An adaptive algorithm for bound constrained quadratic minimization *

Roberto Hugo Bielschowsky   Ana Friedlander
Francisco A. M. Gomes   José Mario Martínez
Marcos Raydan

May 2, 1997

Abstract

A general algorithm for minimizing a quadratic function with bounds on the variables is presented. The new algorithm can use different unconstrained minimization techniques on different faces. At every face, the minimization technique can be chosen according to he structure of the Hessian and the dimension of the face. The strategy for leaving the face is based on a simple scheme that exploits the properties of the "chopped gradient" introduced by Friedlander and Martínez in 1989. This strategy guarantees global convergence even in the presence of dual degeneracy, and finite identification in the nondegenerate case. A slight modification of the algorithm satisfies, in addition, an identification property in the case of dual degeneracy. Numerical experiments combining this new strategy with conjugate gradients, gradient with retards and direct solvers are presented.

Key words. Quadratic programming, conjugate gradients, gradient with retards, active set methods, sparse Cholesky factorization.

AMS subject classifications. 65K10, 49M07, 65F15, 90C20

*Department of Applied Mathematics, IMECC-UNICAMP, CP 6065, 13081-970 Campinas SP, Brazil and Department of Mathematics, Central University of Venezuela (UCV), Caracas, Venezuela.

This work was supported by FAPESP grants 90-3724-6 and 94-3807-0, FINEP, CNPq, FAEP-UNICAMP, and BID-CONICIT project M-51940.

1 Introduction

Many practical problems require the minimization of quadratic functions with bounds on the variables. Frequently, the number of variables is large, and the problem is challenging even for advanced computers. See [6, 7, 12, 14, 15, 20, 21, 22, 25] and references therein. Moreover, many times, quadratic minimization problems with box constraints must be solved in the context of sophisticated algorithms for nonlinear programming. See, for example, [17, 23].

Some effective techniques for solving quadratic box-constrained minimization problems are based on gradient projections (see [3]) and conjugate gradients. See, for example, [14, 22]. By this we mean that the feasible box is implicitly divided into disjoint faces, the conjugate gradient method [19] is used within the faces, where the problem is essentially unconstrained, and the projections of half-lines defined by suitable descent directions are used for leaving a face, when this is necessary. In [22] the convergence of one of these methods is proved in the case of a strictly convex quadratic and finite convergence is proved when the limit point is not dual-degenerate. Friedlander and Martínez [14] used the properties of the "chopped gradient" to introduce a method that has finite convergence even when the Hessian is singular and in the presence of dual-degeneracy.

Looking at the the solution of very large minimization problems, Barzilai and Borwein [2] introduced a nonmonotone method for unconstrained quadratic minimization where only gradient directions are used and the memory requirements are minimal. Raydan [24] proved the convergence of the method for general strictly convex quadratics and Friedlander, Martínez and Raydan [15] extended the proof to singular problems and applied the Barzilai-Borwein technique to the box-constrained quadratic problem. The same authors [16] introduced generalizations of the original Barzilai-Borwein method that exploit the asynchronous philosophy in parallel environments.

One motivation of the present work is the observation that, in a single practical large-scale problem, faces of different dimensions can be explored. So, the most appropriate unconstrained technique within one face might not be the optimal technique for a face of different dimension. In particular, we wish to consider the following unconstrained techniques:

(a) Cholesky factorization of the reduced Hessian (dense or sparse);

(b) conjugate gradients;

(c) Barzilai-Borwein method and the generalizations given in [16] (gradient methods with retards).

Roughly speaking, for faces of small dimension, the dense Cholesky factorization should be used and, as the dimension increases, we should try sparse Cholesky factorization, conjugate gradients and, finally, gradient methods with retards. However, depending on the structure of the problem, the most "natural" choice may not be the most efficient one.

Let us now briefly review the convergence results that have been obtained for predecessors of the method introduced in this paper:

(a) Moré and Toraldo [22] proved convergence of their method for strictly convex box-constrained quadratic minimization and finite convergence when dual-degeneracy is excluded.

(b) Friedlander and Martínez [14] proved finite convergence of their method for (not necessarily strictly) convex box-constrained quadratic minimization even when the problem is dual-degenerate.

(c) Friedlander, Martínez and Raydan [15] proved convergence of their algorithm for (not necessarily strictly) convex box-constrained quadratic minimization, using the Barzilai-Borwein method within the faces.

The finite convergence results of [22] and [14] follow from the fact that the conjugate gradient method converges in a finite number of iterations for unconstrained quadratic problems. Moreover, the finite convergence in degenerate problems of [14] uses a bound of the norm of the quadratic Hessian. Clearly, since the Barzilai-Borwein method is not finitely convergent, it is not possible to obtain finite convergence results when this algorithm is used as an unconstrained minimizer in the box-constrained problem. On the other hand, we observe that computable bounds of the quadratic Hessian can be very unrealistic in many cases, so we decided not to use this type of bounds in the new algorithm. Namely, we are going to prove the following results for the adaptive algorithm:

(1) A limit point of the sequence generated by the algorithm exists, that is a stationary point of the problem.

(2) In the (not necessarily strictly) convex box-constrained quadratic case, the following results hold:

(a) every limit point of a sequence generated by the algorithm is a solution;

(b) either the sequence generated by the algorithm converges to a nondegenerate solution x^* with finite identification of the active constraints at x^*, or there will necessarily be dual degenerate solutions among its limit points.

(c) Even in the dual-degenerate case, a modification of the main algorithm satisfies the following "weak identification property": There exists a set of constraints $\mathcal{S}$ such that, eventually, the sequence of iterates satisfies exactly the constraints of that set. In this case the sequence is convergent and the constraints at $\mathcal{S}$ are active at the limit points. (However, the existence of an active constraint in the limit not belonging to $\mathcal{S}$ is possible in the dual-degenerate case.)

This paper is organized as follows: in Sections 2 and 3 we describe the algorithm and we prove the convergence results. In Section 4 we describe the strategies used for unconstrained minimization within the faces, emphasizing the recently introduced gradient methods with retards. In Section 5, we show the numerical experiments. In Section 6, we give some conclusions and we state the lines for future research.

2 Main model algorithm

We consider the problem of minimizing a quadratic function with bound constrained variables

$$\begin{aligned} &\text{Minimize } \Psi(x) \\ &\text{subject to } x \in \Omega, \end{aligned} \tag{2.1}$$

where $\Omega = \{x \in I\!\!R^n \,|\, l \le x \le u, l < u\}$, $\Psi(x) = \frac{1}{2}x^T H x + b^T x$ and $l, u \in I\!\!R^n$. We denote $\gamma = \min\{u_i - l_i\,,\ i = 1, \ldots, n\}$ and

$$\bar{g}(x) \equiv -\nabla\Psi(x) \equiv -(Hx + b)$$

for all $x \in I\!\!R^n$. Let $L > 0$ be such that $\|H\| \le L$. ($\|\cdot\|$ denotes the 2-norm of vectors or matrices.) Therefore, for all $x, z \in I\!\!R^n$, we have that

$$\Psi(z) - \Psi(x) - \nabla\Psi(x)^T(z - x) = \frac{1}{2}(z - x)^T H(z - x) \le \frac{L}{2}\|z - x\|^2. \tag{2.2}$$

Clearly, if Ψ is convex,

$$\Psi(z) - \Psi(x) - \nabla\Psi(x)^T(z - x) \ge 0. \tag{2.3}$$

We define an open face of Ω as a set $F_I \subset \Omega$ such that I is a (possibly empty) subset of $\{1, 2, \ldots, 2n\}$ such that i and $n + i$ cannot belong simultaneously to I for any $i \in \{1, 2, \ldots, n\}$; and such that

$$F_I = \{x \in \Omega \mid x_i = l_i \text{ if } i \in I,\ x_i = u_i \text{ if } n + i \in I,\ l_i < x_i < u_i \text{ otherwise }\}.$$

Let us call $\bar{F}_I$ the closure of each open face, $[F_I]$ the smallest linear manifold that contains F_I, $S(F_I)$ the parallel subspace to $[F_I]$ and $dim\ F_I$ the dimension of $S(F_I)$. Clearly, $dim\ F_I = n - |I|$. For each $x \in \Omega$ let us define the (negative) projected gradient $\bar{g}_P(x) \in I\!R^n$ as

$$\bar{g}_P(x)_i = \begin{cases} 0 & \text{if} \quad x_i = l_i \quad \text{and} \quad \frac{\partial \Psi}{\partial x_i}(x) > 0 \\ 0 & \text{if} \quad x_i = u_i \quad \text{and} \quad \frac{\partial \Psi}{\partial x_i}(x) < 0 \\ -\frac{\partial \Psi}{\partial x_i}(x) & \text{else.} \end{cases} \tag{2.4}$$

A necessary condition for x being a global solution of (2.1) (sufficient in the convex case) is that

$$\bar{g}_P(x) = 0\ . \tag{2.5}$$

For each $x \in \bar{F}_I$ let us define $\bar{g}_I(x) \in I\!R^n$ as

$$\bar{g}_I(x)_i = \begin{cases} 0 & \text{if} \quad i \in I \quad \text{or} \quad n+i \in I \\ -\frac{\partial \Psi}{\partial x_i}(x) & \text{else.} \end{cases} \tag{2.6}$$

Observe that $\bar{g}_I(x)$ is the orthogonal projection of $-\nabla\Psi(x)$ on $S(F_I)$. We also define for $x \in \bar{F}_I$,

$$\bar{g}_I^C(x)_i = \begin{cases} 0 & \text{if} \quad i \notin I & \text{and} \quad n+i \notin I \\ 0 & \text{if} \quad i \in I & \text{and} \quad \frac{\partial \Psi}{\partial x_i}(x) > 0 \\ 0 & \text{if} \quad n+i \in I & \text{and} \quad \frac{\partial \Psi}{\partial x_i}(x) < 0 \\ -\frac{\partial \Psi}{\partial x_i}(x) & \text{else.} \end{cases} \tag{2.7}$$

The vector $\bar{g}_I^C(x)$ was introduced in [13], and named *chopped gradient*. Observe that for all $x \in F_I$ we have

$$\bar{g}_P(x) = \bar{g}_I(x) + \bar{g}_I^C(x)\ .$$

We say that a sequence $\{z^k\} \subset I\!R^n$ has the *Property C*, related to the quadratic $Q(z)$ if at least one of the following conditions hold:

(a) If $\nabla Q(z^0) \neq 0$, there exist $z^* \in I\!R^n$, a limit point of $\{z^k\}$, such that $\nabla Q(z^*) = 0$ and $Q(z^*) < Q(z^0)$;

(b) $\lim_{k\to\infty} Q(z^k) = -\infty$.

Accordingly, we say that an algorithm designed to minimize unconstrained quadratics is a $C-$algorithm if, for all $z^0 \in I\!R^n$, it generates a sequence $\{z^k\}$ with the *Property C*.

It has been proved (see [17]) that the conjugate gradient algorithm for minimizing quadratics finds, in a finite number of iterations a stationary point of the function, or a direction along which the quadratic tends to $-\infty$. Therefore, after a trivial modification, we can say that it is a $C-$algorithm. The same can be said about direct methods for unconstrained quadratic minimization provided that some device is introduced to detect negative curvature directions when local minimizers do not exist.

Finally, gradient methods with retards are $C-$algorithms if Q is convex and admits stationary points. See [15, 16].

Now we are ready to present the main algorithm. The description will be "high level" in the sense that the specifications of the internal procedures are postponed for further sections. Nevertheless, the theoretical properties of the algorithm will be discussed within this "high level" framework.

Algorithm 2.1 (Main Model Algorithm)
Let $0 < \eta < 1$ be given independently of k, and let $x_0 \in \Omega$ be an arbitrary initial point. The algorithm defines a sequence $\{x^k\}$ in Ω and stops when $\|\bar{g}_P(x^k)\| = 0$. Let us assume that $x^k \in \Omega$ is such that $\|\bar{g}_P(x^k)\| \neq 0$. Let $I = I(x^k)$ be such that $x^k \in F_I$ (observe that there exists only one subset $I \subset \{1, 2, \ldots, 2n\}$ with this property), and let the function $\Phi(x)$ be defined as

$$\Phi(x) = \text{ argmin}\{\Psi(y) \mid y = x + \lambda \bar{g}_I^C(x) \text{ and } y \in \Omega\} \,. \tag{2.8}$$

The following steps define the procedure for obtaining x^{k+1}.

Step 1: If

$$\|\bar{g}_I^C(x^k)\| > \eta \, \|\bar{g}_P(x^k)\| \,, \tag{2.9}$$

then set $x^{k+1} = \Phi(x^k)$. Else go to Step 2.

Step 2:
Compute, using a $C-$algorithm for minimizing unconstrained quadratic functions, a point $z^k \in [F_I]$ such that $\Psi(z^k) < \Psi(x^k)$. (If $k \geq 1$ and $x^{k-1} \in F_I$, use, at this step, the same $C-$algorithm employed to compute x^k in the previous iteration.) If $z^k \in \bar{F}_I$ then set $x^{k+1} = z^k$. Else go to Step 3.

Step 3:
Find $x^{k+1} \in \bar{F}_I - F_I$ such that $\Psi(x^{k+1}) < \Psi(x^k)$.

In Step 2 we use any $C-$algorithm for unconstrained minimization of quadratic functions on the manifold $[F_I]$. This freedom of choice plays an important role in the behavior of the algorithm, and will be discussed in Section 4. Observe that different $C-$algorithms can be used at different iterations of the main algorithm. However, to change the $C-$algorithm is admitted only when the current iterate lies within a different face than the previous one. Step 3, on the other hand, drives us to the boundary of F_I when z^k is not feasible. The condition in Step 1 allows for non exact minimization in the faces, preserving the required arguments for obtaining convergence. Our next result establishes the existence of computable points that satisfy the requirements of Steps 1, 2 and 3.

Theorem 2.1 *Algorithm 2.1 is well defined.*

Proof. If the condition (2.9) at Step 1 is satisfied, then $\bar{g}_I^C(x^k) \neq 0$, so $\Phi(x^k)$ is well defined. If the condition in Step 1 fails then we execute Step 2. Since we are using a $C-$algorithm, the existence of $z^k \in [F_I]$ such that $\Psi(z^k) < \Psi(x^k)$ is guaranteed. Now, if $z^k \notin \bar{F}_I$, since $\varphi(\lambda) \equiv \Psi(x^k + \lambda(z^k - x^k))$ is a one-dimensional quadratic, then

$$\Psi(x^k + \lambda_{break}^+(z^k - x^k)) < \Psi(x^k) ,$$

or

$$\Psi(x^k + \lambda_{break}^-(z^k - x^k)) < \Psi(x^k) ,$$

where

$$\lambda_{break}^+ = \max\{\lambda \geq 0 \mid [x^k,\ x^k + \lambda(z^k - x^k)] \subset \bar{F}_I\}$$

and

$$\lambda_{break}^- = \min\{\lambda \leq 0 \mid [x^k,\ x^k + \lambda(z^k - x^k)] \subset \bar{F}_I\}.$$

Therefore, the choice of $x^{k+1} \in \bar{F}_I - F_I$ satisfying $\Psi(x^{k+1}) < \Psi(x^k)$, in Step 3, is possible. $\Box$

The following lemma establishes a sufficient decrease obtained in the objective function every time the condition in Step 1 is satisfied. This sufficient decrease will be used in our convergence results. Although the proof of this lemma is rather straightforward and can be deduced from well-known results on trust-region strategies, we decided to include it here for the sake of completeness.

Lemma 2.1 *If x^{k+1} is obtained at Step 1 of Algorithm 2.1 then*

$$\Psi(x^k) - \Psi(x^{k+1}) \geq \min\{\frac{\eta\,\gamma}{2}\|\bar{g}_P(x^k)\|\,,\,\frac{\eta^2}{2L}\|\bar{g}_P(x^k)\|^2\}\,.$$

Proof. Since x^{k+1} is obtained at Step 1, then $\bar{g}_I^C(x^k) \neq 0$. Hence, $x^k + \lambda\bar{g}_I^C(x^k) \in \Omega$ for all $\lambda \in [0, \tilde{\lambda}]$, where $\tilde{\lambda} = \gamma/\|\bar{g}_I^C(x^k)\|$. Let us consider the quadratic function given by

$$\mu(\lambda) = \Psi(x^k+\lambda\bar{g}_I^C(x^k)) = \Psi(x^k)+\lambda\nabla\Psi(x^k)^T\bar{g}_I^C(x^k)+\frac{1}{2}\lambda^2\bar{g}_I^C(x^k)^T H\bar{g}_I^C(x^k)\,.$$

If $\bar{g}_I^C(x^k)^T H\bar{g}_I^C(x^k) > 0$ then the unique minimizer of $\mu(\lambda)$ is given by

$$\lambda^* = \frac{\|\bar{g}_I^C(x^k)\|^2}{\bar{g}_I^C(x^k)^T H\bar{g}_I^C(x^k)}\,.$$

If $x^k + \lambda^*\bar{g}_I^C(x^k)$ is not feasible, then $x^{k+1} = \Phi(x^k)$ is attained at some $\bar{\lambda}$ such that $\tilde{\lambda} \leq \bar{\lambda} < \lambda^*$, and

$$\Psi(x^k + \tilde{\lambda}\bar{g}_I^C(x^k) \geq \Psi(x^k + \bar{\lambda}\bar{g}_I^C(x^k))\,. \tag{2.10}$$

Substituting $\tilde{\lambda}$ in $\mu(\lambda)$, we obtain

$$\mu(\tilde{\lambda}) = \Psi(x^k) - \gamma\|\bar{g}_I^C(x^k)\| + \frac{\gamma^2\,\bar{g}_I^C(x^k)H\bar{g}_I^C(x^k)}{2\,\|\bar{g}_I^C(x^k)\|^2}\,. \tag{2.11}$$

Using (2.11) and the fact that $\lambda^* > \tilde{\lambda}$, it follows that

$$\Psi(x^k + \tilde{\lambda}\bar{g}_I^C(x^k)) - \Psi(x^k) < -\frac{\gamma}{2}\|\bar{g}_I^C(x^k)\|\,. \tag{2.12}$$

Combining (2.10) and (2.12), we have

$$\Psi(x^k) - \Psi(x^{k+1}) > \frac{\gamma}{2}\|\bar{g}_I^C(x^k)\| > \frac{\eta\,\gamma}{2}\|\bar{g}_P(x^k)\|\,. \tag{2.13}$$

Now, if $x^k + \lambda^*\bar{g}_I^C(x^k)$ is feasible, then it becomes x^{k+1}, and we obtain

$$\Psi(x^{k+1}) - \Psi(x^k) = -\frac{\|\bar{g}_I^C(x^k)\|^4}{2\,\bar{g}_I^C(x^k)H\bar{g}_I^C(x^k)}\,. \tag{2.14}$$

Using (2.2) and (2.14), we have that

$$\Psi(x^k) - \Psi(x^{k+1}) > \frac{1}{2L}\|\bar{g}_I^C(x^k)\|^2 > \frac{\eta^2}{2L}\|\bar{g}_P(x^k)\|^2 \ . \tag{2.15}$$

Finally, if $\bar{g}_I^C(x^k)^T H \bar{g}_I^C(x^k) \leq 0$, then

$$\mu(\lambda) \leq \Psi(x^k) + \lambda \nabla \Psi(x^k)^T \bar{g}_I^C(x^k) \ ,$$

and $\Psi(x^{k+1}) < \mu(\tilde{\lambda}) = \Psi(x^k) - \gamma\|\bar{g}_I^C(x^k)\|$. Therefore,

$$\Psi(x^k) - \Psi(x^{k+1}) > \gamma\|\bar{g}_I^C(x^k)\| > \eta\gamma\|\bar{g}_P(x^k)\| \ . \tag{2.16}$$

Summing up, there exist three possible cases: $x^k + \lambda^* \bar{g}_I^C(x^k)$ is feasible, or it is not feasible, or $\bar{g}_I^C(x^k)^T H \bar{g}_I^C(x^k) \leq 0$. In each case we obtain, respectively, (2.13), (2.15) and (2.16), and the result holds. $\square$

The global convergence of Algorithm 2.1 is proved in the following theorem.

Theorem 2.2 *Let the infinite sequence $\{x^k\}$ be generated by Algorithm 2.1. Then there exists x^*, a limit point of $\{x^k\}$, such that $\bar{g}_P(x^*) = 0$. If Ψ is convex (H positive semidefinite), all the limit points of $\{x^k\}$ are stationary points of problem (2.1).*

Proof. The sequence $\{x^k\}$ remains in Ω, and Ω is a closed and bounded set of $\mathbb{R}^n$. Hence, the sequence $\{x^k\}$ has limit points. First, we establish that, at least, one of those limit points is a stationary point of (2.1). Suppose, by contradiction, that there exists $\epsilon > 0$ such that

$$\|\bar{g}_P(x^k)\| > \epsilon \ \text{ for all } \ k \ . \tag{2.17}$$

We consider two cases:

(a) The condition (2.9) is satisfied at a finite number of iterations.

(b) There exists an infinite set of indices $K_1 \subset \mathbb{N}$ such that (2.9) is satisfied for all $k \in K_1$.

If (a) holds, let k_0' be greater than the index of the last iteration at which (2.9) is satisfied. It follows that all iterates x^k such that $k \geq k_0'$ belong to the closure of the face that contains $x^{k_0'}$. Moreover, since (2.9) is not satisfied for all $k \geq k_0'$, the dimension of the face that contains x^k is nonincreasing for all $k \geq k_0'$. Then there exists k_0 such that $x^k \in F_I$ for a fixed I, and for all $k \geq k_0$. Hence, the sequence is generated at Step 2, for all $k \geq k_0$, by an unconstrained minimization technique that converges in $\bar{F}_I$. Therefore, $\bar{g}_I(x^k)$ converges to zero as k goes to infinity. Thus, using (2.17), it follows that for some $\bar{k} \geq k_0$, sufficiently large, the condition in Step 1 must be satisfied. This contradicts the assumption (a).

Assume now that (b) holds. Let k_j be the j-th index in K_1, $j \in I\!N$. Using (2.17), Lemma 2.2 and the fact that $\{\Psi(x^k)\}$ is monotonically decreasing, we obtain

$$\begin{aligned}
\Psi(x^{k_j}) - \Psi(x^{k_1}) &= \sum_{l=k_1}^{k_j-1} (\Psi(x^{l+1}) - \Psi(x^l)) \\
&\leq \sum_{l \in K_1,\, l=k_1}^{k_j-1} (\Psi(x^{l+1}) - \Psi(x^l)) \\
&\leq \sum_{l \in K_1,\, l=k_1}^{k_j-1} -\min\{\frac{\eta\,\gamma}{2}\|\bar{g}_P(x^l)\|,\, \frac{\eta^2}{2L}\|\bar{g}_P(x^l)\|^2\} \\
&< -j\, \min\{\frac{\eta\,\gamma}{2}\epsilon,\, \frac{\eta^2}{2L}\epsilon^2\} \qquad (2.18)
\end{aligned}$$

Using (2.18) we conclude that

$$\lim_{j\to\infty} \Psi(x^{k_j}) = -\infty \;,$$

which contradicts the fact that the quadratic function Ψ is bounded below on Ω. Therefore, since $\|\bar{g}_P(\,\cdot\,)\|$ is lower semicontinuous, then at least one of the limit points of $\{x^k\}$, say x^*, is a stationary point of (2.1).

Let $\bar{x} \neq x^*$ be also a limit point of the sequence $\{x^k\}$. Since the sequence $\{\Psi(x^k)\}$ generated by the Algorithm 2.1 is monotonically decreasing, then

$$\Psi(\bar{x}) = \Psi(x^*) \;.$$

If Ψ is convex, both x^* and $\bar{x}$ are global minimizers, then

$$\bar{g}_P(\bar{x}) = 0 \;,$$

and the result is established. $\square$

For the next results of this section, namely Corollaries 2.1, 2.2 and 2.3, and Theorem 2.3, we assume that Ψ is convex.

As a direct consequence of Theorem 2.2, we obtain the following result.

Corollary 2.1 *If Problem (2.1) has a unique solution x^*, then the sequence $\{x^k\}$ generated by Algorithm 2.1 converges to x^*.*

Our next result states that, in the nondegenerate case, Algorithm 2.1 achieves the optimal face in a finite number of iterations if the sequence converges.

Corollary 2.2 *Let $\{x^k\}$ be a convergent sequence generated by Algorithm 2.1, say*

$$\lim_{k\to\infty} x^k = x^* .$$

If x^* is a nondegenerate stationary point then there exists $\hat{k} \in I\!N$ such that $x^k \in F_{I^*}$, i.e., $I(x^k) = I(x^*) = I^*$ for $k \geq \hat{k}$.

Proof. Follows directly from Theorem 2.2 and Corollary 3.6 in Burke and Moré [4]. □

Finally, the next theorem generalizes Corollary 2.2.

Theorem 2.3 *If all the limit points of a sequence $\{x^k\}$ generated by Algorithm 2.1 are non-degenerate, then there exists $\overline{k} \in I\!N$ and a face F_I such that $x^k \in F_I$ for all $k \geq \overline{k}$.*

Proof. Suppose, by contradiction, that the thesis is false. Therefore, x^{k+1} is computed at Step 1 infinitely many times. So, the proposition "$x^{k+1} \notin \bar{F}_{I(x^k)}$" holds for an infinite set of indices k. Since the number of faces is finite, it follows that there exists a face F_J and an infinite set $K_1 \subset I\!N$ such that $x^k \in F_J$ and $x^{k+1} \notin \overline{F}_J$ for $k \in K_1$. Therefore, for all $k \in K_1$, x^{k+1} is obtained at Step 1 of Algorithm 2.1. Since the number of constraints that define F_J is finite, at least one these constraints must be relaxed in an infinite subset $K_2 \subset K_1$. We may suppose, without loss of generality, that this constraint is $x_j = l_j$. So, for $k \in K_2$, we have $j \in I(x^k)$ and $j \notin I(x^{k+1})$. This means that

$$-\frac{\partial \Psi}{\partial x_j}(x^k) > 0. \tag{2.19}$$

Let x^* be a limit point of $\{x^k\}_{k \in K_2}$. Then $x_j^* = l_j$ and, by (2.19),

$$-\frac{\partial \Psi}{\partial x_j}(x^*) \geq 0.$$

By Theorem 2.1, x^* is a stationary point, so we must have $-\frac{\partial \Psi}{\partial x_j}(x^*) = 0$, which contradicts the fact that x^* is nondegenerate. □

Corollary 2.3 *If Algorithm 2.1 generates a sequence $\{x^k\}$ with more than one limit point, then necessarily at least one limit point is degenerate.*

Proof. This is a consequence of Theorem 2.3 and the definition of Step 2 of the algorithm. $\square$

3 The case of dual degeneracy

Throughout this section, we assume that Ψ is convex. If Algorithm 2.1 converges, it has the desirable feature of being able to identify, in the non-degenerate case, the optimal face in a finite number of iterations. In that case, the algorithm reduces eventually to an unconstrained minimization technique, and fast convergence might be observed at the last stage of the process. We now present a modified version of the algorithm that has the same desirable feature even in the presence of dual degeneracy. This algorithm will be denoted, from now on, as Algorithm 3.1, and the only difference with respect to Algorithm 2.1 is at Step 1, which is now given by

Step 1: Given $0 < \eta < 1$, and $\delta > 0$ independently of k,
if

$$\|\bar{g}_I^C(x^k)\| > \eta\, \|\bar{g}_P(x^k)\|\,,$$

and

$$\Psi(\Phi(x^k)) < \Psi(x^k) - \delta\, \|\bar{g}_I(x^k)\|\,, \tag{3.1}$$

then set $x^{k+1} = \Phi(x^k)$. Else go to Step 2.

Steps 2 and 3 in this new algorithm are identical to Steps 2 and 3 in Algorithm 2.1. It is easy to verify that the convergence properties established for Algorithm 2.1, in Section 2, also hold for Algorithm 3.1. In addition, we will establish that the sequence generated by Algorithm 3.1 converges to a solution of (2.1), even in the presence of dual degeneracy and in the case of infinitely many solutions. First, we need to establish the following lemma.

Lemma 3.1 *Assume that $x^k \in F_I$ and that x^{k+1} is obtained at Step 1 of Algorithm 3.1. If $x \in \bar{F}_I$ is such that $\|x - x^k\| \le \delta$, then $\Psi(x^{k+1}) < \Psi(x)$.*

Proof. The convexity of Ψ (see (2.3)) implies that, for all $x \in \bar{F}_I$,

$$\Psi(x) \geq \Psi(x^k) - \bar{g}_I(x^k)^T(x - x^k) \ .$$

Then, using the Cauchy-Schwarz inequality and (3.1), we obtain for all $x \in \bar{F}_I$ such that $\|x - x^k\| \leq \delta$,

$$\Psi(x) \geq \Psi(x^k) - \delta \ \|\bar{g}_I(x^k)\| > \Psi(x^{k+1}) \qquad \square$$

Since the sequence $\Psi(x^k)$ generated by Algorithm 3.1 is monotonically decreasing, then Lemma (3.1) states that when a closed face $\bar{F}_I$ is abandoned, no future iterate produced by Algorithm 3.1, will belong to a ball with center x^k and radius δ in $\bar{F}_I$. This is the key property of Algorithm 3.1 that allows us to establish the strong convergence results in the next theorem. See [14] for more details on the properties of the condition (3.1).

Theorem 3.1 *Let $\{x^k\}$ be the sequence generated by Algorithm 3.1. Then $\{x^k\}$ converges to a global solution x^* of problem (2.1). Moreover, there is a face F_I and $k_0 > 0$ such that $x^k \in F_I$ for all $k \geq k_0$ and $x^* \in \bar{F}_I$. If x^* is not dual-degenerate then $x^* \in F_I$. If the linear solver is the conjugate gradient method or a direct method, there exists $k \geq k_0$ such that $x^k = x^*$.*

Proof. By Lemma 3.1 and the fact that the number of faces in Ω is finite, we have that there exists k_0 and F_I such that $x^k \in F_I$ for all $k \geq k_0$. Hence, from k_0 onward the sequence $\{x^k\}$ is generated by an unconstrained minimization method that converges in $\bar{F}_I$. Using Theorem 2.2 it follows that $\{x^k\}$ converges to a stationary point $x^* \in \bar{F}_I$ of (2.1). So we have established the global convergence and the finite identification even in the presence of dual degeneracy. Finally, if x^* is not dual degenerate, then $x^* \in F_I$ and we also have the identification of the optimal active set for $k \geq k_0$. $\square$

Remark

The role of the condition (3.1) can be better understood considering an example. Let us consider the two-dimensional quadratic defined by

$$\Psi(x) = \frac{1}{2} x^T H x + b^T x$$

with $b = (\varepsilon(1 - M), -1 - M\varepsilon^2)^T$, $H = (h_{ij})$, $h_{11} = M(1 - \varepsilon), h_{12} = h_{21} = \varepsilon(M - 1), h_{22} = M\varepsilon^2 + 1$. For M large enough and ε small enough, Ψ is strictly convex and its global minimizer is $(0,1)^T$. Level sets of Ψ for $M = 10$ and $\varepsilon = 0.3$ are despicted in Figure 1 below.

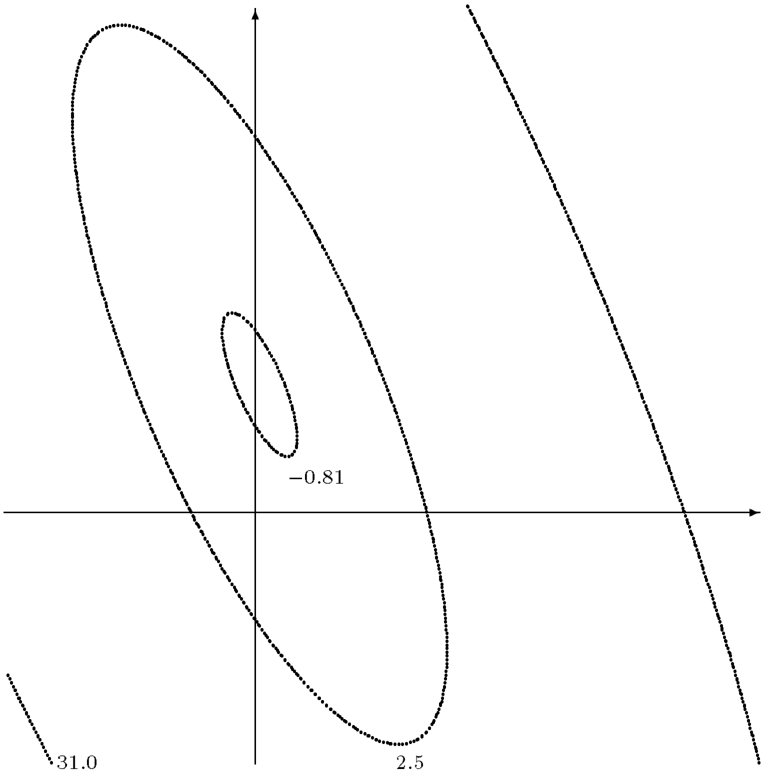

Figure 1: Level sets with degeneracy

Suppose that $\Omega = \{x \in I\!\!R^2 \mid 0 \leq x \leq 100\}$. Then, $(0,1)^T$ is a dual degenerate solution of (2.1). Now, after some elementary calculations, we see that, for all $y < 1$, $\bar{y} = (0, y)^T$,

$$\frac{\|\bar{g}_C(\bar{y})\|}{\|\bar{g}_P(\bar{y})\|} = \frac{\varepsilon(M-1)}{\sqrt{[\varepsilon(M-1)]^2 + (M\varepsilon^2+1)^2}}. \tag{3.2}$$

So,

$$\lim_{M\to\infty} \frac{\|\bar{g}_C(\bar{y})\|}{\|\bar{g}_P(\bar{y})\|} = \frac{1}{\sqrt{1+\varepsilon^2}}. \tag{3.3}$$

So, if

$$\eta < \frac{\varepsilon(M-1)}{\sqrt{[\varepsilon(M-1)]^2 + (M\varepsilon^2+1)^2}} \approx \frac{1}{\sqrt{1+\varepsilon^2}},$$

and $(0, y_k)^T$ is an iterate of Algorithm 2.1 with $y_k < 1$, the criterion (2.9) will indicate to abandon the correct face, independently of the proximity of the iterate to the solution. For example, in the case $M = 10, \varepsilon = 0.3$ this unfortunate decision will take place whenever $\eta < 0.14$. On the other hand, if we also use (3.1), the decision will be to stay in the correct face if $|y_k - 1| < \delta$. Of course, the consequences of incorrect decisions of this type in a two-dimensional problem are not serious, but we can easily develop an example using the ideas above with an arbitrarily large number of variables, where the correct one-dimensional face is abandoned for iterates arbitrarily close to the solution. In this case, if we are using conjugate gradients, the practical consequences can be very serious: either the process becomes infinite or at least n iterations will be necessary to reach the solution. This is the undesirable effect that tends to be eliminated by (3.1).

4 Strategies within the faces

In this section, we describe the different strategies to be used inside the faces in Algorithms 2.1 and 3.1 for convex problems. The main idea is to use iterative methods that require few storage locations and few floating point operations, whenever the dimension of the current face F_I is *large*. If the dimension of F_I is *not so large* and the Hessian H has some preestablished sparsity structure, then we will use the sparse Cholesky factorization to solve the system

$$\hat{H} z^k = -\hat{b} \; , \tag{4.1}$$

and obtain z^k in Step 2. The matrix $\hat{H}$ and the vector $\hat{b}$ are the reductions of H and b associated with the active index I. If the matrix H has no sparsity structure and the dimension of F_I is *small*, we will use the dense Cholesky factorization to solve (4.1), and obtain z^k in Step 2. The terms *large*, *not so large* and *small* will be given a real meaning in Section 5.

If we have used a direct solver to obtain z^k, it is clear that we have found a global minimizer of $\Psi(x)$ subject to $x \in [F_I]$. Hence, $\Psi(z^k) < \Psi(x^k)$ and the condition in Step 2 is satisfied. On the other hand, if we use an iterative method that converges to a minimizer of $\Psi(x)$ in $[F_I]$ then in a finite number of inner steps we must obtain $z^k \in [F_I]$ (not necessarily a minimizer) such that $\Psi(z^k) < \Psi(x^k)$, and once again the condition in Step 2 is satisfied.

The iterative methods considered in this work are the conjugate gradient (CG) method of Hestenes and Stiefel [19], and the family of gradient method with retards (GR) recently introduced by Friedlander, Martínez and Raydan [16] that have global convergence for the unconstrained minimization of convex quadratic functions. The GR family can be viewed as a generalization of the classical steepest descent and the Barzilai-Borwein [2] methods. Given a positive integer m, the GR iteration is given by

$$x^{k+1} = x^k - \lambda(x^{\nu(k)})\bar{g}_I(x^k)\,, \tag{4.2}$$

where $\nu(k)$ might be arbitrary chosen in the set $\{k, k-1, \ldots, \max\{0, k-m\}\}$, and $\lambda(x)$ is the minimizer of $\Psi(x - \lambda\,\bar{g}_I(x))$. For instance, if $\nu(k) = k$ then (4.2) becomes the steepest descent method, and if $\nu(k) = k-1$ then we obtain the Barzilai-Borwein method. In the experimental research of Section 5, we define $\bar{k} = \max\{0, k-m\}$, and we consider, besides the Barzilai-Borwein method, the following choice:

$\nu(k) =$ random between $\bar{k}$ and k (random);

Now, whether we use iterative methods or direct solvers to obtain z^k, it might happen that z^k is not in $\bar{F}_I$. In that case, at Step 3 of both algorithms, we have to find $x^{k+1} \in \bar{F}_I - F_I$ such that $\Psi(x^{k+1}) < \Psi(x^k)$. This Step will be implemented using a backtracking search along the polygonal path defined by the search direction. For further details in how to perform this step in the CG case, see [14]. In the case of the Barzilai-Borwein method the details can be found in [15]. Since the nonmonotonicity is present in all the GR methods, the same technique, for performing Step 3, can be applied for the entire family of GR methods.

5 Numerical Results

In our numerical experiments, the criterion for deciding on the algorithm to be used for unconstrained steps within the faces F_I was based only on the dimension of the face. Moreover, although gradient methods with retards can be used in larger problems than conjugate gradients, we decided to use any of them when the dimension of the face is *large* and to study the relative efficiency of different choices. Finally, although the relative efficiency of dense Cholesky factorization relative to sparse Cholesky factorization depends on the structure of the problem, we decided to use always "dense Cholesky" when the dimension of the face is less than or equal to 30 and "sparse Cholesky" when the dimension is larger than 30.

We also decided to study the efficiency and reliability of the methods as depending of the parameter η at Step 1 of Algorithm 2.1. Finally, the modification of Algorithm 2.1 that uses condition (3.1) was also tested against the "pure" algorithm. Summing up, we began our tests trying to answer the following questions:

(a) For what dimension is it convenient to pass from Cholesky's factorizations to iterative methods for minimization within the faces?

(b) How do gradient methods with retards compare between them and against conjugate gradients?

(c) Which is the best value for the parameter η at Step 1 of Algorithm 2.1?

(d) Is the modification based on (3.1) worthwhile from a practical point of view?

Ideally, we should like to detect, on average, the best values for η, δ, *dimchol* (the face dimension under which we use Cholesky) and the best iterative linear method within the faces. However, to test all the possibilities for each problem would lead to an unacceptable large number of experiments. So, for each set of problems, we proceeded performing four sequences of experiments:

(a) Using $\delta = 0$, $dimchol = 0$ and conjugate gradients, we determined the "best η" trying the values 10^{-10}, 10^{-2}, 0.1 and 0.9 for this parameter.

(b) Using $dimchol = 0$, conjugate gradients, and the best η obtained in (a), we determined the "best δ" testing the values 0, 10^{-4}, 0.1 and 1.

(c) Using conjugate gradients, and the best η and δ obtained in (a) and (b), we determined the "best *dimchol*" testing the values 0, 10, 100 and 1000.

(d) Using the best η, δ and *dimchol* obtained in (a), (b) and (c), we determined the "best linear iterative method", trying conjugate gradients, Barzilai-Borwein, gradient method with random retard ($m = 3$) and the gradient method with random retard ($m = 6$).

All the codes were written in Fortran and ran in a PC-486 with a clock of 66 MHz.

Test 1. Obstacle problems. See [5] (pp. 287-296), [9, 14, 22]. This problem consists of finding the equilibrium position of an elastic membrane which passes through a curve Γ. We considered the three problems described in [9], which correspond to different bounds on the solution. We denote by $u(x, y)$ and $v(x, y)$ the lower obstacle function and the upper obstacle function, respectively. In Case A we set

$$u(x,y) = p_1[\sin(3.2x)\sin(3.3y)]^{p_2}$$

and

$$v(x,y) = 2000.$$

In Case B,

$$u(x,y) = [\sin(9.2x)\sin(9.3y)]^{p_2}$$

and

$$v(x,y) = u(x,y) + 0.02.$$

In Case C,

$$u(x,y) = [16x(1-x)y(1-y)]^{p_1}$$

and

$$v(x,y) = u(x,y) + 0.01.$$

We used the five-point finite-difference approximation to the Laplace operator. After discretization, the obstacle problem becomes a quadratic programming problem with box constraints. We used the following starting points:

(a) $x^0 = \ell = (\ell_1, \ldots, \ell_n)^T$;
(b) $x^0 = \underline{1} = (1, \ldots, 1)^T$;
(c) $x^0 = u = (u_1, \ldots, u_n)^T$.
(d) $x^0 = (\ell + u)/2$.

Finally, we defined 30 obstacle problems, as reported in the following Table 1.

Problem	Case	n	x^0	p_1	p_2
1	A	2601	ℓ	1	1
2				0	1
3				1	2
4				1	3
5			$\underline{1}$	1	1
6				0	1
7				1	2
8				1	3
9	A	5041	ℓ	1	1
10				0	1
11				1	2
12				1	3
13			$\underline{1}$	1	1
14				0	1
15				1	2
16				1	3
17	A	10000	ℓ	1	1
18				0	1
19				1	2
20				1	3
21			$\underline{1}$	1	1
22				0	1
23				1	2
24				1	3
25	B	5041	u	3	2
26			ℓ	3	2
27			$(\ell+u)/2$	3	2
28	C	5041	u	3	2
29			ℓ	3	2
30			$(\ell+u)/2$	3	2

Table 1: Definition of 30 obstacle problems

In Table 1.a, we report the results of the family of experiments (a), which consists in fixing $\delta = 0, dimchol = 0$ and conjugate gradients, varying η. For each experiment, we report $(it, time)$ where it is the number of iterations that were necessary to achieve the convergence criterion

$$\|\bar{g}_P(x^k)\| \leq TOL\|\bar{g}_P(x^0)\|, \tag{4.3}$$

(with $TOL = 10^{-5}$) and $time$ is the computer time, in seconds.

Problem	$\eta = 10^{-10}$	$\eta = 10^{-2}$	$\eta = 0.1$	$\eta = 0.9$
1	127, 5.80	88, 3.90	76, 3.20	75, 3.10
2	115, 5.30	78, 3.50	65, 2.80	64, 2.60
3	202, 8.80	153, 6.80	116, 5.40	116, 5.40
4	229, 10.20	152, 7.00	134, 6.40	137, 6.60
5	250, 11.40	212, 9.60	128, 5.90	146, 6.70
6	288, 13.20	254, 11.70	279, 13.00	175, 8.10
7	259, 11.80	211, 9.70	182, 8.70	217, 10.80
8	240, 11.40	229, 11.00	218, 10.80	274, 14.00
9	200, 19.10	148, 13.80	119, 10.70	120, 10.30
10	211, 20.60	123, 11.70	75, 6.90	96, 8.40
11	330, 30.80	262, 25.40	168, 16.90	166, 17.00
12	401, 37.50	286, 28.00	165, 17.00	219, 22.80
13	314, 30.60	328, 31.60	215, 21.50	171, 16.50
14	410, 39.10	364, 35.40	499, 49.70	478, 47.30
15	413, 39.20	356, 35.40	279, 29.00	335, 36.20
16	406, 39.50	392, 39.50	362, 37.70	388, 42.50
17	428, 90.60	220, 45.20	154, 30.30	171, 32.10
18	408, 87.60	181, 37.40	146, 29.30	158, 29.90
19	813, 165.40	430, 91.60	355, 79.30	230, 51.80
20	742, 150.90	499, 107.40	404, 91.10	314, 72.20
21	1226, 257.20	676, 147.70	351, 74.10	275, 59.30
22	1850, 388.80	1046, 224.30	732, 165.20	584, 129.80
23	765, 159.90	487, 107.90	552, 127.90	472, 114.50
24	850, 179.70	604, 134.10	680, 157.90	649, 159.10
25	454, 53.70	281, 34.00	190, 24.40	166, 21.20
26	490, 57.10	289, 34.90	222, 27.70	223, 28.10
27	234, 28.50	170, 21.50	143, 18.70	113, 15.30
28	276, 31.90	204, 23.80	181, 21.80	191, 23.00
29	187, 21.90	184, 22.10	184, 23.00	195, 24.90
30	227, 27.20	154, 18.40	142, 17.90	128, 16.60
Average	445, 67.80	302, 44.50	251, 37.80	235, 34.50

Table 1.a: Obstacle problems: variation of η

In Table 1.b we report the numerical results fixing $\eta = 0.1, dimchol = 0$ and conjugate gradients, for different values of δ. We fixed $\eta = 0.1$ (instead of 0.9) because the first outperformed the second a (slightly) larger number of times.

Problem	$\delta = 0$	$\delta = 10^{-4}$	$\delta = 0.1$	$\delta = 1$
1	76, 3.00	108, 5.40	370, 23.00	557, 34.90
2	65, 2.60	365, 22.60	698, 43.90	993, 62.70
3	116, 5.20	222, 12.60	1167, 79.40	1916, 130.60
4	134, 6.20	144, 7.30	580, 39.60	1232, 84.80
5	128, 5.60	152, 7.50	503, 33.30	656, 43.20
6	279, 12.60	544, 30.40	518, 34.90	589, 39.40
7	182, 8.40	200, 9.90	713, 48.90	1111, 76.40
8	218, 10.40	225, 10.90	566, 38.70	768, 52.70
9	119, 10.70	178, 20.90	782, 112.30	1254, 180.40
10	75, 7.00	182, 23.80	778, 111.30	1163, 166.60
11	168, 16.90	364, 48.80	2001, 305.60	2001, 304.70
12	165, 17.10	500, 70.80	1223, 186.60	2001, 306.70
13	215, 21.90	313, 38.20	1201, 181.40	1271, 191.60
14	499, 50.90	808, 98.70	1122, 169.70	1284, 193.30
15	279, 29.20	852, 121.20	1497, 232.10	2001, 311.50
16	362, 38.40	595, 79.40	1152, 180.10	1645, 257.40
17	154, 31.80	867, 280.60	1646, 548.60	2001, 671.40
18	146, 30.80	748, 245.60	2001, 672.30	2001, 672.50
19	355, 82.20	689, 215.10	2001, 690.90	2001, 684.70
20	404, 94.30	556, 167.50	2001, 697.00	2001, 690.30
21	351, 77.40	708, 208.20	2001, 699.20	2001, 698.50
22	732, 171.90	1321, 383.70	1886, 654.90	2001, 697.50
23	552, 132.70	973, 292.00	2001, 714.10	2001, 714.10
24	680, 163.80	949, 284.20	2001, 715.20	2001, 714.90
25	190, 24.40	361, 61.10	1608, 311.00	2001, 387.00
26	222, 27.70	529, 92.60	789, 149.10	1342, 255.60
27	143, 18.70	676, 130.90	2001, 402.90	2001, 402.60
28	181, 21.80	708, 124.30	2001, 371.10	2001, 368.90
29	184, 23.10	614, 110.70	1363, 257.60	2001, 377.70
30	142, 17.90	647, 119.50	1007, 193.30	1463, 283.00
Average	251, 38.80	537, 110.80	1053, 296.60	1203, 335.20

Table 1.b: Obstacle problems: variation of δ

The experiments in Table 1.b with 2001 iterations represent cases where convergence was not achieved. We included these cases in the Average row since they do not affect the overall conclusion, which largely favors $\delta = 0$.

In Table 1.c we report the numerical results fixing $\eta = 0.1, \delta = 0$ and conjugate gradients, for different values of *dimchol*.

Problem	$dimchol = 0$	$dimchol = 10$	$dimchol = 100$	$dimchol = 1000$
1	76, 3.80	76, 3.20	76, 3.20	17, 18.00
2	65, 2.80	65, 2.80	65, 2.80	15, 14.60
3	116, 5.60	116, 5.60	116, 5.60	116, 5.60
4	134, 6.60	134, 6.60	134, 6.60	134, 6.60
5	128, 6.00	128, 6.00	128, 6.00	68, 23.00
6	279, 13.20	279, 13.20	279, 13.20	210, 30.90
7	182, 8.80	182, 8.80	182, 8.80	182, 8.80
8	218, 10.90	218, 10.90	218, 10.90	218, 10.90
9	119, 11.20	119, 10.90	119, 10.90	106, 29.80
10	75, 7.00	75, 7.00	75, 7.00	76, 27.00
11	168, 17.50	168, 17.50	168, 17.50	168, 17.50
12	165, 17.60	165, 17.60	165, 17.60	165, 17.60
13	215, 22.30	215, 22.30	215, 22.30	209, 30.70
14	499, 51.60	499, 51.60	499, 51.60	499, 51.60
15	279, 30.10	279, 30.10	279, 30.10	279, 30.10
16	362, 39.10	362, 39.10	362, 39.10	362, 39.10
17	154, 31.50	154, 31.00	154, 31.00	188, 69.20
18	146, 30.00	146, 30.00	146, 30.00	133, 60.00
19	355, 80.50	355, 80.50	355, 80.50	355, 80.50
20	404, 92.40	404, 92.40	404, 92.40	404, 92.40
21	351, 75.40	351, 75.40	351, 75.40	351, 75.40
22	732, 167.60	732, 167.60	732, 167.60	708, 197.40
23	552, 129.40	552, 129.40	552, 129.40	552, 129.40
24	680, 159.90	680, 159.90	680, 159.90	680, 159.90
25	190, 24.40	190, 24.40	190, 24.40	190, 24.40
26	222, 27.70	222, 27.70	222, 27.70	222, 27.70
27	143, 18.70	143, 18.70	143, 18.70	143, 18.70
28	181, 21.80	181, 21.80	181, 21.80	181, 21.80
29	184, 23.00	184, 23.00	184, 23.00	184, 23.00
30	142, 17.90	142, 17.90	142, 17.90	142, 17.90
Average	251, 38.50	251, 38.40	251, 38.40	242, 45.30

Table 1.c: Obstacle problems: variation of *dimchol*

Observe that the executions with $dimchol \leq 100$ are identical. This means that all the faces visited have more than 100 free variables.

In Table 1.d we report the numerical results fixing $\eta = 0.1, \delta = 0, dimchol = 0$, for different iterative linear methods.

Problem	Conjugate gradients	Barzilai-Borwein	Retard $m = 3$	Retard $m = 6$
1	76, 3.00	88, 3.90	126, 5.80	100, 4.50
2	65, 2.60	88, 3.90	122, 5.40	98, 4.40
3	116, 5.40	143, 7.20	182, 9.00	219, 11.00
4	134, 6.40	201, 10.40	223, 11.90	212, 10.40
5	128, 5.60	170, 8.50	185, 9.20	160, 8.10
6	279, 12.50	201, 9.80	212, 11.00	219, 10.80
7	182, 8.60	232, 12.00	225, 11.50	258, 12.90
8	218, 10.70	216, 11.60	260, 13.60	329, 16.40
9	119, 10.20	149, 15.20	171, 17.40	146, 14.90
10	75, 6.60	112, 11.40	137, 13.70	165, 16.20
11	168, 16.30	251, 28.20	252, 29.90	280, 31.80
12	165, 16.40	251, 28.80	286, 33.30	294, 34.10
13	215, 20.70	215, 24.00	232, 26.10	280, 30.00
14	499, 48.10	284, 32.90	343, 38.50	308, 32.70
15	279, 28.00	302, 35.30	343, 40.10	374, 43.60
16	362, 36.50	379, 44.90	401, 47.20	398, 46.30
17	154, 30.30	232, 53.50	224, 51.70	267, 61.90
18	146, 29.30	173, 39.60	201, 46.50	223, 50.70
19	355, 79.40	292, 77.10	362, 95.20	341, 87.30
20	404, 91.20	363, 93.80	417, 114.10	408, 102.20
21	351, 74.00	320, 81.40	343, 87.20	379, 92.90
22	732, 165.30	427, 110.10	549, 146.60	466, 117.90
23	552, 127.90	419, 112.40	515, 139.30	586, 155.90
24	680, 158.10	551, 149.60	577, 156.30	601, 160.50
25	190, 24.40	222, 32.70	253, 37.30	273, 39.50
26	222, 27.70	195, 29.50	214, 31.90	266, 40.00
27	143, 18.70	162, 23.90	176, 26.90	175, 26.90
28	181, 21.80	187, 27.00	225, 32.80	255, 36.30
29	184, 23.00	196, 29.30	226, 33.10	238, 33.50
30	142, 17.90	176, 25.70	185, 28.40	220, 33.80
Average	251, 37.60	240, 39.10	272, 45.00	285, 45.60

Table 1.d: Obstacle problems: variation of the linear solver

Test 2. Function reconstruction from line integrals. See [14]. Assume that the interval $[0,1]$ is divided into $ndiv$ intervals of equal size, $h = 1/ndiv$, and, consequently, the square $[0,1] \times [0,1]$ is divided into $n \equiv ndiv^2$ pixels. In our experiments $ndiv = 256$, so $n = 65586$. The problem is to find a function $u : [0,1] \times [0,1] \to [\ell, u]$, which is constant on each pixel, such that the line integrals along certain rays are known.

We generate the following sets of rays:

a) Horizontal rays: $ndiv$ rays, from $(0, (i-1)h + h/2)$ to $(1, (i-1)h + h/2), i = 1, \ldots, ndiv$.

b) Vertical rays: $ndiv$ rays, from $((i-1)h+h/2, 0)$ to $((i-1)h+h/2, 1), i = 1, \ldots, ndiv$.

c) Rays forming an angle of 135 degrees with the $x - axis$:

i) $ndiv$ rays, from $(ih, 0)$ to $(0, ih), i = 1, \ldots, ndiv$.

ii) $ndiv - 1$ rays, from $(1, ih)$ to $(ih, 1), i = 1, \ldots, ndiv - 1$.

d) Rays forming an angle of 45 degrees with the $x - axis$:

i) $ndiv$ rays, from $((i-1)h, 0)$ to $(1, 1-(i-1)h), i = 1, \ldots, ndiv$.

ii) $ndiv - 1$ rays, from $(0, ih)$ to $(1 - ih, 1), i = 1, \ldots, ndiv - 1$.

Each ray defines one equation where only the variables corresponding to the pixels that are crossed by the ray are involved. Therefore, we have a sparse linear $m \times n$ system $\bar{A}x = d$, where $m = 6ndiv - 2$ and $n = ndiv^2$ with the constraints $\ell \leq x_i \leq u$ for all $i = 1, \ldots, n$. The unknowns x_i correspond to the values of the unknown function u on each pixel, numbered lexicographically. The quadratic function to be minimized is $\|\bar{A}x - d\|_2^2$.

For generating the data (vector d) we define a function

$$u_* : [0,1] \times [0,1] \to [0,1]$$

and a function $\overline{u}$ which is constant on each pixel and coincides with u_* at the center of the pixel. The data vector d is formed by the line integrals of $\overline{u}$ along the rays. So, different experiments are characterized by different choices of u_*. We used the following functions:

Function 1

$u_*(x,y) \equiv u_1(x,y) = 1$ if $0.25 \leq x \leq 0,75$ and $0.25 \leq y \leq 0.75$; 0 otherwise.

Function 2

$$u_*(x,y) \equiv u_2(x,y) = (x^2 + y)/2.$$

Function 3

$$u_*(x,y) = \min \{1, u_1(x,y) + u_2(x,y)\}$$

Tables 2.a, 2.b and 2.c were constructed using similar criteria as those used in previous tables. However, after the experiments reported in Table 2.a, we fixed $\eta = 0.9$ instead of $\eta = 0.1$. We do not report experiments with different values of *dimchol* because, in this case, no faces are visited with dimension inferior to 6000. Therefore, direct methods are not used at all. Observe that, in this case, the involved matrices are, in general, singular, so we were prepared to use a Cholesky factorization that included a small regularization parameter. For consistency with experiments in [14], we used (4.3) with $TOL = 10^{-4}$.

Problem	$\eta = 10^{-10}$	$\eta = 10^{-2}$	$\eta = 0.1$	$\eta = 0.9$
1	70, 359.90	68, 323.20	47, 223.50	35, 159.20
2	22, 115.00	22, 114.70	19, 104.20	19, 104.20
3	222, 1015.5	144, 704.80	130, 653.00	147, 738.10
Average	105, 496.80	78, 380.90	65, 326.90	67, 333.80

Table 2.a: Reconstruction problems: variation of η

Problem	$\delta = 0$	$\delta = 10^{-4}$	$\delta = 0.1$	$\delta = 1$
1	35, 159.10	35, 159.40	35, 159.30	35, 159.40
2	19, 104.00	19, 104.10	19, 104.10	19, 104.10
3	147, 738.00	147, 738.30	147, 738.00	147, 738.50
Average	67, 333.70	67, 333.90	67, 333.80	67, 334.00

Table 2.b: Reconstruction problems: variation of δ using $\eta = 0.9$, $dimchol = 0$ and conjugate gradients

Problem	Conjugate gradients	Barzilai-Borwein	Retard $m = 3$	Retard $m = 6$
1	35, 159.30	23, 148.90	26, 159.60	26, 158.10
2	19, 106.00	27, 177.60	28, 193.70	25, 165.70
3	147, 738.70	35, 226.00	45, 290.20	47, 298.00
Average	67, 334.70	28, 184.20	33, 214.50	33, 207.30

Table 2.c: Reconstruction problems: variation of the linear iterative method using $\eta = 0.9, \delta = 0, dimchol = 0$

Test 3. Random problems. This is the set of problems described in [14] and [22], with $n = 1000$. We call *nsing* the number of zero eigenvalues of the Hessian. In the nonsingular cases, the condition number is 1000.

Dual degenerate problems depend on a parameter $kdeg$, which indicates the number of zeros of the gradient at x^* corresponding to active components. Problems with $kdeg > 0$ are degenerate. If $kdeg = 0$, the parameter $ndeg$ is a measure of "near-degeneracy". The greater the value of $ndeg$ the more "near degenerate" the problem is. Finally, $na(x^*)$ and $na(x^0)$ represent the number of active bounds at x^* and x^0 respectively.

We have 22 problems of this class, which are defined in Table 3.

Problem	$nsing$	$kdeg$	$na(x^*)$	$na(x^0)$	$ndeg$
1	0	0	100	100	1
2					12
3				500	1
4					12
5				900	1
6					12
7			500	100	1
8					12
9				500	1
10					12
11				900	1
12					12
13			900	100	1
14					12
15				500	1
16					12
17				900	1
18					12
19	238	0	900	100	0
20		431			
21		719			
22	740	0			

Table 3: Definition of 22 random problems

Tables 3.a, 3.b and 3.c and 3.d were constructed using similar criteria as tables 1.a, 1.b, 1.c and 1.d. The stopping criterion was the same used in the obstacle problems.

Problem	$\eta = 10^{-10}$	$\eta = 10^{-2}$	$\eta = 0.1$	$\eta = 0.9$
1	355, 9.90	299, 8.60	285, 8.30	283, 8.30
2	764, 21.30	490, 14.10	406, 12.20	329, 9.60
3	320, 9.10	299, 8.60	298, 8.60	307, 9.00
4	765, 21.90	634, 17.80	480, 14.00	309, 9.10
5	362, 10.10	328, 9.30	315, 9.10	316, 9.20
6	838, 24.20	514, 14.40	470, 13.60	306, 9.00
7	375, 9.70	382, 9.80	336, 8.80	294, 7.60
8	1856, 52.50	660, 19.70	570, 15.70	377, 9.80
9	407, 10.40	373, 9.70	361, 9.30	310, 8.00
10	6827, 181.50	638, 18.20	543, 14.80	393, 10.20
11	397, 10.20	367, 9.50	355, 9.20	327, 8.40
12	3808, 101.50	650, 17.90	582, 16.20	374, 9.70
13	322, 7.70	350, 5.90	244, 5.70	240, 5.50
14	1690, 53.00	550, 15.80	495, 14.30	429, 9.80
15	328, 7.90	304, 7.20	287, 6.70	251, 5.70
16	3011, 90.60	673, 19.20	592, 16.60	421, 9.60
17	376, 9.10	309, 7.40	276, 6.40	301, 6.80
18	9526, 282.20	559, 15.40	700, 18.70	397, 8.90
19	250, 6.00	246, 5.90	248, 5.90	241, 5.70
20	4089, 117.90	502, 14.10	350, 9.00	305, 7.30
21	797, 22.70	482, 14.00	486, 14.30	328, 8.00
22	158, 4.00	158, 4.00	110, 2.70	134, 3.20
Average	1710, 48.30	444, 12.10	400, 10.90	317, 8.10

Table 3.a: Random problems: variation of η

Problem	$\delta = 0$	$\delta = 10^{-4}$	$\delta = 0.1$	$\delta = 1$
1	283, 8.30	283, 8.30	3295, 145.20	1785, 77.20
2	329, 9.60	329, 9.60	2199, 95.00	2553, 111.40
3	307, 9.00	307, 9.00	3223, 142.00	1641, 69.90
4	309, 9.10	309, 9.10	336, 10.70	6264, 279.70
5	316, 9.20	316, 9.20	4521, 200.10	2352, 102.70
6	306, 9.00	306, 9.00	653, 24.90	4516, 200.20
7	294, 7.60	294, 7.60	2219, 93.30	3121, 133.30
8	377, 9.80	377, 9.80	758, 26.30	1857, 76.60
9	310, 8.00	310, 8.00	328, 9.50	2030, 84.80
10	393, 10.20	393, 10.20	393, 10.20	1584, 64.30
11	327, 8.40	327, 8.40	1053, 41.10	2379, 98.60
12	374, 9.70	374, 9.70	673, 23.50	1336, 54.40
13	240, 5.50	240, 5.50	927, 35.30	1793, 71.50
14	429, 9.80	429, 9.80	459, 12.30	1222, 47.20
15	251, 5.70	251, 5.70	301, 7.70	613, 21.10
16	421, 9.60	421, 9.60	486, 14.00	566, 18.10
17	301, 6.90	301, 6.90	979, 36.10	2884, 116.50
18	397, 8.90	397, 8.90	541, 14.60	2843, 117.10
19	241, 5.70	241, 5.70	479, 15.90	1533, 62.20
20	305, 7.30	305, 7.30	739, 26.00	1991, 83.00
21	328, 8.00	328, 8.00	358, 11.10	1018, 39.30
22	134, 3.20	134, 3.20	285, 10.00	678, 27.20
Average	317, 8.10	317, 8.10	1146, 45.70	2116, 88.90

Table 3.b: Random problems: variation of δ using $\eta = 0.9$, $dimchol = 0$ and conjugate gradients

Problem	$dimchol = 0$	$dimchol = 10$	$dimchol = 100$	$dimchol = 1000$
1	283, 8.30	283, 8.30	283, 8.30	11, 840.80
2	329, 9.70	329, 9.70	329, 9.70	9, 773.90
3	307, 9.00	307, 9.00	307, 9.00	8, 496.80
4	309, 9.10	309, 9.10	309, 9.10	8, 659.00
5	316, 9.20	316, 9.20	316, 9.20	10, 672.60
6	306, 9.00	306, 9.00	306, 9.00	12, 955.90
7	294, 7.60	294, 7.60	294, 7.60	7, 271.90
8	377, 9.80	377, 9.80	377, 9.80	8, 394.20
9	310, 8.00	310, 8.00	310, 8.00	13, 305.60
10	393, 10.20	393, 10.20	393, 10.20	17, 552.80
11	327, 8.40	327, 8.40	327, 8.40	15, 341.50
12	374, 9.70	374, 9.70	374, 9.70	20, 608.50
13	240, 5.50	240, 5.50	240, 5.50	10, 199.20
14	429, 9.80	429, 9.80	429, 9.80	16, 288.40
15	251, 5.70	251, 5.70	251, 5.70	11, 123.30
16	421, 9.60	421, 9.60	421, 9.60	15, 217.70
17	301, 6.90	301, 6.80	301, 6.80	18, 161.00
18	397, 8.90	397, 8.90	397, 8.90	16, 226.10
19	241, 5.70	241, 5.70	241, 5.70	71, 1309.7
20	305, 7.30	305, 7.30	305, 7.30	51, 1052.4
21	328, 8.00	328, 8.00	328, 8.00	58, 1338.9
22	134, 3.20	134, 3.20	102, 28.60	35, 467.20
Average	317, 8.10	317, 8.10	315, 9.30	20, 557.20

Table 3.c: Random problems: variation of *dimchol* using $\eta = 0.9$, $\delta = 0$ and conjugate gradients

Problem	Conjugate gradients	Barzilai-Borwein	Retard $m = 3$	Retard $m = 6$
1	283, 8.30	379, 11.50	417, 12.70	466, 12.50
2	329, 9.60	426, 12.80	468, 13.20	618, 15.60
3	307, 9.00	434, 13.00	412, 12.10	395, 11.40
4	309, 9.10	388, 12.00	394, 11.50	456, 12.90
5	316, 9.20	434, 13.30	405, 12.20	432, 11.80
6	306, 9.00	366, 11.40	390, 11.00	426, 12.70
7	294, 7.60	463, 13.00	511, 13.60	382, 10.10
8	377, 9.80	450, 12.40	451, 11.90	523, 13.50
9	310, 8.00	487, 13.70	494, 12.90	422, 11.40
10	393, 10.20	464, 12.80	381, 10.60	492, 12.60
11	327, 8.40	507, 13.30	497, 13.80	623, 15.00
12	374, 9.70	462, 13.00	510, 13.80	425, 10.90
13	240, 5.50	368, 9.60	385, 9.60	405, 9.90
14	429, 9.80	535, 13.80	413, 10.50	361, 9.00
15	251, 5.70	408, 10.40	382, 9.50	472, 11.20
16	421, 9.60	501, 12.90	501, 12.20	376, 9.30
17	301, 6.80	347, 9.10	383, 9.80	479, 11.30
18	397, 8.90	384, 9.90	453, 11.40	435, 10.80
19	241, 5.70	483, 12.10	466, 11.40	592, 13.60
20	305, 7.30	384, 10.30	369, 9.80	319, 7.80
21	328, 8.00	444, 12.40	377, 10.40	469, 12.00
22	134, 3.20	333, 9.00	354, 8.70	266, 6.70
Average	317, 8.10	429, 11.90	428, 11.50	447, 11.50

Table 3.d: Random problems: variation of the linear iterative method using $\eta = 0.9, \delta = 0, dimchol = 0$

Test 4. Projections on polytopes. The projection of $y \in I\!R^n$ on a polytope given by a set of inequalities is the solution of

$$\text{Minimize } \frac{1}{2}\|w - y\|^2$$

$$\text{subject to } Aw \leq d.$$

Using duality, this problem is transformed in the following box constrained convex minimization problem:

$$\text{Minimize } \frac{1}{2}x^T AA^T x - (Ay - d)^T x$$

$$\text{subject to } x \geq 0,$$

where $w = y - A^T x$.

We consider two problems of this type:

$$\text{Minimize } \frac{1}{2}\|w - y\|^2 \text{ s.t } w_i \leq w_{i+1}, \tag{4.4}$$

$i = 1, \ldots, n-1$, and

$$\text{Minimize } \frac{1}{2}\|w - y\|^2 \text{ s.t } w_i \leq \frac{w_{i-1} + w_{i+1}}{2}, \tag{4.5}$$

$i = 2, \ldots, n-1$. For each problem, we used the vector y given by $y_i = y_j(0.01i) + \varepsilon_i, i = 1, \ldots, 100$, where ε_i is a random number belonging to $(-0.1, 0.1)$ and $y_j, j = 1, \ldots, 5$ are the following functions:

(a) $y_1(z) = z$;

(b) $y_2(z) = \log(z + 0.01)$;

(c) $y_3(z) = \sin(1.5z)$;

(d) $y_4(z) = 1/(1 + 9\exp(-6z))$;

(e) $y_5(z) = 1.6z^2 - 0.7z + 0.1$.

Consequently, we have 10 problems, the first five correspond to (4.4) and the last five correspond to (4.5), for the functions $y_1, \ldots, y_5$ defined above.

Tables 4.a, 4.b and 4.c are constructed using the criteria of the previous ones. Comparison of iterative linear methods is not meaningful here, because, as shown in Table 4.c, the direct method is much better than iterative ones in this case. In these tables, 50001 iterations correspond to cases where convergence was not achieved. The Average rows in Table 4.a were computed considering only the cases of convergence. In Table 4.b we computed the averages only for the two first columns since these outperform clearly the last two. Finally, in Table 4.c we computed the averages only for the last column, due to the same reasons as in Table 4.b. In (4.4) we used (4.3) with $TOL = 10^{-6}$ and in (4.5) we used $TOL = 10^{-3}$.

Problem	$\eta = 10^{-10}$	$\eta = 10^{-2}$	$\eta = 0.1$	$\eta = 0.9$
1	47, 0.09	47, 0.09	40, 0.08	40, 0.08
2	82, 0.17	82, 0.17	92, 0.19	88, 0.19
3	140, 0.30	132, 0.29	125, 0.28	180, 0.40
4	107, 0.24	107, 0.24	108, 0.25	178, 0.40
5	184, 0.40	140, 0.31	161, 0.35	198, 0.44
6	138, 0.31	137, 0.31	139, 0.31	184, 0.42
7	78, 0.16	77, 0.16	81, 0.17	71, 0.15
8	167, 0.33	137, 0.28	130, 0.26	119, 0.25
9	61, 0.12	61, 0.12	54, 0.11	51, 0.11
10	145, 0.30	164, 0.34	128, 0.27	127, 0.27
11	7148, 18.10	5706, 14.40	5538, 14.10	7327, 18.70
12	8418, 21.50	7931, 20.30	8395, 21.50	8311, 21.30
13	5191, 12.90	4625, 11.50	4589, 11.50	4148, 10.40
14	4882, 12.20	4556, 11.40	5307, 13.30	4043, 10.20
15	3615, 9.00	3843, 9.60	3712, 9.30	2860, 7.20
16	3740, 9.40	3250, 8.20	3633, 9.20	3674, 9.30
17	50001, 126.90	50001, 126.80	50001, 127.30	50001, 127.50
18	50001, 127.20	50001, 127.20	50001, 127.60	50001, 127.50
19	3859, 9.60	3733, 9.30	3722, 9.30	3652, 9.20
20	3605, 9.10	3122, 7.90	3402, 8.60	3397, 8.60
Average	2311, 5.79	2103, 5.27	2186, 5.50	2147, 5.42

Table 4.a: Projection problems: variation of η

Problem	$\delta = 0$	$\delta = 10^{-4}$	$\delta = 0.1$	$\delta = 1$
1	47, 0.09	46, 0.10	95, 0.27	113, 0.33
2	82, 0.17	94, 0.21	159, 0.47	216, 0.64
3	132, 0.29	132, 0.29	238, 0.63	765, 2.29
4	107, 0.24	107, 0.24	300, 0.85	492, 1.45
5	140, 0.31	172, 0.38	595, 1.75	1016, 3.06
6	137, 0.31	137, 0.31	521, 0.52	914, 2.75
7	77, 0.16	93, 0.21	381, 1.12	373, 1.11
8	137, 0.28	407, 1.13	552, 1.67	772, 2.36
9	61, 0.12	68, 1.54	126, 0.35	208, 0.62
10	164, 0.34	190, 0.41	482, 1.45	676, 0.25
11	5706, 14.40	7003, 18.50	50001, 187.30	50001, 186.90
12	7931, 20.30	8323, 21.40	8323, 21.40	8323, 21.40
13	4625, 11.50	5458, 14.90	50001, 186.80	50001, 186.90
14	4556, 11.40	7422, 20.70	50001, 68.40	50001, 93.40
15	3843, 9.60	6487, 19.00	50001, 185.90	50001, 186.20
16	3250, 8.20	4632, 13.30	50001, 74.60	50001, 104.80
17	50001, 126.80	50001, 158.60	50001, 186.40	50001, 186.40
18	50001, 127.30	50001, 134.60	50001, 186.70	50001, 186.80
19	3733, 9.30	4899, 13.40	50001, 58.80	50001, 85.90
20	3122, 7.90	3850, 9.90	50001, 186.70	50001, 186.70
Average	2103, 5.27	2751, 7.55	–	–

Table 4.b: Projection problems: variation of δ using $\eta = 0.01$, $dimchol = 0$ and conjugate gradients

Problem	$dimchol = 0$	$dimchol = 10$	$dimchol = 100$
1	47, 0.09	47, 0.09	8, 0.13
2	82, 0.17	82, 0.17	9, 0.22
3	132, 0.29	132, 0.29	10, 0.31
4	107, 0.24	107, 0.24	1, 0.09
5	140, 0.31	140, 0.31	11, 0.32
6	137, 0.31	137, 0.31	1, 0.09
7	77, 0.16	77, 0.16	9, 0.17
8	137, 0.28	137, 0.28	10, 0.26
9	61, 0.12	61, 0.12	8, 0.13
10	164, 0.34	164, 0.34	9, 0.22
11	5706, 14.40	5706, 14.40	12, 0.16
12	7931, 20.30	7931, 20.30	17, 0.26
13	4625, 11.50	4625, 11.50	24, 0.34
14	4556, 11.40	4556, 11.40	25, 0.38
15	3843, 9.60	3843, 9.60	22, 0.32
16	3250, 8.20	3250, 8.20	23, 0.35
17	50001, 126.80	50001, 126.80	53, 0.78
18	50001, 127.30	50001, 127.30	1, 0.04
19	3733, 9.30	3733, 9.30	18, 0.23
20	3122, 7.90	3122, 7.90	19, 0.26
Average	–	–	14.5, 0.25

Table 4.c: Projection problems: variation of *dimchol* using $\eta = 0.01$, $\delta = 0$ and conjugate gradients

6 Conclusions

Numerical experiments are, of course, quite dependent on the test problems used, so, the answers of the questions formulated at the beginning of the former section do not have an absolute value. However, we think that the experiments are conclusive enough, if not to take definite decisions, at least to guide the lines for future research. So, based on the numerical experiments actually performed, we can conclude that:

(a) With large values of η (namely, $\eta \geq 0.1$) we obtain, in general, better performances than with small ones. Therefore, the strategy of maintaining the active constraints whenever the "internal gradient" $g_I(x)$ dominates the projected gradient $g_P(x)$ is correct. On the other hand, once a "large value" for η has been adopted, it is not worthwhile to try "large" values for δ.

In fact, it seems that, in these cases, the large value of η is sufficient to guarantee that faces are not repeated frequently, independently of the choice δ. The poor performance of the methods for some large values of δ is due to difficulties of the linear iterative solvers of achieving high precisions in the unconstrained problems. We performed additional experiments (see also [14]) where we verified that the behavior of the sequence is not identical in the cases $\delta = 0$ and $\delta =$ (say)10^{-8}. Roughly speaking, the performance for $\eta = 0, \delta = 10^{-8}$ is very similar to the performance for $\eta = 0.9, \delta = 0$ and much better than the alternative $\eta = 0, \delta = 0$, in the obstacle problem.

The experiments and the considerations above could lead us to give up the criterion (3.1), leaving the responsibility of deciding to stay in the current face or not, exclusively to (2.9). However, the remark at the end of Section 3 recommends us to be cautious at this respect. There we saw that even very large values of η could not be large enough to guarantee identification of the correct face in a simple degenerate case. Table 3.a also sheds some light on the degenerate situation. The most degenerate problems on this set are $2, 4, 6, \ldots, 20$ and 21. On the other hand, $1, 3, 5, \ldots, 19$ and 22 are safely nondegenerate. We see that the difference between $\eta = 10^{-10}$ and $\eta = 0.9$ in the nondegenerate problems is negligible, but that difference is extremely impressive in the (almost) degenerate cases. In these problems, $\eta = 0.9$ was large enough to produce the same effect that (3.1) should produce. The example of Section 3 warns us about situations in which the situation could not be easily solved by a large η.

Anyway, the more efficient relation between η and δ should still be discovered, and this is an interesting subject for future research. The conflict between "identification" and "efficiency of the linear solver" opens the possibility of defining strategies where η, δ or both could be dependent on k.

(b) The best choice of *dimchol* is dramatically dependent of the structure of the problem. Iterative linear methods can produce much better or much worse results than the direct (Cholesky) method. In our experiments, we did not detect very large variations of the dimension of the faces visited by the algorithm. In such cases, it is recommendable to choose the direct or the iterative linear alternative on the same basis that we do when dealing with unconstrained problems of the same dimensions. For example, not very large problems with many different eigenvalues (as our projection problems) should be solved using direct methods. On the other hand, very large and highly structured problems favor the utilization of iterative linear methods.

(c) Gradient methods with retards were competitive with conjugate gradients. The Barzilai-Borwein method exhibited the best behavior among them. However, the potentiality of these new methods has not been fully exploited, especially in connection to their advantages in parallel computer environments. This will be one important subject of future research.

This paper was presented at the ICIAM Conference held in Hamburg in July 1995. At the same meeting, Dostál [11] presented an algorithm that, essentially, uses the chopped gradient and the strategy based on the criterion (2.9) for leaving the faces. He proved that, for strictly convex quadratic functions, identification holds under degeneracy if the parameter η is large enough. Both theoretical and practical results presented in [11] are complementary to the ones presented here.

References

[1] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, Ch. Romine and H. van der Vorst, *Templates for the solution of linear systems: building blocks for iterative methods*, SIAM Publications,

Philadelphia, 1994.

[2] J. Barzilai and J. M. Borwein, "Two point step size gradient methods", *IMA Journal of Numerical Analysis* 8, (1988) 141-148.

[3] D. P. Bertsekas, "Projected Newton methods for optimization problems with simple constraints", *SIAM Journal on Control and Optimization* 20, (1982) 141-148.

[4] J. V. Burke and J. J. Moré, "On the identification of active constraints", *SIAM Journal on Numerical Analysis* 25, (1988) 1197-1211.

[5] P. Ciarlet, *The finite element method for elliptic problems*, North Holland, Amsterdam, 1978.

[6] T. F. Coleman and L. A. Hulbert, "A direct active set algorithm for large sparse quadratic programs with simple bounds", *Mathematical Programming* 45, (1989) 373-406.

[7] T. F. Coleman and Y. Li, "An interior trust region approach for nonlinear minimization subject to bounds", *SIAM Journal on Optimization* 6, (1996) 418-445.

[8] A. R. Conn, N. I. M. Gould and Ph. L. Toint, "A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds", *SIAM Journal on Numerical Analysis* 28, (1988) 545-572.

[9] R. Dembo and U. Tulowitzki, "On the minimization of quadratic functions subject to box constraints", Working Paper B-71, School of Organization and Management, Yale University, New Haven, 1983.

[10] J. E. Dennis and L. N. Vicente, "Trust-region interior-point algorithms for minimization problems with simple bounds", in *Applied Mathematics and Parallel Computing (Fetschrift for Klaus Ritter)*, edited by H. Fischer, B. Riedmüller and S. Schäffer, Physica-Verlag, Springer-Verlag (1996) 97-107.

[11] Z. Dostál, "Box constrained quadratic programming with proportioning and projections", Technical Report, Department of Applied Mathematics, Technical University of Ostrava, Ostrava, Czech Republic, 1995.

[12] L. Fernandes, A. Fischer, J. J. Júdice, C. Requejo and C. Soares, "A block active set algorithm for large-scale quadratic programming with box constraints", Technical Report MATH-NM-14, Technical University of Dresden, 1995, to appear in *Annals of Operations Research.*

[13] A. Friedlander and J. M. Martínez, "On the numerical solution of bound constrained optimization problems", *RAIRO Operations Research* 23, (1989) 319-341.

[14] A. Friedlander and J. M. Martínez, "On the maximization of a concave quadratic function with box constraints", *SIAM Journal on Optimization* 4, (1994) 177-192.

[15] A. Friedlander, J. M. Martínez and M. Raydan, "A new method for large-scale box constrained quadratic minimization problems", *Optimization Methods and Software* 5, (1995) 57-74.

[16] A. Friedlander, J. M. Martínez and M. Raydan, "Gradient method with retards and generalizations". Submitted to *SIAM Journal on Numerical Analysis.*

[17] A. Friedlander, J. M. Martínez and S. A. Santos, "A new trust region algorithm for bound constrained minimization", *Applied Mathematics and Optimization* 30, (1994) 235-266.

[18] G. H. Golub and Ch. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore and London, 1989.

[19] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems", *Journal of Research of the National Bureau of Standards* B 49 (1952) 409-436.

[20] J. J. Júdice and F. M. Pires, "Direct methods for convex quadratic programming subject to box constraints", *Investigação Operacional* 9, (1989) 23-56.

[21] P. Lötstedt, "Solving the minimal least squares problem subject bounds on the variables", *BIT* 24, (1984) 206-224.

[22] J. J. Moré and G. Toraldo, "On the solution of large quadratic programming problems with bound constraints", *SIAM Journal on Optimization* 1, (1991) 93-113.

[23] R. H. Nickel and J. W. Tolle, "A sparse sequential programming algorithm", *JOTA* 60, (1989) 453-473.

[24] M. Raydan, "On the Barzilai and Borwein choice of steplength for the gradient method", *IMA Journal of Numerical Analysis* 13 (1993) 321-326.

[25] E. K. Yang and J. W. Tolle, "A class of methods for solving large, convex quadratic programs subject to box constraints", *Technical Report UNC/ORSA/TR-86-3*, Deptartment of Operations Research and Systems Analysis, University of North Carolina, Chapel Hill, NC., (1986).