∎

In particular, if the segment joining y and z contains a point ω such that $f(\omega) < f(x_c)$, the following result holds (Rockafellar [18]):

Lemma 2.4: Given $y,\ z \in C(x_c)$, $y \neq z$, with $\nabla f(y) \neq 0$, if $\omega = \lambda y + (1 - \lambda)z$, $0 < \lambda < 1$, exists such that $f(\omega) < f(x_c)$, then $\nabla f(y)^T(z - y) < 0$.

Proof. By Lemma 2.3, we know that $\nabla f(y)^T(z - y) \leq 0$. Suppose now that $\nabla f(y)^T(z - y) = 0$. From hypotheses about ω and f, it follows that there exists $\epsilon > 0$ such that, for all ω' with $\|\omega - \omega'\| < \epsilon$, where $\omega' = \omega + \delta \nabla f(y)$, $\delta > 0$, they satisfy $f(\omega') < f(x_c)$.

Clearly, the assumption $\nabla f(y)^T(z - y) = 0$ and the definition of ω' implies $\nabla f(y)^T(\omega' - y) > 0$, which contradicts Lemma 2.3. ∎

Now we can define the algorithm for determining the points on $C(x_c)$ which will define the search direction as the solution of a finite difference system of equations similar to (2.2).

Let us define $y^0 = x_c$, $g^0 = \nabla f(x_c)$. The definition of the points y^i, $i \geq 1$ will be done sequentially by means of linearly independent directions $p^0, \ldots, p^{i-1}$ in such a way that $y^i = y^{i-1} + h_{i-1}p^{i-1}$, satisfying $f(y^i) = f(y^{i-1})$ for $1 \leq i \leq n$.

We shall take $p^0 = -g^0$ as the first direction. The directions p^i at the following points y^i used for determining y^{i+1} will be taken orthogonal to the differences $g^1 - g^0, \ldots, g^i - g^0$ a condition identical to (2.8). Therefore, we shall write

$p^i = P^i(-g^0)$ where P^i is the orthogonal projector onto the orthogonal subspace to $[g^1 - g^0, \ldots, g^i - g^0]$. From this definition when for some $j \le n$ is $p^j = 0$, it follows that $g^0 \in [g^1 - g^0, \ldots, g^j - g^0]$. Furthermore, as we shall justify later, in such a case the point $x^+ = y^0 + (y^j - x_c)/2$ is the solution of the finite difference system of equations and corresponds to the intersection of the supporting hyperplanes at the points y^i.

In order to formalize the algorithm for computing the points on $C(x_c)$, the following notation will be used:

A_i the matrix whose rows are $\{(g^j - g^0)^T, j = 1, \ldots, i\}$,

$R({A_i}^T)$ the subspace spanned by the columns of ${A_i}^T$,

$P^0 = I_n$, I_n $n \times n$ identity matrix.

P^i is formally updated by([2], [9])

$$P^i = P^{i-1} - P^{i-1}(g^i - g^0)(g^i - g^0)^T P^{i-1}/(g^i - g^0)^T P^{i-1}(g^i - g^0) \qquad (2.15)$$

In the numerical implementation, the standard orthogonalization procedure is used ([3]).

From the definition of p^i, as a consequence of $P^i(g^i - g^0) = 0$, we can use either $p^i = P^i(-g^0)$ or $P^i(-g^i)$.

Algorithm 2.1 :
Given x_c, $\nabla f(x_c) \neq 0$,
Step 1:
Define $y^0 = x_c$, $g^0 = \nabla f(x_c)$, $i = 0$.
Step2:
If $P^i(-g^i) = 0$, define $j = i$. Stop.
Else,
Step 3:
Take $p^i = P^i(-g^i)$; $y^{i+1} = y^i + h_i p^i$, such that $f(y^{i+1}) = f(y^i)$.
If $\nabla f(y^{i+1}) \neq 0$ define $g^{i+1} = \alpha_{i+1} \nabla f(y^{i+1})$ satisfying
$$\alpha_{i+1} \nabla f(y^{i+1})^T (y^{i+1} - y^0) = \nabla f(y^0)^T (y^0 - y^{i+1}) \ .$$
Else, $g^{i+1} = 0$.
Update P^{i+1}.
i=i+1; go to Step 2. •

The Algorithm 2.1 stops after having computed points $y^1, \ldots, y^j$, $j \le n$ on $C(x_c)$. The index j is the first index for which $P^j(-g^j) = 0$, meaning that $g^j \in R({A_j}^T)$. In the Appendix an implementation of the Step 3 is described, which computes approximate points on the level set while keeping the essential properties of the resulting descent direction.

The aim of proving the following results is twofold: first, to show the directions p^i are well defined, and second, to derive some suitable properties arising from the way in which the auxiliary points are defined.

Lemma 2.5: If $P^i(-g^i) \neq 0$ and if y^{i+1} is defined as in Step 3 of Algorithm 2.1, then

a) $(g^0)^T(y^{i+1} - y^0) < 0$

b) $\nabla f(y^{i+1})^T(y^{i+1} - y^0) > 0$, if $\nabla f(y^{i+1}) \neq 0$.

Proof. Since $y^{i+1} - y^0 = \sum_{l=0}^{i} h_l p^l = \sum_{l=0}^{i} h_l P^l(-g^l)$, then $(g^0)^T(y^{i+1} - y^0) < 0$ considering that $P^l(-g^l) = P^l(-g^0)$.

From a), $\nabla f(y^{i+1}) \neq 0$ and Lemma 2.4, it follows b). ■

Remark 2.1: The coefficients α_i in Step 3 of Algorithm 2.1 are well defined as a consequence of Lemma 2.5 b).

Theorem 2.2: If the Algorithm 2.1 does not terminate at y^i (i.e. $p^i \neq 0$) then

a) $\{g^0, g^1, \ldots, g^i\}$ are linearly independent.

b) $[g^0, g^1, \ldots, g^i] = [p^0, p^1, \ldots, p^i]$.

Moreover, if y^{i+1}, g^{i+1} are defined as in Algorithm 2.1, then

c)$\{g^{i+1} - g^0, g^i - g^0, \ldots, g^1 - g^0\}$ are linearly independent.

Proof. We prove a), b) and c) simultaneously by induction. Clearly, a) and b) hold for $i = 0$.

To prove c), we consider $\|g^1 - g^0\|^2 = \|g^1\|^2 + \|g^0\|^2 + 2(g^1)^T(-g^0) \geq \|g^0\|^2$, as a consequence of (2.13). Since $g^0 \neq 0$, then $g^1 - g^0 \neq 0$.

Now, assuming a), b) and c) are valid for i, we will prove they also hold for *i+1*.

We have, $p^{i+1} = P^{i+1}(-g^{i+1}) \neq 0$, then the subspace

$[g^{i+1} - g^0, g^i - g^0, \ldots, g^1 - g^0, g^{i+1}] = [g^{i+1} - g^0, g^i - g^0, \ldots, g^1 - g^0, p^{i+1}]$ has rank *i+2*. Since $[g^{i+1} - g^0, g^i - g^0, \ldots, g^1 - g^0, g^{i+1}]$ is included in

$[g^{i+1}, g^i, \ldots, g^1, g^0]$, we obtain a).

From the definition, $p^{i+1} \in [g^{i+1} - g^0, \ldots, g^1 - g^0, g^{i+1}] = [g^{i+1}, g^i, \ldots, g^1, g^0]$. Furthermore, $p^{i+1} \notin [g^0, \ldots, g^i] = [p^0, \ldots, p^i]$, since otherwise $\|p^{i+1}\| = \|p^i\|$ which contradicts that $P^i(g^{i+1} - g^0) \neq 0$ (the inductive hypothesis on c) guarantees that $g^{i+1} - g^0$ is linearly independent of $\{g^i - g^0, \ldots, g^1 - g^0\}$). Thus $[g^0, g^1, \ldots, g^{i+1}] = [p^0, p^1, \ldots, p^{i+1}]$.

To prove c), since $p^{i+1} = P^{i+1}(-g^0) \neq 0$, by Lemma 2.4 we get $(g^{i+2} - g^0)^T p^{i+1} > 0$. Thus, we conclude that $\{g^l - g^0\}_{l=1}^{i+2}$ are linearly independent.■

Remark 2.2: As a consequence of the proof of Theorem 2.2, if $P^i(-g^i) \neq 0$ for $i \geq 1$, the subspace $[g^i - g^0, \ldots, g^1 - g^0, p^i]$ is coincident with the subspace $[g^i, \ldots, g^1, g^0]$, and has rank $i+1$.

Corollary 2.1: If j, $1 \leq j \leq n$, is the first index for which $P^j(-g^j) = 0$ then $[g^1 - g^0, \ldots, g^j - g^0] = [p^0, \ldots, p^{j-1}]$.

Proof. Since $g^j \in [g^1 - g^0, \ldots, g^j - g^0]$, then $g^0, g^1, \ldots, g^{j-1}$ are in that subspace. Therefore, by Theorem 2.2 we conclude the proof.■

2.2 Defining the search direction

We can now define the search direction from x_c and prove its properties.

Lemma 2.6: When Algorithm 2.1 stops, $y^j - x_c$ is a descent direction.

Proof. It follows from Lemma 2.5 a) for $i = j - 1$.■

Let us call $S_j, j \leq n$, the matrix whose rows are $(y^1 - y^0)^T, \ldots, (y^j - y^0)^T$.

Lemma 2.7: When the Algorithm 2.1 stops at a point y^j such that $\nabla f(y^j) \neq 0$, then $\tilde{x} = (y^j + x_c)/2$ is a solution of the system

$$A_j(x - x_c) = S_j(-\nabla f(x_c)).$$

Furthermore, $d_c = \tilde{x} - x_c$ is the minimum norm solution to the system

$$A_j d = S_j(-\nabla f(x_c)) \ .$$

Proof. The point $\tilde{x}$ is a solution because $d_c = \tilde{x} - x_c = (y^j - x_c)/2$ satisfies each one of the equations since $y^j - x_c = (y^j - y^i) + (y^i - x_c)$ and $(g^i - g^0)^T(y^j - x_c)/2 = (g^i - g^0)^T(y^i - x_c)/2 = -{g^0}^T(y^i - x_c)$ as a consequence of the definitions of p^i and g^i.

Since $d_c = \tilde{x} - x_c = (y^j - x_c)/2$, $d_c \in R({S_j}^T)$ and also $d_c \in R({A_j}^T)$ because of Corollary 2.1, then it is the minimum norm solution to the system. Therefore

$d_c = {A_j}^{\dagger} S_j(-\nabla f(x_c))$.■

We choose the descent direction $d_c = \tilde{x} - x_c = (y^j - x_c)/2$. As a consequence of Lemma 2.7 such direction is the solution of the system (2.12). Moreover, when f is a convex quadratic function d_c is the Newton's direction according to Lemma 2.2.

3 The general algorithm and global convergence

Now we are able to define a general minimization algorithm for quasiconvex functions under the stated hypotheses in Section 2, using the descent direction previously introduced, and to prove that it is globally convergent.

Algorithm 3.1

Given the starting point x_0, $f(x_0)$, $\nabla f(x_0)$ and the parameters m_1, m_2 such that $0 < m_1 < 1/2$, $m_1 < m_2 < 1$. Set $k := 0$

Step 1. If "convergence" stop. Else,

Step 2. Define $y^0, \dots, y^j \in C(x_k)$, $1 \le j \le n$ using Algorithm 2.1.

Step 3. Define $d_k = (y^j - x_k)/2$ and perform a line search along it, starting with $\lambda_k = 1$, until finding a value of $\lambda_k > 0$ such that

$$f(x_k + \lambda_k d_k) \le f(x_k) + m_1 \lambda_k \nabla f(x_k)^T d_k \tag{3.1}$$

$$\nabla f(x_k + \lambda_k d_k)^T d_k \ge m_2 \lambda_k \nabla f(x_k)^T d_k \tag{3.2}$$

Define $x_{k+1} = x_k + \lambda_k d_k$, and compute $\nabla f(x_{k+1})$,
$k := k + 1$ and go to Step 1. •

Remark 3.1: Under the hypotheses stated for f some $\lambda_k > 0$ satisfying both (3.1) and (3.2) always exist due to the fact that $f(y^j) = f(x_k)$([9]).

We shall give in the following some results required for proving the global convergence of Algorithm 3.1.

Lemma 3.1: At any iteration k of Algorithm 3.1, there exists a constant $c > 0$ such that for y^i, $i \le j$, obtained in Step 2 the following relations hold:

$$-\nabla f(x_k)^T (y^i - x_k) \ \le\ (1/2)\, c \, \|y^i - x_k\|^2 \tag{3.3}$$

$$\|y^i - x_k\| \ \ge\ (2/c)\, \|\nabla f(x_k)\| \tag{3.4}$$

Proof. Taking into account that $f(y^i) = f(x_k)$ and Taylor's expansion, we have $f(y^i) = f(x_k) + \nabla f(x_k)^T(y^i - x_k) + 1/2(y^i - x_k)^T\bar{H}(y^i - x_k)$, where $\bar{H}$ denotes the Hessian matrix at an intermediate point between y^i and x_k. Thus, $-\nabla f(x_k)^T(y^i - x_k) = (1/2)(y^i - x_k)^T\bar{H}(y^i - x_k)$, from which we get $-\nabla f(x_k)^T(y^i - x_k) \leq (1/2)c\|y^i - x_k\|^2$ considering that $f \in C^2(L(x_0))$ and the compactness of $L(x_0)$. This proves (3.3).

By (3.3) and the definition of p^0, $\|y^1 - x_k\| \geq (2/c)\|\nabla f(x_k)\|$.

Furthermore, for $1 < i \leq j$, we have that $\|y^i - x_k\| > \|y^1 - x_k\|$ due to $\|y^i - x_k\|^2 = \|y^i - y^1\|^2 + \|y^1 - x_k\|^2 + 2(y^i - y^1)^T(y^1 - x_k)$ and the fact that $(y^i - y^1)^T(y^1 - x_k) > 0$. Therefore, (3.4) follows.■

Theorem 3.1: Under the hypotheses stated for f, the sequence $\{x_k\}$ given by Algorithm 3.1 is well defined and $\lim_{k\longrightarrow\infty}\|\nabla f(x_k)\| = 0$

Proof. From Lemma 2.6 we know that d_k is a descent direction. As a consequence of the hypotheses some λ_k exist satisfying (3.1) and (3.2).

Moreover, since the gradient is Lipschitz continuous in $L(x_0)$ we get that $\lim_{k\longrightarrow\infty}(\nabla f(x_k)^T d_k)/\|d_k\| = 0$ (Wolfe [19], [20], Zoutendijk [21]).

From the definition of d_k in Algorithm 3.1, we know that $d_k = (y^j - x_k)/2$, where $f(y^j) = f(x_k)$. Since $\nabla f(x_k)^T(y^j - x_k) \leq \nabla f(x_k)^T(y^1 - x_k) = -\|\nabla f(x_k)\|\|y^1 - x_k\|$, using (3.4) and the compactness of $L(x_k)$, we get $\nabla f(x_k)^T d_k/\|d_k\| \leq -\gamma\|\nabla f(x_k)\|^2$ with $\gamma = 1/(cM)$ and $\|d_k\| \leq M$. From that, it follows that $\lim_{k\longrightarrow\infty}\|\nabla f(x_k)\| = 0$. ■

Since Algorithm 3.1 defines a descent method and $L(x_0)$ is a compact set, it turns out that the sequence $\{x_k\}$ has limit points which are stationary ones of $f(x)$. Therefore, we obtain

Theorem 3.2: Under the hypotheses stated for f, the sequence $\{x_k\}$ given by Algorithm 3.1 has at least a limit point, and every limit point is a stationary one.

4 Local quadratic convergence

We have proved Algorithm 3.1 is globally convergent in the sense that every limit point of the sequence $\{x_k\}$ must satisfy the first order stationary condition.

We shall prove here that under the hypotheses stated in Section 3, the sequence $\{x_k\}$ generated using Algorithm 3.1 is locally convergent to stationary points x^* at which the Hessian matrix is positive definite.

If we also assume the Hessian is Lipschitz continuous over Ω, that is a constant $L > 0$ exists such that for all $x, y \in \Omega$

$$||\nabla^2 f(x) - \nabla^2 f(y)|| \leq L\,||x - y||, \tag{4.1}$$

then we will prove an index k_1 exists such that $\{x_k\}$ satisfies the Wolfe conditions (3.1) and (3.2) with $\lambda_k = 1$ for $k > k_1$, and moreover the sequence is quadratically convergent. This key result will be proved in Theorem 4.2.

For that purpose we shall state in the following some results for the sequence x_k given by Algorithm 3.1.

Lemma 4.1: If $\lim_{k \longrightarrow \infty} x_k = x^*$ where $H(x^*) > 0$, then an index k_0 exists such that for every $k > k_0$

a) f is uniformly convex on $L(x_k)$.

b) $||d_k|| \leq (1/m)||\nabla f(x_k)||$.

Proof. Since $H(x^*) > 0$, let us denote by $2m > 0$ its least eigenvalue. Then $\epsilon > 0$ exists such that for all x with $||x - x^*|| \leq \epsilon$, the least eigenvalue of $H(x)$ is greater or equal than m. Hence, in such domain

$$m||y||^2 \leq y^T H(x) y \leq c||y||^2, \;\; for\ y \in \Re^n. \tag{4.2}$$

Taking into account that $x' = argmin(f)$ in $||x - x^*|| = \epsilon$, then an ϵ' exists, $0 < \epsilon' \leq \epsilon$, such that for all x, $||x - x^*|| \leq \epsilon'$, $f(x) \leq f(x')$ hold. This follows from the hypotheses and considering that

$$f(x) - f(x^*) = (1/2)(x - x^*)^T \hat{H}(x - x^*) \leq (1/2)c||x - x^*||^2 \leq f(x') - f(x^*),$$

where $\hat{H}$ is the Hessian at the corresponding intermediate point, and defining $\epsilon' = min\{\epsilon, (2(f(x') - f(x))/c)^{1/2}\}$.

Since $\lim_{k \longrightarrow \infty} x_k = x^*$, k_0 exists such that for all $k > k_0$, $||x_k - x^*|| \leq \epsilon'$; then $f(x_k) \leq f(x')$. Hence, for all $k > k_0$, $L(x_k)$ is included in
$\{x : ||x - x^*|| \leq \epsilon\}$ where f is uniformly convex.

b) By definition given in Step 3 of the Algorithm 3.1 $d_k = (y^j - x_k)/2$, y^j being the point in $L(x_k)$ with $f(y^j) = f(x_k)$ given by Algorithm 2.1. Then using the Taylor's expansion of $f(y^j)$ at x_k we get

$$-\nabla f(x_k)^T (y^j - x_k) = 1/2(y^j - x_k)^T \hat{H}(y^j - x_k),$$

where Ĥ is the Hessian at the corresponding intermediate point.

Using a), for all $k > k_0$ is

$$-\nabla f(x_k)^T(y^j - x_k)/2 \geq m\|(y^j - x_k)/2\|^2. \tag{4.3}$$

Thus, for $k > k_0$ it follows that $\|d_k\| \leq (1/m)\|\nabla f(x_k)\|$.■

Lemma 4.2: If $\lim_{k\longrightarrow\infty} x_k = x^*$ and $H(x^*) > 0$ and $x_{k+1} = x_k + \lambda_k d_k$ as defined by Algorithm 3.1 , λ_k satisfying (3.1) and (3.2), then there is an index $k_1 \geq k_0$ such that for every $k > k_1$, $\lambda_k = 1$ is admissible.

Proof. From Lemma 4.1, for all $k > k_0$, $\|d_k\| \leq (1/m)\ \|\nabla f(x_k)\|$ and

$$\nabla f(x_k)^T d_k \leq -m\|d_k\|^2 \tag{4.4}$$

The choice $d_k = (y^j - x_k)/2$ with $f(y^j) = f(x_k)$, implies that $d_k^T\nabla f(x_k) + d_k^T H_k d_k = o(\|d_k\|^2)$. Using the Lipschitz continuity of the Hessian, and $\|d_k\| \leq 1/m\|\nabla f(x_k)\|$ there is an index $k' \geq k_0$ such that $\|d_k^T(\nabla f(x_k) + H_k d_k)\| \leq \delta\|d_k\|^2$ and $\|\nabla f(x_k + d_k) - \nabla f(x_k) - H_k d_k\| \leq \delta\|d_k\|$ for all $k > k'$, where $\delta = m\ min(1/2 - m_1, m_2/2)$. Furthermore, there is an index $k_1 \geq k'$ such that for all $k > k_1$, $\|d_k\| \leq \delta/L$. Then for $k > k_1$, we get that $f(x_k + d_k) - f(x_k) = \nabla f(x_k)^T d_k + 1/2 {d_k}^T H_k d_k + 1/2 {d_k}^T(\hat{H} - H_k)d_k \leq 1/2\nabla f(x_k)^T d_k + \delta\|d_k\|^2$. Therefore,

$$f(x_k + d_k) - f(x_k) - m_1\nabla f(x_k)^T d_k \leq (1/2 - m_1)\nabla f(x_k)^T d_k + \delta\|d_k\|^2$$

from which, using (4.4) for $k > k_1$ we get that $\lambda_k = 1$ satisfies (3.1).

Analogously, and using the same arguments, we deduce that for $k > k_1$

$$\nabla f(x_k + d_k)^T d_k = (\nabla f(x_k + d_k) - \nabla f(x_k) - H_k d_k)^T d_k + (\nabla f(x_k) + H_k d_k)^T d_k$$
$$\geq -\delta\|d_k\|^2 - \delta\|d_k\|^2$$

Hence, using (4.4) it follows that for $k > k_1$

$$\nabla f(x_k + d_k)^T d_k - m_2\nabla f(x_k)^T d_k \geq (mm_2 - 2\delta)\|d_k\|^2 \geq 0.$$

Therefore $\lambda_k = 1$ satisfies (3.2).■

It is known that for quasiconvex functions every strict local minimizer is the unique global minimizer, a fact we shall use in the following theorem.

Theorem 4.1: Under the hypotheses stated in Section 3, if $x^* \in \Omega$ is a stationary point where $H(x^*) > 0$, then $\epsilon > 0$ exists such that for all x_0 satisfying $\|x_0 - x^*\| \leq \epsilon$ the sequence $\{x_k\}$ generated by Algorithm 3.1 converges to x^*.

Proof. As a consequence of the hypotheses on x^*, and using the same arguments given in the proof of Lemma 4.1, there is an $\epsilon > 0$ such that for all x_0, $\|x_0 - x^*\| \leq \epsilon$, f is strictly convex on the level set $L(x_0)$. The result follows straightforwardly from the compactness of $L(x_0)$, the uniqueness of x^* in $L(x_0)$, and the fact that d_k is a descent direction. ■

The purpose of the following results is to prove the local quadratic convergence of the sequence defined by Algorithm 3.1, under the hypotheses stated at the beginning of this Section when x_k tends to x^* with $H(x^*) > 0$.

Lemma 4.3: If $\lim_{k \longrightarrow \infty} x_k = x^*$ and $H(x^*) > 0$, then k_1 exists such that for all $k > k_1$, $x_{k+1} = x_k + d_k$ and $\|\nabla f(x_{k+1})\| = O(\|\nabla f(x_k)\|^2)$.

Proof. It follows from Lemma 4.1, 4.2 and Lemma B.6. ■

Theorem 4.2: If $\lim_{k \longrightarrow \infty} x_k = x^*$ and $H(x^*) > 0$, then under the hypotheses stated above, the rate of convergence is quadratic.

Proof. From Lemma 4.3 , there exists an index k_1 such that for all $k > k_1$, $\|\nabla f(x_{k+1})\| = O(\|\nabla f(x_k)\|^2)$.

Since $\|x_{k+1} - x^*\| = O(\|\nabla f(x_{k+1})\|)$, then $\|x_{k+1} - x^*\| = O(\|x_k - x^*\|^2)$ and thus quadratic convergence follows. ■

5 Quadratic global convergence for pseudo-convex functions

The local quadratic convergence of the sequence given by Algorithm 3.1 to stationary points x^* at which the Hessian matrices are nonsingular has been proved in Section 4 under some restrictive hypotheses.

Let us consider now a particular subclass of the quasiconvex functions, the pseudoconvex functions, under the same hypotheses of Section 4.

Definition 5.1: f is said to be pseudoconvex if for each $x_1,\ x_2 \in \Omega$ such that $f(x_2) < f(x_1)$ then $\nabla f(x_1)^T(x_2 - x_1) < 0$.

This function is characterized by the fact that every stationary point is a global minimizer ([1]). Using the same hypotheses of Section 4 we shall be able to prove the global quadratic convergence of the sequence given by Algorithm 3.1 in the next theorem.

Theorem 5.1: Under the stated hypotheses, and if $x^* \in \Omega$ is a stationary point where $H(x^*) > 0$, then the sequence $\{x_k\}$ generated by Algorithm 3.1 converges quadratically to x^*, the global minimizer of $f(x)$.

Proof. Since $f(x)$ is a pseudoconvex function and has the stationary point x^* such that $H(x^*) > 0$, x^* must be the unique stationary point of $f(x)$ on $L(x_0)$.

Moreover, due to the compactness of $L(x_0)$ and the fact x^* is the only possible limit point of $\{x_k\}$, we have that $\lim_{k\to\infty} x_k = x^*$.

On the other hand, using the same arguments of Theorem 4.2 we can prove that the sequence $\{x_k\}$ generated by Algorithm 3.1 converges
quadratically to x^*.■

6 Numerical Experiences

The aim of the following experiences is just to assess the qualitative behavior of the search direction introduced in this paper. We have performed those experiences using only three convex functions. Two of them are the Penalty I and the Variably dimensioned functions ([12],[16]) and the third is an extension of the one presented in [4] in order to use higher dimensions and for showing the direction's behavior when the level sets correspond to ill-conditioned problems. We have compared the number of iterations required to achieve convergence using different Newton type algorithms, like Newton with trust regions, TRON of Chih-Jen Lin and Jorge J. Moré ([15]), the Truncated Newton TN of Stephen G. Nash ([17]), and the NMTR of the Minpack-2 Project headed by J.J. Moré. The executable versions are TRON as available in http://www.mcs.anl.gov/~more, TN from NETLIB, and NMTR is the version run in the NEOS SERVER of Argonne National Laboratory and Northwestern University (http://www-neos.mcs.anl.gov). We have also tested the results with our implementation of Newton's method (NW) with line searches based on LINPACK routines and the subroutine GSRCH developed for M.J.D. Powell, using the standard stopping condition $\|\nabla f(x_k + \lambda_k d_k)^T d_k\| \leq grhtol\|\nabla f(x_k)^T d_k\|$ with $grhtol = 0.1$.

An experimental Fortran 90 program in double precision was written implementing Algorithm 3.1 and which uses in Step 2 the computation of the approximate points on the level set for Algorithm A.1 as described in the Appendix. The Step 2 is done using an implementation of Algorithm A.1 with parameters $\epsilon_1 = 10^{-5}$, $\epsilon_2 = 10^{-3}$, $\alpha = 10^{-4}$, $\tau_1 = 10^{-4}$, $\tau_2 = 10^{-4}$, and $\beta = 3$. A point y^j is accepted if $\|P^j(-g^j)\| \leq 10^{-6}\|g^0\|$ (see Algorithm A.2 in the Appendix). Projections were calculated using the modified Gram-Schmidt algorithm [3]. The line search of Step 3 is also performed with GSRCH using $grhtol = 0.1$. The stopping criterion for Algorithm 3.1(QSI) and Newton's(NW) is $\|\nabla f(x)\|_\infty \leq \epsilon$ with $\epsilon = 10^{-5}$. For TN and NMRT we used the default values; in TRON the condition $\|\nabla f(x)\|_2 \leq gtol\|\nabla f(x^0)\|_2$, adapts gtol for obtaining a point with a lower values of the gradient's norm, indicating that in the Table by TRON(gtol) (gtol=1.d-5, 1.d-10, 1.d-16 or 1.d-20) according to the starting gradient for each problem.

The definition of the Extended Convex function ([4]) with variable para-meter σ

is as follows: Given $x \in \Re^n$, A *and* $D \in \Re^{n\times n}$

$$f(x) = \tfrac{1}{2}x^T Dx + \sigma(\tfrac{1}{2}x^T Ax)^2 \quad \text{where} \quad A = \begin{bmatrix} A_1 & 0 \\ 0 & I \end{bmatrix}, \quad A_1 = \begin{bmatrix} 5 & 1 \\ 1 & 3 \end{bmatrix}; \quad \text{and}$$

D was defined as : (a) D = $diag(1,\ldots,1)$, or (b) D = $diag(1,\ldots,i^2,\ldots,n^2)$. The starting point is $x_0 = (-1,1,\ldots,-1,1)$.

The Penalty I function ([12]) was run from two different starting points: (a)$x_0 = i$, and (b) $x_0 = (-1)^{i+1}$ for $i = 1,\ldots,n$.

First, we will compare for $n = 100$ the function reduction obtained in each iteration with the codes QSI, NW(Newton) and TRON(the last two use second derivatives), together with the comparison with the approximation to the optimal solution of each problem. Those results are given for each problem in the following tables where it is possible to see that in all cases the direction used in QSI leads to a fast descent and approximation to the solution although we were using just an approximation to the real direction defined in Section 3. The same tendency is observed in higher dimensions. In each of the following tables $f_k - f_*$ gives the decrease obtained in each iteration and the approximation to the solution is given by $\|x_k - x^*\|_2$, for the least number of iterations needed for achieving convergence,

	QSI		NW		TRON	
iter	$f_k - f_*$	$\|x_k - x^*\|$	$f_k - f_*$	$\|x_k - x^*\|$	$f_k - f_*$	$\|x_k - x^*\|$
0	2.70d+05	1.00d+01	2.70d+05	1.00d+01	2.70d+05	1.00d+01
1	3.86d-08	2.78d-04	1.12d+04	4.52d+00	5.34d+04	6.67d+00
2	1.68d-18	1.83d-09	4.64d+02	2.04d+00	5.14d+03	3.64d+00

Extended Convex function (a), $\sigma = 100$.

	QSI		NW		TRON	
iter	$f_k - f_*$	$\|x_k - x^*\|$	$f_k - f_*$	$\|x_k - x^*\|$	$f_k - f_*$	$\|x_k - x^*\|$
0	3.20d+02	1.00d+01	3.20d+02	1.00d+01	3.20d+02	1.00d+01
1	7.01d-07	1.18d-03	1.58d+01	4.06d+00	6.82d+01	6.45d+00
2	1.28d-18	1.60d-09	0.49d+00	0.96d+00	1.50d+01	3.99d+00

Extended Convex function (a), $\sigma = 0.1$.

	QSI		NW		TRON	
iter	$f_k - f_*$	$\|x_k - x^*\|$	$f_k - f_*$	$\|x_k - x^*\|$	$f_k - f_*$	$\|x_k - x^*\|$
0	1.69d+05	1.00d=01	1.69d+05	1.00d+01	1.69d+05	1.00d+01
1	8.17d-02	3.90d-02	4.29d+01	2.66d+00	4.66d+01	2.64d+00
2	1.21d-13	3.85d-07	1.82d=00	1.13d=00	2.79d+00	1.35d+00

Extended Convex function (b), $\sigma = 0.1$.

	QSI		NW		TRON	
iter	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$
0	4.39d+05.	1.00d+01	4.3d+05	1.00d+00	4.39d+05	1.00d+00
1	8.58d-02	1.31d-02	1.64d+04	4.02d+00	8.56d+04	6.20d+00
2	1.18d-06	3.96d-04	8.72d+02	1.84d+00	5.36d+03	2.76d+00
3	2.91d-08	1.79d-05	6.07d+01	0.86d+00	1.51d+03	2.02d+00
4	1.09d-09	1.02d-06	4.76d=00	0.41d=00	4.92d+02	1.50d+00
5	2.27d-13	1.59d-08	0.36d+00	0.18d+00	6.58d+01	0.85d=00
6	1.92d-16	4.30d-10	2.63d-02	8.08d-02	6.20d+00	0.49d+00

Extended Convex function (b), $\sigma = 100$

	QSI		NW		TRON	
iter	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$
0	1.31d+14	5.82d=00	1.31d+14	5.82d+00	1.31d+14	5.82d+00
1	2.31d-02	2.58d-04	5.43d+12	2.62d+00	5.66d+09	0.47d+00
2	4.55d-16	3.67d-11	2.25d+11	1.18d+00	7.24d+08	0.28d+00

Variably Dimensioned.

	QSI		NW		TRON	
iter	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$
0	1.15d+08	5.74d+02	1.15d+08	5.74d+02	1.15d+08	5.74d+02
1	9.23d+01	8.68d+00	4.78d+06	2.54d+02	9.23d+01	8.68d+00
2	2.69d-03	0.430d-01	2.01d+05	1.10d+02	2.59d+00	1.31
3	1.51d-09	0.25d-04	8.71d+03	4.38d+01	1.82d-01	0.36d+00
4	5.80d-10	1.06d-09	3.09d+02	1.30d+01	1.25d-02	0.9d-01

Penalty I (a)

	QSI		NW		TRON	
iter	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$	$f_k - f_*$	$\|\|x_k - x^*\|\|$
0	2.03d+02	1.33d+01	2.03d+02	1.33d+01	2.03d+02	1.33d+01
1	3.06d+00	1.48d+00	3.59d+00	1.63d+00	1.81d-01	0.35d+00
2	0.92d-03	0.25d-01	0.22d-03	0.13d-01	0.59d-03	0.02d+00
3	5.99D-10	0.36d-05	5.82d-10	0.13d-05	0.40d-04	0.01d+00
4	5.800d-10	0.43d-09	5.800d-10	0.10d-06	0.278d-5	0.138d-02

Penalty I (b)

In the last Table it is possible to observe fastest initial descents for NW and TRON with regard to QSI; which differs from what happened in the previous problems. The reason seems to be the use of approximate points in the vicinity of the solution.

For higher dimensions the tables 1-4 use the following notation: in the first column the name of the problem is given. In the next columns under *Method* the name of the program ; *iter* the number of iterations; *nfu* the number of function evaluations; *ngr* the number of gradient evaluations; *gnor* is the l_2 norm of the gradient at the final point, and *fmin* is the best function value. In particular for the methods TN and TRON under *iter* we write between parenthesis the number of CG iterations used. In Table 1 the first row gives the results of Algorithm 3.1(QSI), the second with Newton (NW), the third with TN, the fourth with TRON, and the fifth with NMTR.

Problem	Method	iter	nfu	ngr	gnor	fmin
Extend.	QSI	2	818	407	0.3d-05	0.4d-14
Convex	NW	6	37	37	0.5d-04	0.2 d-11
(b)	TN	23(63)	24	24	0.1d-05	0.5d-16
$\sigma = 0.1$	TRON(-10)	9(8)	9	9	0.1d-10	0.6d-22
	NMTR	8	10	9	0.0d+00	0.0d+00
Extend.	QSI	6	2267	1125	0.2d-4	0.3d-14
Convex	NW	12	73	73	0.1d-04	0.3 d-14
(b)	TN	28(126)	40	40	0.1d-06	0.4d-17
$\sigma = 100$	TRON(-10)	19(18)	19	19	0.5d-04	0.1d-08
	NMTR	17	19	18	0.0d+00	0.0d+00

Table 1: $n = 250$

Those results show that for defining the direction, the method QSI computed almost n points close to the level set and their gradients, something which is computationally expensive. On the other hand they also show that only a few major iterations are needed in spite of approximate points were used. The same situation occurred with ill-conditioned problems using different dimensions. An example is the results of Table 2 absolutely similar to those of Table 1, except for NMTR because its use in NEOS is restricted to $n \leq 250$.

Problem	method	iter	nfu	ngr	gnor	fmin
Extend.	QSI	4	6334	3166	0.1d-04	0.1d-15
Convex	NW	7	43	43	0.5d-08	0.3d-20
(b)	TN	31(143)	32	32	0.5d-06	0.6d-17
$\sigma = 0.1$	TRON(-10)	9(8)	9	9	0.1d-05	0.5d-12
Extend.	QSI	5	6493	3239	0.2d-04	0.3d-15
Convex	NW	14	85	85	0.5d-09	0.5d-24
(b)	TN	38(251)	43	43	0.1d-05	0.1d-16
$\sigma = 100$	TRON(-10)	27(26)	27	27	0.2d-03	0.3d-07

Table 2: $n = 1000$ The output of TRON code also shows the number of Matrix-Vector products, which in this case takes a considerable amount of time because the Hessians are dense.

Contrariwise to what it was observed in the previous tests in regard to the number of inner iterations required for obtaining the search direction, it can be observed that when the ill-conditioned case was excluded, only a few points close to the level set were enough. In those cases, except for the problem Penalty I (b), the number of major iterations needed for achieving convergence is less than for the other algorithms. A substantial saving of CPU time was required, because of fewer gradients were computed in the inner iterations. Table 3 illustrates this situation

Problem	Method	iter	nfu	ngr	gnor	fmin
Extend. Convex (a) $\sigma = 0.1$	QSI	2	22	7	0.3d-06	0.4d-13
	NW	5	32	32	0.9d-10	0.4d-20
	TN	5(12)	9	9	0.1d-09	0.7d-20
	TRON(-5)	9(8)	9	9	0.8d-06	0.3d-12
	MNTR	8	10	9	0.7d-16	0.3d-32
Extend. Convex (a) $\sigma = 100$	QSI	2	25	9	0.5d-09	0.1d-18
	NW	9	55	55	0.2d-05	0.2d-11
	TN	7(17)	14	14	0.2d-08	0.1d-17
	TRON(-10)	19(18)	19	19	0.1d-06	0.6d-14
	MNTR	16	18	17	0.2d-11	0.2d-23
Penalty (a)	QSI	5	39	18	0.1d-06	0.35d+02
	NW	9	55	55	0.2d-07	0.35d+02
	TN	484(1498)	490	490	0.6d-01	0.4d+02
	TRON(-20)	21(21)	22	21	0.5d-2	0.4d+02
	MNTR	15	17	16	0.1d-07	0.4d+02
Penalty (b)	QSI	4	32	14	0.8d-04	0.4d+02
	NW	4	26	26	0.2d-07	0.4d+02
	TN	7(19)	10	10	0.3d-04	0.4d+02
	TRON(-10)	18(18)	19	18	0.7d-02	0.4d+02
	MNTR	4	6	5	0.3d-08	0.4d+02
Variably Dimens.	QSI	2	20	5	0.2d-04	0.1d-16
	NW	80	254	254	0.8d-09	0.7d-25
	TN	9 (16)	26	26	0.1d-05	0.6d-18
	TRON(-16)	20(51)	52	20	0.9d-04	0.1d-09
	MNTR	3976	4000 (F)	3977	0.4d+08	0.4d+05

Table 3: $n = 250$
(F) means that the maximum allowed number of evaluations was reached without getting convergence.

In the previous Table, QSI sistematically required fewer iterations than the other methods. This trend kept unchanged when using different dimensions (n=100, n=300, n=1000). In Table 4 the results for $n = 1000$ are given.

Problem	Method	iter	nfu	ngr	gnor	fmin
Extend.	QSI	2	25	8	0.3d-11	0.6d-23
Convex	NW	6	38	38	0.4d-14	0.7d-29
(a)	TN	8(17)	11	11	0.3d-11	0.3d-23
$\sigma = 0.1$	TRON(-10)	11(10)	11	11	0.1d-10	0.9d-22
Extend.	QSI	2	27	9	0.2d-07	0.2d-15
Convex	NW	10	61	61	0.2d-07	0.4d-15
(a)	TN	15(49)	45	45	0.2d-11	0.1d-22
$\sigma = 100$	TRON(-10)	19(18)	19	19	0.8d-05	0.3d-10
	QSI	5	49	17	0.3d-06	0.3d+03
Penalty	NW	13	167	167	0.1d-02	0.3d+03
(a)	TN	1294(3276)	2877	2877	0.6d-01	0.3d+03
	TRON(-20)	54(56)	57	54	0.8d-01	0.3d+03
	QSI	4	34	15	0.3d-04	0.3d+03
Penalty	NW	5	119	119	0.1d-01	0.3d+03
(b)	TN	11(30)	31	31	0.4d-04	0.3d+03
	TRON(-10)	63(62)	62	62	0.7d-01	0.3d+03
	QSI	2	30	5	0.9d-07	0.7d-23
Variably	NW	981	2131	2131	0.1d-04	0.2d-18
Dimens.	TN	12(17)	33	33	0.1d-05	0.1d-19
	TRON(-20)	19(52)	53	19	0.7d-05	0.5d-11

Table 4: $n = 1000$

The reported results are not conclusive for evaluating the efficiency of Algorithm 3.1 and its implementation QSI using the approximation described in Section A. As we said at the beginning of this section, these experiences had the purpose of illustrating the behavior of the new direction when it was derived by means of a feasible implementation. As shown by the former Tables the new direction led to, except for the Penalty I (b), a better functional decrease and final approximation to x^*.

7 Conclusions

This paper shows that first order information gathered in points close to the level set at a certain iteration, together with the intersection of the corresponding considered hyperplanes, allow us to define an efficient descent direction. Such direction for convex quadratics functions coincides with Newton's, and for non quadratic functions leads to sharp functional reductions. Needless to say, the question of how to obtain a good but cheaper approximation to this new direction keeping its theoretical properties, remains as a research challenge.

Acknowledgment

To Dr. Jorge Nocedal and to the unknown referees for their interesting comments and suggestions.

A Line Search Procedure for obtaining points in step 2 of Algorithm 3.1

The theoretical characterization of Algorithm 2.1 according to which the direction d_k in Step 3 of Algorithm 3.1 is defined, leads naturally to consider its practical implementation. Then, the points $y^0, y^1, \ldots, y^j$ of step 2 of Algorithm 3.1 will be now obtained by the following approximate procedure.

At the beginning of the k-th iteration we define $y^0 = x_k$. Each point y^{i+1} is obtained from y^i by performing a linear search along the direction p^i, whose definition is the same as the one given in Algorithm 2.1, to determine an approximation to the root $h_i > 0$ of the equation

$$\varphi(h) = f(y^i + hp^i) - f(y^i) = 0$$

The function $\varphi(h)$ shares the properties of f and satisfies $\varphi(0) = 0$, $\varphi'(0) = \nabla f(y^0)^T p^i < 0$, recalling that $\nabla f(y^0)^T p^i = (g^i)^T p^i$ due to the definition of $p^i = P^i(-g^i)$ as given in Section 2.

The Search Algorithm generates a finite sequence $\{h^l\}$, $l \geq 0$, and stops when $h^l > 0$ exists such that

$$\varphi(h^l) > \varphi(0) + \alpha h^l \varphi'(0), \; 0 < \alpha < 1/2 \tag{A.1}$$

and at least one of the three following criteria is fulfilled:

(i) If $\varphi(h^l) \leq 0$, and a $\tilde{h}$ exists such that $0 < \tilde{h} < h^l$ where $\varphi(\tilde{h}) < \varphi(h^l)$, making sure that $y^{i+1} = y^i + h^l p^i$ is sufficiently far away from y^i and that it preserves the property $\nabla f(y^{i+1})^T (y^i - y^{i+1}) < 0$ as a consequence of Lemma 2.4.

(ii) If $\varphi(h^l) \geq 0$, $k \geq 1$ and

$$\varphi(h^l) \leq min\{\epsilon_1 max(| f(y^i) |, 1), (f(x_{k-1}) - f(y^i))/2\}, \; 0 < \epsilon_1 < 1 \tag{A.2}$$

(iii) or, si $h^l \in [h_{min}, h_{max}]$, such that $\varphi(h_{min}) < 0$, $0 < \varphi(h_{max}) < f(x_{k-1}) - f(y^i)$, and

$$h_{max} \leq (1 + \epsilon_2) * h_{min}, 0 < \epsilon_2 < 1 \tag{A.3}$$

When (A.1) and (i) hold we accept $h_i = h^l$ making sure that that $y^{i+1} = y^i + h^l p^i$ satisfies $\nabla f(y^{i+1})^T (y^i - y^{i+1}) < 0$. Analogously when (A.1) and the second condition both hold, we accept $h_i = h^l$, making sure that in $y^{i+1} = y^i + h_i p^i$, $f(y^{i+1})$ is close to $f(y^i)$, keeping also $\nabla f(y^{i+1})^T (y^i - y^{i+1}) < 0$. Moreover, in this case we accept y^{i+1}

if it satisfies $f(y^{i+1}) \leq f(y^i) + (f(x_{k-1}) - f(y^i))/2$, being $f(y^i) < f(x_{k-1})$ due to the constraints on y^i for being accepted in the previous internal iteration. Therefore, we also keep that $\|y^{i+1} - y^i\|$ bounded by the radius of the ball containing $L(x_{k-1})$, and $L(x_{k-1})$ contained in $L(x_0)$.

If (A.1) and the third alternative are satisfied, an interval is computed which contains the seeked h_i with the desired precision, and in such a case taking into account that $\tilde{h} = \frac{h_{min}+h_{max}}{2}$ we define $h_i = \tilde{h}$ if $\varphi(\tilde{h}) > \varphi(h_{min})$ and $\varphi(\tilde{h}) > \varphi(0) + \alpha\tilde{h}\varphi'(0)$, or $h_i = h_{max}$ otherwise. Consequently we are sure that for y^{i+1}, the condition $\nabla f(y^{i+1})^T(y^i - y^{i+1}) < 0$ is satisfied, and moreover we assure that $\|y^{i+1} - y^i\| \leq (1+\epsilon_2)h_{min}\|p_i\|$, is bounded by the radius of the ball which contains $L(x_{k-1})$.

With the purpose of defining the scheme of the algorithm for approximate searches we give the following definitions and notations.

From the initial guess h^0 we define $h_{min} = h^0$ if $\varphi(h^0) < 0$ or $h_{max} = h^0$ and $h_{min} = 0$ otherwise.

When the process declares that a current h^l, $l \geq 0$ is not successful, the next candidate h^{l+1} is obtained by quadratic interpolation using
$(\varphi(0), \varphi'(0), \varphi(h^l))$, or using the values of φ at the best three lower or upper bounds obtained up to that moment of h_i, with the necessary safeguards to guarantee convergence. More precisely,

Line Search Procedure
Given y^i, p^i, g^i, $\varphi'(0) = (g^i)^T p^i = \nabla f(y^0)^T p^i$, $0 < \epsilon_1 < 1$, $0 < \epsilon_2 < 1$, $\tau_1 > 0$, $\tau_2 > 0$, $0 < \alpha < 1/2$, $\beta > 1$, and the initial guess h^0.

Step 0 : Set $h_{min} = 0$, $l = 0$.

Step 1 : Compute $\varphi(h^l) = f(y^i + h^l p^i) - f(y^i)$.
If $\varphi(h^l) \geq 0$, set $h_{max} = h^l$. Else, set $h_{min} = h^l$.
If $\varphi(h^l) \leq \varphi(0) + \alpha h^l \varphi'(0)$, go to Step 3.
Else,

Step 2 : Stopping criteria
If condition (i) is satisfied define $h_i = h^l$, go to Step 5.
otherwise,
If (ii) holds, define $h_i = h^l = h_{max}$ go to Step 5.
otherwise,
If $h_{max} > 0$ and if $h_{min} \neq 0$ and (iii) is satisfied
set $h_i = h_{max}$, or $h_i = (h_{min} + h_{max})/2$ according to (iii)
go to Step 5.
Else,

Step 3 : If h_{max} is still undefined, go to Step 4.
Else, set $l = l + 1$.
Set h^l equal to the zero of the quadratic interpolating polynomial,

and the safeguard
$h^l = min[max(h_{min} + \tau_2(h_{max} - h_{min}), h^l), h_{max} - \tau_2(h_{max} - h_{min})]$,
is imposed.
go to Step 1.

Step 4: Set $l = l + 1$ and define h^l equal to the zero of the quadratic interpolating polynomial (using for example $(\varphi(0), \varphi(h_{min}), \varphi'(0))$, or the three best values of $\varphi(h)$ already found).
If $h^l > h_{min}$ define $h^l = min(max((1+\tau_1)h_{min}, h^l), 9h_{min})$. Else, set $h^l = \beta h_{min}$,
Return to Step 1.

Step 5 : Define $y^{i+1} = y^i + h_i p^i$.
Stop.•

The safeguards imposed in Steps 3 and 4 guarantee convergence in a finite number of iterations.

The Algorithm 2.1 is replaced by the following, using the same notation of Section 2.

Algorithm A.2 :
Given x_k, $\nabla f(x_k) \neq 0$,
Step 1: Define $y^0 = x_k$, $g^0 = \nabla f(x_k)$, $i = 0$.
Step 2: If $P^i(-g^i) = 0$, define $j = i$. Stop.
Else,
Step 3: Take $p^i = P^i(-g^i)$; $y^{i+1} = y^i + h_i p^i$, with h_i calculated as in Algorithm A.1.
If $\nabla f(y^{i+1})^T(y^0 - y^{i+1}) < 0$ define $g^{i+1} = \alpha_{i+1}\nabla f(y^{i+1})$ satisfying $\alpha_{i+1}\nabla f(y^{i+1})^T(y^{i+1} - y^0) = \nabla f(y^0)^T(y^0 - y^{i+1})$.
Else, $g^{i+1} = 0$.
Update P^{i+1}.
i=i+1; go to Step 2. •

Remark A.1: The imposed condition (A.1), and the alternatives (i), (ii) or (iii) for accepting each y^{i+1} as an approximation to the point on the level set, do not necessarily lead to $\nabla f(y^{i+1})^T(y^{i+1} - y^0) > 0$, in spite of the condition $\nabla f(y^{i+1})^T(y^{i+1} - y^i) > 0$ has been preserved. Hence, in order to guarantee the good definition of g^{i+1} the modification in the step 3 of algorithm A.2 has been introduced.

The final accepted point y^j, and the search direction $d_k = y^j - x_k$ of Algorithm 3.1, satisfy $\|y^j - x_k\| \leq \sum_{i=1}^{j} \|y^i - y^{i-1}\| \leq n * M$, if M is the bound of the radius of the ball containing $L(x_0)$, which is a rough bound but easy to compute.

Moreover, as a consequence of condition (A.1) each $i = 1, \ldots, j$ satisfies $f(y^i) > f(y^{i-1}) + \alpha\nabla f(x_k)^T(y^i - y^{i-1})$. Thus

$$f(y^i) > f(x_k) + \alpha \nabla f(x_k)^T (y^i - x_k) \tag{A.4}$$

Remark A.2: i) The direction $d_k = y^j - x_k$ obtained in Step 3 of Algorithm 3.1 using the point y^j given by the above procedure is a descent direction as a consequence of the definition of p^i and the arguments of Lemma 2.5 a).

ii) Furthermore, the condition (A.4) satisfied for each y^i $i = 1, \ldots, j$, together with the bound on $d_k = y^j - x_k$ make sure the results of Lemma 3.1 hold, and therefore also Theorem 3.1 is valid. This follows from the fact that for each y^i $f(y^i) - f(x_k) = \nabla f(x_k)^T (y^i - x_k) + 1/2 (y^i - x_k)^T \bar{H} (y^i - x_k)$, where $\bar{H}$ denotes the Hessian matrix at an intermediate point between y^i and x_k.

Since $f(y^i) - f(x_k) > \alpha \nabla f(x_k)^T (y^i - x_k)$, we get $\nabla f(x_k)^T (y^i - x_k) + 1/2 (y^i - x_k)^T \bar{H} (y^i - x_k) > \alpha \nabla f(x_k)^T (y^i - x_k)$

Such a condition makes sure that $1/2 (y^i - x_k)^T \bar{H} (y^i - x_k) > (\alpha - 1) \nabla f(x_k)^T (y^i - x_k)$. Then, we obtain $-(1-\alpha) \nabla f(x_k)^T (y^i - x_k) < (1/2) c \|y^i - x_k\|^2$ considering that $f \in C^2(L(x_0))$ and the compactness of $L(x_0)$. This proves (3.3) modified by a factor $(1-\alpha)$.

In particular, for $i = 1$ we have $-(1-\alpha) \nabla f(x_k)^T (y^1 - x_k) < c/2 \|y^1 - x_k\|^2$, obtaining $(1-\alpha) \|\nabla f(x_k)\| < (1/2) c \|y^1 - x_k\|$

Since $-\nabla f(x_k)^T (y^j - x_k) \geq -\nabla f(x_k)^T (y^1 - x_k) = \|\nabla f(x_k)\| \|y^1 - x_k\|$, and using the previous remark which shows that $\|d_k\|$ is bounded , we get $-\nabla f(x_k)^T d_k / \|d_k\| \geq \gamma_2 \|\nabla f(x_k)\|^2$ with $\gamma_2 = 2(1-\alpha)/(c\tilde{M})$ and $\|d_k\| \leq \tilde{M}$.

That result is identical to the one used for proving Theorem 3.1. Hence, Theorem 3.2 still holds proving that Algorithm 3.1 is globally convergent when the direction d_k is used, arising from the y^j obtained by the inexact search.

B

Let us assume the hypotheses on $f(x)$ stated in Section 4. Let $\{x_k\}$ be the sequence given by Algorithm 3.1 satisfying that there is an index k_0 such that for all $k \geq k_0$, $f(x)$ is uniformly convex on $L(x_k)$, i.e. a constant $m > 0$ exists such that for $x \in L(x_k)$, $y \in \Re^n$, we get

$$m \|y\|^2 \leq y^T \nabla^2 f(x) y \leq c \|y\|^2. \tag{B.1}$$

Lemma B.1: At the kth iteration of Algorithm 3.1, $k \geq k_0$, the following relations hold among the directions, step sizes, and coefficients defined for all $i = 1, \ldots, j \leq n$:

a) $(1/2) m \|y^i - y^0\|^2 \leq (-\nabla f(x_k))^T (y^i - y^0) \leq (1/2) c \|y^i - y^0\|^2$

b) $\|y^i - y^0\| \leq (2/m)\|\nabla f(x_k)\|$

c) $(1/2)\alpha_i m\|y^i - y^0\|^2 \leq {g^i}^T(y^i - y^0) \leq (1/2)\alpha_i c\|y^i - y^0\|^2$

d) $\|g^i\| \geq (1/2)m\alpha_i\|y^i - y^0\|$

e) Since $\alpha_i = \nabla f(x_k)^T(y^0 - y^i)/\nabla f(y^i)^T(y^i - y^0)$ we get $\frac{m}{c} \leq \alpha_i \leq \frac{c}{m}$

f) $\mid \alpha_i - 1 \mid \leq (L/m)\|y^i - y^0\| \leq (2L/m^2)\|\nabla f(x_k)\|$

g) $\|\nabla f(y^i)\| \leq w\|\nabla f(x_k)\|$, where $w = (2c/m) + 1$.

h) From the definition of h_i, we get $2/c \leq h_i \leq 2/m$, $0 \leq i \leq j-1$.

Proof. a) Follows from (B.1) and $f(y^i) = f(y^0)$.

b) Follows from the first inequality of a).

c) From the definition of y^i, g^i, and the uniform convexity on $L(x_k)$.

d) From c).

e) From $f(y^i) = f(x_k)$ and the uniform convexity of f.

f) From the definitions of α_i and g^i, using the hypotheses and e).

g) From the uniform convexity of f and e).

h) From the definition of the points y^i and y^{i+1}, the Taylor's expansion for $f(y^{i+1})$ at y^i, and (B.1).■

Lemma B.2: Under the above stated hypotheses, $\mu > 0$ exists such that

$$cos\theta_k{}^i = (g^i - g^{i-1})^T(y^i - y^{i-1})/(\|g^i - g^{i-1}\| \ \ \|y^i - y^{i-1}\|) \geq \mu$$

$1 \leq i \leq j \leq n$, for $k \geq k_0$.

Proof. Due to the definition of $g^i, g^{i-1}, y^i, y^{i-1}$, condition (B.1), Lemma B.1 f), and the Taylor's expansion at y^{i-1} and y^i we have that

$$(g^i - g^{i-1})^T(y^i - y^{i-1}) = (1/2)\alpha_i(y^i - y^{i-1})^T\bar{H}(y^i - y^{i-1}) \qquad \text{(B.2)}$$

$$+(1/2)\alpha_{i-1}(y^i - y^{i-1})^T\hat{H}(y^i - y^{i-1}) \geq (1/2)(\alpha_i + \alpha_{i-1})m\| \, y^i - y^{i-1} \, \|^2$$

$$\geq (m^2/c)\| \, y^i - y^{i-1} \, \|^2,$$

where $\bar{H}$ and $\hat{H}$ are the Hessians at the corresponding intermediate points.

Using the definitions of α_i and α_{i-1} we shall bound the denominator of the expression for $cos\theta_k{}^i$.

From

$$\alpha_i = (\alpha_{i-1}\nabla f(y^{i-1})^T(y^{i-1}-y^0) + \nabla f(y^0)^T(y^{i-1}-y^i))/\nabla f(y^i)^T(y^i-y^0)$$

we get that

$$\alpha_i - \alpha_{i-1} = \alpha_{i-1}\nabla f(y^i)^T(y^{i-1}-y^i)/\nabla f(y^i)^T(y^i-y^0)$$

$$+(\alpha_{i-1}(\nabla f(y^{i-1}) - \nabla f(y^i))^T(y^{i-1}-y^0) + \nabla f(y^0)^T(y^{i-1}-y^i))/\nabla f(y^i)^T(y^i-y^0).$$

This implies that

$$\mid \alpha_i - \alpha_{i-1} \mid \; \|\nabla f(y^i)\|$$

$$\leq \alpha_{i-1}\|\nabla f(y^{i-1}) - \nabla f(y^i)\| \; \|y^{i-1}-y^0\| \; \|\nabla f(y^i)\|/((1/2)m\|y^i-y^0\|^2)$$

$$+\|\nabla f(y^0) + \alpha_{i-1}\nabla f(y^i)\| \; \|y^{i-1}-y^i\| \; \|\nabla f(y^i)\|/((1/2)m\|y^i-y^0\|^2).$$

Using Lemmas 3.1 and B.1, it is easy to prove a constant $\sigma > 0$ exists such that the right hand side of the above expression can be bounded by $\sigma\|y^i - y^{i-1}\|$. Therefore, $\mid \alpha_i - \alpha_{i-1} \mid \; \|\nabla f(y^i)\| \leq \sigma\|y^i - y^{i-1}\|$.

Taking into account the definition of g^i, g^{i-1}, the hypotheses, and the previous bounds, we have that

$$\|g^i - g^{i-1}\| \;\leq \mid \alpha_i - \alpha_{i-1} \mid \; \|\nabla f(y^i)\| + \mid \alpha_{i-1} \mid \; \|\nabla f(y^{i-1}) - \nabla f(y^i)\|$$

$$\leq \sigma\|y^i - y^{i-1}\| + (c^2/m)\|y^i - y^{i-1}\| = (\sigma + c^2/m)\|y^i - y^{i-1}\|.$$

Therefore,

$$\|g^i - g^{i-1}\| \; \|y^i - y^{i-1}\| \leq (\sigma + (c^2/m))\|y^i - y^{i-1}\|^2. \tag{B.3}$$

Hence, because of (B.2) and (B.3), $\mu > 0$ exists such that $cos\theta_k{}^i \geq \mu$.■

We shall denote by $P_k{}^i$, $A_k{}^i$, and $S_k{}^j$, the matrices generated within Step 2 of Algorithm 3.1 by means of Algorithm 2.1.

Lemma B.3: If $P_k{}^{i-1}(-\nabla f(x_k)) \neq 0$, and under the hypotheses stated for $L(x_k)$, then a constant α exists such that $0 \leq \alpha < 1$ and $\|P_k{}^i(-\nabla f(x_k))\| \leq \alpha\|P_k{}^{i-1}(-\nabla f(x_k))\|$, $1 \leq i \leq j$.

Proof. Using Greville's formula (Ben Israel and Greville [2])

$$P_k{}^i(-\nabla f(x_k)) = P_k{}^{i-1}(-\nabla f(x_k)) -$$

$$P_k{}^{i-1}(g^i - g^{i-1})(g^i - g^{i-1})^T P_k{}^{i-1}(-\nabla f(x_k))/\|P_k{}^{i-1}(g^i - g^{i-1})\|^2$$

from which it follows that

$$\|P_k{}^i(-\nabla f(x_k))\|^2 =$$

$$\|P_k{}^{i-1}(-\nabla f(x_k))\|^2 - ((g^i - g^{i-1})^T P_k{}^{i-1}(-\nabla f(x_k)))^2)/\|P_k{}^{i-1}(g^i - g^{i-1})\|^2.$$

Since $cos\theta_k{}^i \geq \mu > 0$ according to Lemma B.2, we get

$$\|P_k{}^i(-\nabla f(x_k))\|^2 \leq (1-\mu^2)\|P_k{}^{i-1}(-\nabla f(x_k))\|^2.$$

Defining $\alpha = (1-\mu^2)^{1/2}$, the proof is complete.■

We shall denote by e_i the vectors of the canonical basis.

Lemma B.4: Under the stated hypotheses, if $y^0, y^1, ..., y^j$ and $g^0, g^1, ..., g^j$ are the vectors generated by the kth iteration of Algorithm 3.1, then the $j \times n$ matrix $T_k{}^j$ whose ith row is $(y^i - y^{i-1})^T$, $1 \leq i \leq j$, is such that its pseudoinverse $T_k{}^{j\dagger}$ has columns $T_k{}^{j\dagger} e_i$ satisfying

$$\|T_k{}^{j\dagger} e_i\| \leq 1/\gamma \|y^i - y^{i-1}\|, \quad with \ \ \gamma > 0.$$

Proof. For the non trivial case $j \geq 2$ we shall denote by γ_i for $2 \leq i \leq j$, the angle between the ith row of $T_k{}^j$ and the subspace spanned by the $i-1$ first rows.

Let us denote by $T_k{}^{i-1}$ the matrix composed of the first $i-1$ rows of $T_k{}^j$, and by $P_{T_k{}^{i-1}}$ the projection matrix onto its null space. Then

$$sin\gamma_i = \|P_{T_k{}^{i-1}}(y^i - y^{i-1})\|/\|y^i - y^{i-1}\| = \|P_{T_k{}^{i-1}} p^{i-1}\|/\|p^{i-1}\|.$$

Since $R((T_k{}^{i-1})^T) = R((A_k{}^{i-2})^T \cup \{p^{i-2}\})$ as a consequence of Remark 2.2, using Greville's formula we get

$$P_{T_k{}^{i-1}} = P_k{}^{i-2} - (P_k{}^{i-2}p^{i-2})(P_k{}^{i-2}p^{i-2})^T/\|P_k{}^{i-2}p^{i-2}\|^2$$

$$= P_k{}^{i-2} - p^{i-2}(p^{i-2})^T/\|p^{i-2}\|^2.$$

Thus,

$$sin\gamma_i = \|p^{i-1} - p^{i-2}((p^{i-2})^T p^{i-1}/\|p^{i-2}\|^2)\|/\|p^{i-1}\| \ .$$

Using the fact the numerator is the orthogonal projection of p^{i-1} onto the orthogonal subspace to p^{i-2} we have that

$$sin\gamma_i = ((\|p^{i-1}\|^2 - \|p^{i-1}\|^4)/\|p^{i-2}\|^2)^{1/2}/\|p^{i-1}\| \ .$$

Due to Lemma B.3, a constant α exists such that $0 \leq \alpha < 1$ and $sin\gamma_i \geq (1-\alpha^2)^{1/2}$.

Hence, if β_i is the angle between the ith row of ${T_k}^j$ and the subspace spanned by the remaining rows (Dennis et al. [6]) we obtain

$$sin\beta_i \geq (1-\alpha^2)^{(j-1)/2} \geq (1-\alpha^2\ )^{(n-1)/2} = \gamma \ .$$

Then, using the same arguments of Dennis et al. [6], we can prove that the pseudoinverse ${T_k}^{j\dagger} = (b_1, ..., b_j)$ is such that $\|b_i\| \leq 1/(\gamma\|y^i - y^{i-1}\|)$. ■

Lemma B.5: Under the stated hypotheses for the kth iteration, $k \geq k_0$,

$$(\alpha_i - \alpha_{i-1})\nabla f(y^i)^T(y^i - y^0) = (y^i - y^{i-1})^T v_i, \quad for\ i = 1, ..., j,$$

where $\| \ v_i \ \| = O(\| \ \nabla f(x_k) \ \|^2)$

Proof. Taking into account the definition of α_i and $y^i - y^{i-1}$ we can write

$$(\alpha_i - \alpha_{i-1})\nabla f(y^i)^T(y^i - y^0) = \alpha_{i-1}[(\nabla f(y^{i-1}) + \nabla f(y^i))^T(y^{i-1} - y^i)$$
$$+(\nabla f(y^{i-1}) - \nabla f(y^i))^T(y^{i-1} - y^0)].$$

Using the Taylor's expansion at y^i and y^{i-1} and the equality $f(y^i) = f(y^{i-1})$ in the first term of the right hand side we get

$$(\nabla f(y^{i-1}) + \nabla f(y^i))^T(y^{i-1} - y^i) = (y^i - y^{i-1})^T(\bar{H}_i - \tilde{H}_i)(y^i - y^{i-1}).$$

Considering that $y^i - y^{i-1} = h_{i-1}{P_k}^{i-1}(-\nabla f(x_k))$ and using the Taylor's expansion, the second term of the right hand side can be written as

$$(\nabla f(y^{i-1}) - \nabla f(y^i))^T(y^{i-1} - y^0) = (y^{i-1} - y^i)^T\hat{H}_i(y^{i-1} - y^0)$$
$$= (y^i - y^{i-1})^T[g^{i-1} - \nabla f(x_k) + (1 - \alpha_{i-1})\nabla f(y^{i-1}) + (\hat{H}_i - \bar{H}_0)(y^{i-1} - y^0)]$$

$$= (y^i - y^{i-1})^T[(1-\alpha_{i-1})\nabla f(y^{i-1}) + (\hat{H}_i - \bar{H}_0)(y^{i-1} - y^0)].$$

Then, from the Lipschitz continuity of the Hessian and the relationships of Lemma B.1, we obtain

$$(\alpha_i - \alpha_{i-1})\nabla f(y^i)^T(y^i - y^0) = (y^{i-1} - y^i)^T v_i,$$

with $\|v_i\| = O(\|\nabla f(x_k)\|^2)$.■

Theorem B.1: Under the stated hypotheses for all $k \geq k_0$, if j is the first index in the kth iteration such that ${P_k}^j(-\nabla f(x_k)) = 0$, then

$$\|\nabla f(y^j) + \nabla f(x_k)\| = O(\|\nabla f(x_k)\|^2).$$

Proof. The search direction given by Algorithm 3.1 is the solution of the system

$${A_k}^j d_k = {S_k}^j(-\nabla f(x_k)). \tag{B.4}$$

Because the matrix ${S_k}^j$ can be written as ${S_k}^j = E_k {T_k}^j$, where the $j \times j$ matrix E_k is such that its ith row is $\sum_{r=1}^i {e_r}^T$, and the matrix ${A_k}^j = E_k {B_k}^j$ where ${B_k}^j$ is a $j \times n$ matrix such that the ith row is $g^i - g^{i-1}$, we have that the system (B.4) is equivalent to

$${B_k}^j d_k = {T_k}^j(-\nabla f(x_k)). \tag{B.5}$$

Due to the fact that $d_k = (y^j - y^0)/2$ and the definition of g^i and ${T_k}^j$, the left hand side of B.5 can be written as

$${B_k}^j d_k = {B_k}^j(y^j - y^0)/2$$

$$= \sum_{i=1}^{j} e_i[(\alpha_i - 1)\nabla f(y^i) - (\alpha_{i-1} - 1)\nabla f(y^{i-1})]^T(y^j - y^0)/2$$

$$+ \sum_{i=1}^{j} e_i(\nabla f(y^i) - \nabla f(y^{i-1}))^T(y^j - y^0)/2$$

$$= \sum_{i=1}^{j} e_i[(\alpha_i - 1)\nabla f(y^i) - (\alpha_{i-1} - 1)\nabla f(y^{i-1})]^T(y^j - y^0)/2$$

$$+ \sum_{i=1}^{j} e_i[(\bar{H}_i - H_k)(y^i - y^{i-1})]^T(y^j - y^0)/2 + {T_k}^j H_k(y^j - y^0)/2,$$

where $H_k = \nabla^2 f(x_k)$ and $\bar{H}_i$ is the Hessian at an intermediate point.

Multiplying both members of (B.5) by ${T_k^j}^\dagger$, using the equality

$$H_k(y^j - y^0)/2 = (\nabla f(y^j) - \nabla f(y^0))/2 + (H_k - \bar{H}_0)(y^j - y^0)/2,$$

rearranging terms, and considering that $\nabla f(x_k), \nabla f(y^j) \in R((T_k^j)^T)$ due to Corollary 2.1, we obtain

$$(\nabla f(y^j) + \nabla f(x_k))/2 = {T_k^j}^\dagger T_k^j(\bar{H}_0 - H_k)(y^j - y^0)/2 \tag{B.6}$$

$$+{T_k^j}^\dagger \sum_{i=1}^{j} e_i[(1-\alpha_i)\nabla f(y^i) - (1-\alpha_{i-1})\nabla f(y^{i-1})]^T(y^j - y^0)/2$$

$$+{T_k^j}^\dagger \sum_{i=1}^{j} e_i(y^i - y^{i-1})^T(H_k - \bar{H}_i)(y^j - y^0)/2.$$

Then, because of Lemma B.1 (f), the second term of the right hand side can be written as

$${T_k^j}^\dagger \sum_{i=1}^{j} e_i[(\alpha_{i-1} - \alpha_i)\nabla f(y^i)^T(y^i - y^0)/2$$

$$-(1-\alpha_{i-1})(\nabla f(y^{i-1}) - \nabla f(y^i))^T(y^i - y^0)/2 + {\omega_i}^T(y^j - y^i)/2],$$

where $\|\omega_i\| = O(\|\nabla f(x_k)\|^2)$.

Also, using the Lipschitz continuity of the Hessian and Lemmas B.3, B.4 and B.5 it is easy to see that the norm of the right hand side of (B.6) is $O(\| \nabla f(x_k)\|^2)$.

Therefore, it follows that $\|\nabla f(y^j) + \nabla f(x_k)\| = O(\|\nabla f(x_k)\|^2)$.■

Lemma B.6: Under the stated hypotheses, if $k \geq k_0$ and $x_{k+1} = x_k + d_k$, then

$$\|\nabla f(x_{k+1})\| = O(\|\nabla f(x_k)\|^2).$$

Proof. From Taylor's expansion, the Lipschitz continuity of the Hessian and Lemma B.1 b),

$$\nabla f(x_{k+1}) = \nabla f(x_k) + H_k(x_{k+1} - x_k) + O(\|\nabla f(x_k)\|^2) \tag{B.7}$$

$$\nabla f(x_{k+1}) = \nabla f(y^j) + H_k(x_{k+1} - y^j) + O(\|\nabla f(x_k)\|^2) \tag{B.8}$$

By adding (B.7) and (B.8) and considering that $x_{k+1} = (x_k + y^j)/2$, we get

$$\nabla f(x_{k+1}) = (\nabla f(y^j) + \nabla f(x_k))/2 + O(\|\nabla f(x_k)\|^2).$$

Therefore, from Theorem B.1 it follows that $\|\nabla f(x_{k+1})\| = O(\|\nabla f(x_k)\|^2)$. ■

References

[1] BAZARAA, M.S. and G.M. SHETTY,*Nonlinear Programming: Theory and Algorith ms*, John Wiley & Sons, 1979.

[2] BEN-ISRAEL, A. and T. GREVILLE, *Generalized Inverses: Theory and Aplications*, John Wiley & Sons, 1974.

[3] BJÖRCK, A., *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia , 1996.

[4] BYRD, R.H., J. NOCEDAL and Y.YUAN, "Global convergence of a class of Quasi-Newton methods on convex problems",*SIAM J. Numer. Anal.*, 24(1987), 1171-1190.

[5] CROUZEIX, J. and J. FERLAND, "Criteria for quasi-convexity and pseudo-convexity: Relationships and Comparisons", *Mathematical Programming*, 23(1982), 193-205.

[6] DENNIS, J.E. Jr., N. ECHEBEST, M.T. GUARDARUCCI, J.M. MARTINEZ, H.D. SCOLNIK and M.C. VACCHINO, "A curvilinear search using tridiagonal secant updates for unconstrained optimization", *SIAM J. Optimization*,1 (1991), 333-357.

[7] ECHEBEST, N., M.T. GUARDARUCCI, H.D. SCOLNIK and M.C. VACCHINO, "Large Step Discrete Newton Methods for minimizing quasi-convex functions",*Notas de Matemática 53, Departamento de Matemática, UNLP*, La Plata, Argentina, 1993.

[8] ECHEBEST, N., M.T. GUARDARUCCI, H.D. SCOLNIK and M.C. VACCHINO, "Método de Newton discretizado con pasos largos para minimizar funciones con conjunto de nivel acotado", *Proc. of VII Congreso Latino-Ibero Americano de Investigación Operativa e Ingeniería de Sistemas, Universidad de Santiago*, Chile, 1994.

[9] FLETCHER, R., *Practical Methods of Optimization*, J. Wiley & Sons, 1987.

[10] FRIEDLANDER, A., J.M. MARTINEZ and H.D. SCOLNIK, "Generalized inverses and a new stable type minimization algorithm", *Proc. of the 8th IFIP Conference on Optimization Techniques*, Springer-Verlag, 1977.

[11] GAUDIOSO, M. and M.F. MONACO, "The Newton direction and the cutting plane", *European Journal of Operational Research*,73(1994), 172-174.

[12] GILL, P.E. and W. MURRAY, "Conjugate-gradient methods for large-scale nonlinear optimization", *Technical report SOL 79-15, Department of Operations Research, Stanford University*,(Stanford,CA,1979)

[13] GOLUB, G.H. and C.H.F. VAN LOAN, *Matrix Computations*, The Johns Hopkins Univ. Press, 1985.

[14] GREENBERG, H.J. and W.P. PIERSKALLA, "A review of quasi-convex functions", *Operations Research*, 19(1971), 1553-1570.

[15] LIN, C.-J. and J.J. MORE, "Newton's Method for Large Bound- Constrained Optimization Problems", Preprint, ANL/MCS-P724-0898, August 1998.

[16] MORE, J.J., B.S. GARBOW and K.L. HILLSTROM, "Fortran subroutines for testing unconstrained optimization software", *ACM Trans. Math. Software*, 7(1981), 17-41.

[17] NASH, S.G., "Newton-like minimization method", *SIAM J. Num. Anal.* 21(1984), 770-788.

[18] ROCKAFELLAR, R.T., *Convex Analysis*, Princeton Univ.Press, Princeton, NJ, 1970.

[19] WOLFE, P., "Convergence conditions for ascent methods", *SIAM Rev.*, 11(1969), 226-235.

[20] WOLFE, P., "Convergence conditions for ascent methods. II: Some corrections", *SIAM Rev.*, 13(1971), 185-188.

[21] ZOUTENDIJK, G., *Nonlinear Programming, Computational Methods, in Integer and Nonlinear Programming*, J. Abadie, ed., North-Holland, Amsterdam, 1970, 37-86.