Spectral projected gradient and variable metric methods for optimization with linear inequalities

Roberto Andreani * Ernesto G. Birgin † José Mario Martínez ‡
Jinyun Yuan §

July 8, 2004

Abstract

A family of variable metric methods for convex constrained optimization was introduced recently by Birgin, Martínez and Raydan. One of the members of this family is the Inexact Spectral Projected Gradient (ISPG) method for minimization with convex constraints. At each iteration of these methods a strictly convex quadratic function with convex constraints must be (inexactly) minimized. In the case of the ISPG method it was shown that, in some important applications, iterative projection methods can be used for this minimization. In this paper the particular case in which the convex domain is a polytope described by a finite set of linear inequalities is considered. For solving the linearly constrained convex quadratic subproblem a dual approach is adopted, by means of which subproblems become (not necessarily strictly) convex quadratic minimization problems with box constraints. These subproblems are solved by means of an active-set box-constraint quadratic optimizer with a proximal-point type unconstrained algorithm for minimization within the current faces. Convergence results and numerical experiments are presented.

Key words: Linearly-constrained optimization, projected gradient, nonmonotone line search, spectral gradient.

AMS Subject Classification: 49M07, 49M10, 65K, 90C06, 90C20.

*Departamento de Matemática Aplicada, IMECC-UNICAMP, CP 6065, 13081-970 Campinas SP, Brazil (andreani@ime.unicamp.br). Sponsored by FAPESP (Grant 01/04597-4), CNPq and FAEP-UNICAMP.

†Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, Rua do Matão 1010 Cidade Universitária, 05508-090 São Paulo, SP - Brazil (egbirgin@ime.usp.br). Sponsored by FAPESP (Grants 01/04597-4 and 02/00094-0), CNPq (Grant 300151/00-4) and Pronex.

‡Departamento de Matemática Aplicada, IMECC-UNICAMP, CP 6065, 13081-970 Campinas SP, Brazil (martinez@ime.unicamp.br). Sponsored by FAPESP (Grant 01/04597-4), CNPq and FAEP-UNICAMP.

§Departamento de Matemática, Universidade Federal do Paraná, Centro Politécnico, CP: 19.081, 81531-990, Curitiba, PR, Brazil (jin@mat.ufpr.br). Sponsored partially by CNPq and Fundação Araucária.

1 Introduction

We consider the problem

$$\text{Minimize } f(x) \text{ subject to } x \in \Omega, \tag{1}$$

where

$$\Omega = \{x \in I\!R^n \mid Ax \leq b\}. \tag{2}$$

The matrix A has m rows and n columns and f admits continuous first derivatives on an open set that contains Ω. We will assume that the level sets $\{x \in \Omega \mid f(x) \leq c\}$ are bounded.

This is a classical problem in practical continuous optimization. See, for example, [14, 22, 28, 33]. The best known general-purpose algorithm for solving this problem is implemented in the software MINOS [34].

The Spectral Projected Gradient (SPG) method for solving (1) when Ω is a general convex set was introduced in [7]. In [8] it was shown that, when the projections onto the convex set Ω are not difficult to compute, the performance of the SPG method for solving very large problems can be quite impressive.

The SPG method is a variation of the projected gradient method (see [4, 29, 32]). The features of the SPG method that make it computationally efficient are the spectral (Barzilai-Borwein) choice of the steplength, introduced in [2] and analyzed by several authors (see, among others, [15, 16, 23, 24, 36, 37]) and the use of nonmonotone line-search strategies [30]. The difficulty in computing projections in critical cases motivated the introduction of the Inexact Spectral Projected Gradient (ISPG) method in [9]. The idea is that the search direction does not need to be computed exactly for obtaining global convergence and efficiency. So, iterative methods with premature stopping criteria can be used. In [9] the ISPG method was also generalized giving rise to a general Inexact Variable Metric method (IVM) for convex constrained optimization. In [9] the authors implemented the ISPG method using a projection method [11, 20, 31, 38] for approximate computation of projections on a family of matricial optimization problems with linear constraints.

In this paper we introduce a general purpose IVM algorithm for solving (1). Subproblems are solved using the dual of a strictly convex quadratic programming problem. This problem can be reduced to the minimization of a convex quadratic function subject only to nonnegativity constraints. For solving the quadratic programming problem we use a well established active-set algorithm [25, 26] and, for minimization within current faces, a damped proximal-point unconstrained minimization method is employed.

This paper is organized as follows. The inexact variable metric method proposed in [9] is briefly reviewed in Section 2. Local convergence of Newton-like instances is discussed in Section 3. The dual of quadratic programming with linear inequality constraints is derived in Section 4. In Section 5 it is explained how to obtain an admissible trial point starting from an approximate solution of the dual problem mentioned in Section 4. In Section 6 the algorithms used to solve the non-negatively constrained dual problem are justified. In Section 7 we describe the damped proximal-point algorithm used for internal minimization within faces. Numerical experiments are presented in Section 8. In Section 9 we state some conclusions and lines for future research.

Notation.

$\|\cdot\|$ is the 2-norm of vectors and matrices;

$\mathcal{B}(z,\delta) = \{x \in I\!R^n \mid \|x - z\| < \delta\}$;

$\bar{\mathcal{B}}(z,\delta) = \{x \in I\!R^n \mid \|x - z\| \leq \delta\}$;

$I\!N = \{0, 1, 2, \ldots\}$;

$I\!R^n_+ = \{x \in I\!R^n \mid x \geq 0\}$;

$I\!R^n_{++} = \{x \in I\!R^n \mid x > 0\}$.

The symbol $[v]_i$ denotes the i-th component of the vector v in the cases in which the usual notation v_i can lead to confusion.

We denote $g(x) = \nabla f(x)$. Consequently, $g_k = \nabla f(x_k)$.

$Int(\Omega)$ denotes the set of interior points of Ω.

$\mathcal{N}(A)$ denotes the null-space of the matrix A.

$\mathcal{R}(Z)$ denotes the column-space of the matrix Z.

2 Inexact Variable Metric method

In this section the Inexact Variable Metric (IVM) method is reviewed and the global convergence results proved in [9] are recalled.

Let $L \in I\!R_{++}$. We denote by $\mathcal{G}$ the set of $n \times n$ symmetric positive definite matrices such that, for all $G \in \mathcal{G}$,

$$\|G\| \leq L \text{ and } \|G^{-1}\| \leq L.$$

The Inexact Variable Metric method introduced in [9] is as follows.

Algorithm 2.1: Inexact Variable Metric Method

Assume $\eta \in (0,1)$, $\gamma \in (0, \frac{1}{2})$, $0 < \sigma_1 < \sigma_2 < 1$, M a positive integer. Let $x_0 \in \Omega$ be an arbitrary initial point. Given $x_k \in \Omega$, $G_k \in \mathcal{G}$, the steps of the $k-$th iteration of the algorithm are:

Step 1. *Compute the search direction*

Consider the subproblem

$$\text{Minimize } Q_k(d) \quad \text{subject to} \quad A(x_k + d) \leq b, \tag{3}$$

where

$$Q_k(d) = \frac{1}{2} d^T G_k d + g_k^T d \quad \forall \; d \in I\!R^n.$$

Compute d_k such that $x_k + d_k \in \Omega$ and

$$Q_k(d_k) \le \eta \; Q_k(\bar{d}_k), \tag{4}$$

where $\bar{d}_k$ is the exact solution of (3).

If $d_k = 0$, stop the execution of the algorithm declaring that x_k is a stationary point.

Step 2. *Compute the steplength*

Set $\alpha \leftarrow 1$ and $f_{\max} = \max\{f(x_{k-j+1}) \mid 1 \le j \le \min\{k+1, M\}\}$.
If

$$f(x_k + \alpha d_k) \le f_{\max} + \gamma \alpha g_k^T d_k,$$

set $\alpha_k = \alpha$, $x_{k+1} = x_k + \alpha_k d_k$ and finish the iteration. Otherwise, choose

$$\alpha_{new} \in [\sigma_1 \alpha, \sigma_2 \alpha], \tag{5}$$

set $\alpha \leftarrow \alpha_{new}$ and repeat If.

Remark. Since G_k is positive definite, the subproblem (3) has a unique solution $\bar{d}_k$. This solution is mentioned in the test (4) but does not need to be computed at all. Instead, we will see how to generate a sequence of lower bounds $C_\ell \le Q_k(\bar{d}_k)$ that converges to the true value $Q_k(\bar{d}_k)$. Clearly, if the computable criterion $Q_k(d_k) \le \eta C_\ell$ is satisfied, the test (4) is satisfied as well. The lower bounds C_ℓ will be the values of the Lagrangian associated with the dual problem of (3).

Lemma 2.1 and Theorem 2.1 were proved in [9].

Lemma 2.1. *Algorithm 2.1 is well defined.*

Theorem 2.1. *Assume that the set $\{x \in \Omega \mid f(x) \le f(x_0)\}$ is bounded. Then, either the sequence generated by Algorithm 2.1 stops at some stationary point x_k, or every limit point of the generated sequence is stationary.*

3 Local convergence

In this section we prove that, under some conditions, the whole sequence generated by the IVM method converges to a stationary point of the original problem. Moreover, when the matrices G_k are chosen as Hessians or secant approximations of the Hessians (see [18]) and the subproblems are solved exactly, the convergence is quadratic or superlinear respectively.

Lemma 3.1 characterizes the exact solutions of the quadratic subproblems.

Lemma 3.1. *For all $k \in I\!N$ there exists $\lambda_k \in I\!R^m_+$ such that*

$$G_k \bar{d}_k + g_k + A^T \lambda_k = 0, \tag{6}$$

$$A(x_k + \bar{d}_k) \le b, \tag{7}$$

$$\lambda_k^T [A(x_k + \bar{d}_k) - b] = 0. \tag{8}$$

Proof. Recall that $\bar{d}_k$ is the exact solution of (3). The equations (6)–(8) are the KKT conditions of this subproblem. □

Assumptions L1–L7 will be used for proving different results along this section. None of these assumptions is going to be used in forthcoming sections.

Assumption L1 *(Isolation)*

There exists $x_* \in \Omega$, a limit point of the sequence generated by Algorithm 2.1, and $\delta > 0$ such that x_* is the unique stationary point of (1) in the ball $\bar{B}(x_*, \delta)$.

Theorem 3.1 *Assume that $x_* \in \Omega$ is a limit point of sequence generated by Algorithm 2.1 that x_* satisfies Assumption L1. Then, the sequence converges to x_*.*

Proof. Let $\delta > 0$ be such that x_* is the unique stationary point of problem (1) in $\bar{\mathcal{B}}(x_*, \delta)$.

Define

$$K = \{k \in I\!N \ \mid \ \|x_k - x_*\| \le \frac{\delta}{2}\}. \tag{9}$$

Clearly

$$\lim_{k \in K} x_k = x_*, \tag{10}$$

otherwise we would have a set of the form $\{x \in \Omega \mid \varepsilon \le \|x - x_*\| \le \frac{\delta}{2}\}$ containing infinitely many elements of $\{x_k\}_{k \in I\!N}$, which is impossible because x_* is an isolated limit point.

By Lemma 2.10 of [9] we have that

$$\lim_{k \in K} \|x_{k+1} - x_k\| = 0. \tag{11}$$

Consider the set

$$C_0 = \{x \in \Omega \mid \frac{\delta}{2} \le \|x - x_*\| \le \delta\}.$$

Since x_* is isolated, it follows that C_0 does not contain any stationary point of (1). Therefore, by Theorem 2.1, C_0 cannot contain infinitely many members of the sequence $\{x_k\}_{k \in I\!N}$. This implies that there exists $k_1 \in I\!N$ such that

$$x_k \notin C_0 \ \ \forall \ \ k \ge k_1. \tag{12}$$

By (10) and (11) there exists $k_2 \in K$, $k_2 \ge k_1$, such that

$$\|x_{k_2} - x_*\| < \frac{\delta}{2}. \tag{13}$$

and

$$\|x_{k+1} - x_k\| \leq \frac{\delta}{2} \, \forall \, k \in K, k \geq k_2. \tag{14}$$

By (12), (13) and (14), we obtain that

$$\|x_{k_2+1} - x_*\| \leq \frac{\delta}{2}.$$

Therefore, $k_2 + 1 \in K$. Continuing this argument inductively, we obtain that

$$\|x_k - x_*\| \leq \frac{\delta}{2} \;\; \forall \, k \geq k_2.$$

By (9) and (10) this implies that

$$\lim_{k \to \infty} x_k = x_*,$$

as we wanted to prove. □

Since the constraints of (1) are linear, the KKT conditions are satisfied by local minimizers. We assume, without loss of generality that the active constraints at x_* are those corresponding to the first s rows of A. Let us denote

$$A = (a_1, \ldots, a_m)^T, \;\; \bar{A} = (a_1, \ldots, a_s)^T, \;\; \bar{b} = (b_1, \ldots, b_s)^T.$$

By the KKT conditions, there exists $\bar{\lambda}_* \in I\!R^s_+$ such that

$$\nabla f(x_*) + \bar{A}^T \bar{\lambda}_* = 0,$$

$$\bar{A} x_* = \bar{b},$$

$$a_i^T x_* < b_i, \; i = s+1, \ldots, m.$$

Assumption L2 *(Strict complementarity)*

$$[\bar{\lambda}_*]_i > 0 \;\; \forall \;\; i = 1, \ldots, s.$$

Assumption L3 *(Regularity)*

The vectors $a_1, \ldots, a_s$ are linearly independent. (In particular, $s \leq n$.)

Assumption L4 *(Taylor)*

The function f has continuous second derivatives and its Hessian satisfies a Lipschitz condition. Therefore, for all $x, y \in \Omega$,

$$f(y) = f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(x)(y - x) + o(\|y - x\|^2).$$

Assumption L5 *(Dennis-Moré condition [17, 18])*

If $Z \in I\!R^{n\times(n-s)}$ is such that $\mathcal{R}(Z) = \mathcal{N}(\bar{A})$, then:

$$\lim_{k\to\infty} \frac{\|Z^T[G_k - \nabla^2 f(x_k)]\bar{d}_k\|}{\|\bar{d}_k\|} = 0.$$

Assumption L6 *(Nonsingularity)*

If $n > s$ and $Z \in I\!R^{n\times(n-s)}$ is such that $\mathcal{R}(Z) = \mathcal{N}(\bar{A})$, then the reduced Hessian $Z^T\nabla^2 f(x_*)Z$ is positive definite.

Assumption L7

The subproblem (3) is solved exactly. (So, $d_k = \bar{d}_k$ for all $k \in I\!N$.)

Assumption L8

$G_k = \nabla^2 f(x_k) \in \mathcal{G}$ *for all* $k \in I\!N$.

Lemma 3.2. *Suppose that Assumptions L1, L2 and L3 are satisfied by the sequence $\{x_k\}$ generated by Algorithm 2.1. Then,*

$$\lim_{k\to\infty} \bar{d}_k = 0. \tag{15}$$

Moreover:

1. *There exist $k_0 \in I\!N$ and $\{\bar{\lambda}_k\}_{k\geq k_0} \subset I\!R^s_{++}$ such that, for all $k \geq k_0$,*

$$G_k\bar{d}_k + g_k + \bar{A}^T\bar{\lambda}_k = 0, \tag{16}$$

$$\bar{A}(x_k + \bar{d}_k) = \bar{b} \tag{17}$$

and

$$a_i^T(x_k + \bar{d}_k) < b_i \ \ \forall\, i = s+1,\ldots,m. \tag{18}$$

2.

$$\lim_{k\to\infty} [\bar{\lambda}_k]_i = [\bar{\lambda}_*]_i \ \ \forall\, i = 1,\ldots,s.$$

3. *If, in addition, Assumption L7 holds and, for some $\bar{k} \geq k_0$ we have that*

$$\bar{A}x_{\bar{k}} = \bar{b},$$

then

$$\bar{A}x_k = \bar{b} \tag{19}$$

and

$$\bar{d}_k = -Z(Z^TG_kZ)^{-1}Z^Tg_k \tag{20}$$

for all $k \geq \bar{k}$, where the columns of $Z \in I\!Rn \times (n-s)$ form a basis of $\mathcal{N}(A)$.

Proof. By Assumption L1 and Theorem 3.1 the sequence x_k converges to x_*. By Lemma 2.12 of [9],

$$\lim_{k\to\infty} Q_k(\bar{d}_k) = 0.$$

Let us prove first that (15) holds. We proceed by contradiction. Suppose that there exists an infinite sequence of indices K_1 such that $\bar{d}_k$ is bounded away from zero. By Lemma 2.6 of [9], $\bar{d}_k$ is bounded, therefore there exists an infinite sequence of indices K_2 such that

$$\lim_{k\in K_2} \bar{d}_k = \bar{d} \neq 0. \tag{21}$$

Let $K_3 \subset K_2$ be a subsequence of indices such that

$$\lim_{k\in K_3} G_k = G \in \mathcal{G}.$$

By continuity, it is easy to see that $\bar{d}$ is a global minimizer of $\frac{1}{2}d^T G d + \nabla f(x_*)^T d$ subject to $x_* + d \in \Omega$. But, since x_* is stationary, $d = 0$ is a global minimizer of the quadratic $\frac{1}{2}d^T G d + \nabla f(x_*)^T d$. Since $G \in \mathcal{G}$ this quadratic has only one global minimizer, so $\bar{d} = 0$, which contradicts (21). So, (15) is proved.

By (15), since $a_i^T x_* < b_i$ for all $i = s+1, \ldots, m$, we have that (18) holds for k large enough. Therefore, by (8) and (15),

$$[\lambda_k]_i = 0 \; \forall \; i = s+1, \ldots, m.$$

By (6), defining $\bar{\lambda}_k = ([\lambda_k]_1, \ldots, [\lambda_k]_s)^T$, the equation (16) also holds.

Let us prove that (17) holds for k large enough. Observe first that, by L3,

$$\bar{\lambda}_* = -(\bar{A}\bar{A}^T)^{-1} g(x_*). \tag{22}$$

By (16),

$$\bar{A} G_k \bar{d}_k + \bar{A} g_k + \bar{A}\bar{A}^T \bar{\lambda}_k = 0.$$

Moreover, by L3 and the nonsingularity of $\bar{A}\bar{A}^T$:

$$(\bar{A}\bar{A}^T)^{-1}\bar{A} G_k \bar{d}_k + (\bar{A}\bar{A}^T)^{-1}\bar{A} g_k + \bar{\lambda}_k = 0.$$

Taking limits and using that $\bar{d}_k \to 0$ we obtain that the sequence $\bar{\lambda}_k$ is convergent and

$$\lim_{k\to\infty} \bar{\lambda}_k = -(\bar{A}\bar{A}^T)^{-1}\bar{A} g(x_*).$$

By (22), this implies that

$$\lim_{k\to\infty} \bar{\lambda}_k = \bar{\lambda}_*.$$

By L2, $\bar{\lambda}_* > 0$. Then, $\bar{\lambda}_k > 0$ for all k large enough. By (8) the equalities (17) also hold.

Finally, assume that for some $\bar{k} \geq k_0$, $\bar{A} x_{\bar{k}} = \bar{b}$. By (18) we have that $\bar{d}_{\bar{k}} \in \mathcal{N}(\bar{A})$. So, by Assumption L7 and the definition of $x_{\bar{k}+1}$, we obtain that $\bar{A} x_{\bar{k}+1} = \bar{b}$. Proceeding by induction, we deduce that $\bar{A} x_k = \bar{b}$ for all $k \geq \bar{k}$. Therefore, (20) follows from (16) and (17). $\square$

Lemma 3.3. *Suppose that Assumptions L1, L2, L3 and L7 are satisfied by an infinite sequence $\{x_k\}$ generated by Algorithm 2.1. Then, every limit point of the sequence $\{\bar{d}_k/\|\bar{d}_k\|\}$ belongs to $\mathcal{N}(\bar{A})$.*

Moreover, for all $k \in I\!N$,

$$\frac{\bar{d}_k}{\|\bar{d}_k\|} = \tilde{d}_k^{(1)} + \tilde{d}_k^{(2)},$$

where $\tilde{d}_k^{(1)} \in \mathcal{N}(\bar{A})$, $\tilde{d}_k^{(2)} \in \mathcal{R}(\bar{A}^T)$ and

$$\lim_{k\to\infty} \tilde{d}_k^{(2)} = 0.$$

Proof. By (16),

$$\bar{d}_k^T G_k \bar{d}_k + g_k^T \bar{d}_k + \lambda_k^T \bar{A}\bar{d}_k = 0 \ \ \forall\, k \geq k_0. \tag{23}$$

Since f has continuous first derivatives, we have that

$$f(x_k + \bar{d}_k) = f(x_k) + \gamma g_k^T \bar{d}_k + (1-\gamma) g_k^T \bar{d}_k + o(\|\bar{d}_k\|).$$

So, by (23), and the positive definiteness of G_k,

$$\begin{aligned} f(x_k + \bar{d}_k) - f(x_k) - \gamma g_k^T \bar{d}_k &= -(1-\gamma)\bar{d}_k^T G_k \bar{d}_k - (1-\gamma)\lambda_k^T \bar{A}\bar{d}_k + o(\|\bar{d}_k\|) \\ &\leq (\gamma - 1)\lambda_k^T \bar{A}\bar{d}_k + o(\|\bar{d}_k\|) \end{aligned}$$

for all $k \geq k_0$. So,

$$\frac{f(x_k + \bar{d}_k) - f(x_k) - \gamma g_k^T \bar{d}_k}{\|\bar{d}_k\|} \leq (\gamma - 1)\lambda_k^T \bar{A}\frac{\bar{d}_k}{\|\bar{d}_k\|} + o(1) \tag{24}$$

for all $k \geq k_0$.

By Lemma 3.2, $\bar{A}(x_k + \bar{d}_k) = \bar{b}$ for all $k \geq k_0$. So, by the feasibility of x_k,

$$-\bar{A}\bar{d}_k = \bar{A}x_k - \bar{b} \leq 0 \ \ \forall\, k \geq k_0. \tag{25}$$

Therefore,

$$\bar{A}\frac{\bar{d}_k}{\|\bar{d}_k\|} \geq 0 \ \ \forall\, k \geq k_0. \tag{26}$$

Assume that $K \subset I\!N$ is an infinite set of indices such that

$$\lim_{k\in K} \frac{\bar{d}_k}{\|\bar{d}_k\|} = \tilde{d}. \tag{27}$$

Suppose, by contradiction, that

$$\bar{A}\tilde{d} \neq 0. \tag{28}$$

Taking limits in (26) we see that $\bar{A}\tilde{d} \geq 0$. Therefore, there exists $j \in \{1, \ldots, s\}$ such that $[\bar{A}\tilde{d}]_j > 0$. Then, by Assumption L2,

$$\lambda_*^T \bar{A}\tilde{d} = \rho > 0.$$

By Lemma 3.2, since $\gamma < 1$, this implies that there exists an infinite set $K_1 \subset \{k \in K \mid k \geq k_0\}$ such that

$$(\gamma - 1)\lambda_k^T \bar{A} \frac{\bar{d}_k}{\|\bar{d}_k\|} < -\rho/2 < 0$$

for all $k \in K_1$. Therefore, by (24), if $k \in K_1$ is large enough,

$$\frac{f(x_k + \bar{d}_k) - f(x_k) - \gamma g_k^T \bar{d}_k}{\|\bar{d}_k\|} < -\rho/4 < 0.$$

So, by Assumption L7, if $k \in K_1$ is large enough, $x_{k+1} = x_k + \bar{d}_k$. Thus, by Lemma 3.2, $\bar{A}x_{k+1} = \bar{b}$ for all $k \in K_1$ large enough. Again by Lemma 3.2, this implies that $\bar{A}x_k = \bar{b}$ and $\bar{A}\bar{d}_k = 0$ for all k large enough. This contradicts (28). So, the first part of the thesis is proved.

For proving the second part, let us write

$$\frac{\bar{d}_k}{\|\bar{d}_k\|} = \tilde{d}_k^{(1)} + \tilde{d}_k^{(2)},$$

where $\tilde{d}_k^{(1)} \in \mathcal{N}(\bar{A})$, $\tilde{d}_k^{(2)} \in \mathcal{R}(\bar{A}^T)$. By Pythagoras,

$$\|\tilde{d}_k^{(1)}\|^2 + \|\tilde{d}_k^{(2)}\|^2 = 1,$$

so $\{\tilde{d}_k^{(1)}\}$ and $\{\tilde{d}_k^{(2)}\}$ are bounded. If $\|\tilde{d}_k^{(2)}\|$ does not tend to zero then, some subsequence of $\tilde{d}_k^{(2)}$ converges to some nonnull vector

$$\tilde{d}^{(2)} \in \mathcal{R}(\bar{A}^T). \tag{29}$$

Taking an appropriate subsequence, $\tilde{d}_k^{(1)}$ converges to $\tilde{d}^{(1)} \in \mathcal{N}(\bar{A})$. So, for some subsequence, $\frac{\bar{d}_k}{\|\bar{d}_k\|}$ converges to $\tilde{d}^{(1)} + \tilde{d}^{(2)}$. By (29), this vector does not belong to $\mathcal{N}(\bar{A})$. This contradicts the first part of the thesis. □

Corollary 3.1. *If the Assumptions L1, L2, L3 and L7 are satisfied and $n = s$ the sequence terminates with $x_k = x_*$ for some finite k.*

Proof. Suppose that the sequence $\{x_k\}$ is infinite. Then, by Lemma 3.3, any limit point of $\{d_k/\|d_k\|\}$ must belong to $\mathcal{N}(\bar{A})$. But, when $n = s$ we have that $\mathcal{N}(\bar{A}) = \{0\}$. This is a contradiction, since all the terms of $\{d_k/\|d_k\|\}$ are unitary. Therefore, the sequence $\{x_k\}$ must be finite and, so, it terminates at x_*. □

Lemma 3.4. *Suppose that Assumptions L1– L7 are satisfied by the sequence $\{x_k\}$ generated by Algorithm 2.1. Then, there exists $k_1 \in I\!N$ such that, for all $k \geq k_1$*

$$f(x_k + \bar{d}_k) \leq f(x_k) + \gamma g_k^T \bar{d}_k.$$

Proof. By Assumption L4, for all $k \in I\!N$ we have:

$$f(x_k + \bar{d}_k) = f(x_k) + g_k^T \bar{d}_k + \frac{1}{2} \bar{d}_k^T \nabla^2 f(x_k) \bar{d}_k + o(\|\bar{d}_k\|^2)$$

$$= f(x_k) + \gamma g_k^T \bar{d}_k + (1-\gamma) g_k^T \bar{d}_k^T + \frac{1}{2} \bar{d}_k^T \nabla^2 f(x_k) \bar{d}_k + o(\|\bar{d}_k\|^2)$$

$$= f(x_k) + \gamma g_k^T \bar{d}_k + (1-\gamma) g_k^T \bar{d}_k + (1-\gamma) \bar{d}_k^T \nabla^2 f(x_k) \bar{d}_k + (\gamma - \frac{1}{2}) \bar{d}_k^T \nabla^2 f(x_k) \bar{d}_k + o(\|\bar{d}_k\|^2)$$

$$= f(x_k) + \gamma g_k^T \bar{d}_k + (1-\gamma)[g_k^T \bar{d}_k + \bar{d}_k^T G_k \bar{d}_k] + (1-\gamma) \bar{d}_k^T [\nabla^2 f(x_k) - G_k] \bar{d}_k$$
$$+ (\gamma - \frac{1}{2}) \bar{d}_k^T \nabla^2 f(x_k) \bar{d}_k + o(\|\bar{d}_k\|^2).$$

Therefore,

$$\frac{f(x_k + \bar{d}_k) - f(x_k) - \gamma g_k^T \bar{d}_k}{\|\bar{d}_k\|^2} = (1-\gamma) \frac{[g_k + G_k \bar{d}_k]^T \bar{d}_k}{\|\bar{d}_k\|^2}$$
$$+ (1-\gamma) \frac{\bar{d}_k^T [\nabla^2 f(x_k) - G_k] \bar{d}_k}{\|\bar{d}_k\|^2} + (\gamma - \frac{1}{2}) \frac{\bar{d}_k^T \nabla^2 f(x_k) \bar{d}_k}{\|\bar{d}_k\|^2} + \frac{o(\|\bar{d}_k\|^2)}{\|\bar{d}_k\|^2}. \quad (30)$$

Now, $\bar{d}_k$ is the exact solution of (3) and $\nabla Q_k(\bar{d}_k) = G_k \bar{d}_k + g_k$. By the convexity of Q_k, the function $Q_k(d)$ decreases monotonically along the segment $\{t\bar{d}_k,\ t \in [0,1]\}$. This implies that, for all $k \in I\!N$,

$$(g_k + G_k \bar{d}_k)^T \bar{d}_k = \nabla Q_k(\bar{d}_k)^T \bar{d}_k \leq 0. \quad (31)$$

Let $\tilde{d}_k^{(1)}$ and $\tilde{d}_k^{(2)}$ be as in the thesis of Lemma 3.3. Then,

$$\frac{\bar{d}_k^T [\nabla^2 f(x_k) - G_k] \bar{d}_k}{\|\bar{d}_k\|^2} = \frac{\bar{d}_k^T}{\|\bar{d}_k\|} [\nabla^2 f(x_k) - G_k] \frac{\bar{d}_k}{\|\bar{d}_k\|}$$

$$= (\tilde{d}_k^{(1)} + \tilde{d}_k^{(2)})^T [\nabla^2 f(x_k) - G_k] \frac{\bar{d}_k}{\|\bar{d}_k\|}$$

$$= (\tilde{d}_k^{(1)})^T [\nabla^2 f(x_k) - G_k] \frac{\bar{d}_k}{\|\bar{d}_k\|} + (\tilde{d}_k^{(2)})^T [\nabla^2 f(x_k) - G_k] \frac{\bar{d}_k}{\|\bar{d}_k\|}. \quad (32)$$

The first term of (32) tends to zero by the Dennis-Moré condition and the second term tends to zero since, by Lemma 3.3, $\tilde{d}_k^{(2)}$ tends to zero. Thus,

$$\lim_{k \to \infty} \frac{\bar{d}_k^T [\nabla^2 f(x_k) - G_k] \bar{d}_k}{\|\bar{d}_k\|^2} = 0. \quad (33)$$

By Lemma 3.3 and Assumption L6, there exist $k_2 \in I\!N$ and $\varrho > 0$ such that

$$\frac{\bar{d}_k^T \nabla^2 f(x_k) \bar{d}_k}{\|\bar{d}_k\|^2} > \varrho \tag{34}$$

for all $k \geq k_2$.

Since $\gamma < \frac{1}{2}$, using (30), (31), (33) and (34) we obtain, for $k \geq k_2$,

$$\frac{f(x_k + \bar{d}_k) - f(x_k) - \gamma g_k^T \bar{d}_k}{\|\bar{d}_k\|^2} \leq (\gamma - \frac{1}{2})\varrho + o(1)$$

So, taking $k_1 \geq k_2$ large enough and $k \geq k_1$,

$$\frac{f(x_k + \bar{d}_k) - f(x_k) - \gamma g_k^T \bar{d}_k}{\|\bar{d}_k\|^2} \leq 0.$$

This implies the desired result. □

Theorem 3.2. *Suppose that Assumptions L1–L7 hold and that $\{x_k\}$ is a sequence generated by Algorithm 2.1. Then, there exists $k_3 \in I\!N$ such that for all $k \geq k_3$, we have that $\alpha_k = 1$ and $\bar{A}x_k = \bar{b}$. Moreover, $\{x_k\}$ converges to x_* superlinearly.*

Proof. Let Z be a matrix whose columns are a basis of the null-space of $\bar{A}$. Define

$$v_k = -(Z^T G_k Z)^{-1} g_k. \tag{35}$$

By (20) and Lemma 3.4, we have that $\bar{d}_k = Z v_k$ for k large enough.

Then, by Lemma 3.4, there exists $k_3 \in I\!N$ such that for all $k \geq k_3$, $\bar{A}x_k = \bar{b}$ and

$$x_{k+1} = x_k + \bar{d}_k = x_k + Z v_k. \tag{36}$$

Then, for all $k > k_3$,

$$x_k = x_{k_3} + Z u_k, \tag{37}$$

where

$$u_k = v_{k_3} + \ldots + v_{k-1}. \tag{38}$$

Consider the nonlinear system of equations

$$\Phi(u) \equiv Z^T \nabla f(x_{k_3} + Zu) = 0. \tag{39}$$

Define $u_* \in I\!R^{n-s}$ by

$$x_* = x_{k_3} + Z u_*.$$

Clearly, $\Phi(u_*) = 0$ and, by Assumption L6, $\Phi'(u_*)$ is nonsingular.

By (36)–(39) the sequence u_k is defined, for k large enough, by

$$u_{k_3} = 0, \; u_{k+1} = u_k + v_k \; \forall \; k \geq k_3. \tag{40}$$

But, by (35), for all $k \geq k_3$,

$$v_k = -B_k^{-1}\Phi(u_k) \tag{41}$$

where

$$B_k = Z^T G_k Z.$$

Then, by Assumption L5 and the linear independence of the columns of Z,

$$\lim_{k\to\infty} \frac{\|[B_k - \Phi'(u_*)](v_k)\|}{\|v_k\|} = 0. \tag{42}$$

By (40) and (41) $\{u_k\}$ is a quasi-Newton sequence associated to the nonlinear system (39). By (42) the Dennis-Moré condition holds. Therefore (see [17, 18]) the convergence of u_k is superlinear. By the definition of v_k and the linear independence of the columns of Z this implies the superlinear convergence of $\{x_k\}$. □

Theorem 3.3. *Suppose that Assumptions L1–L6 and L8 hold and that $\{x_k\}$ is a sequence generated by Algorithm 2.1. Then, there exists $k_3 \in I\!N$ such that, for all $k \geq k_3$, we have that $\alpha_k = 1$ and $\bar{A}x_k = \bar{b}$. Moreover $\{x_k\}$ converges quadratically to x_*.*

Proof. Clearly, Assumption L8 implies Assumption L7. Then, the arguments for this proof are the same as the ones of Theorem 3.2, invoking the local convergence theory of Newton's method [18], instead of the one of quasi-Newton algorithms. □

4 The Dual of Quadratic Programming

Writing $G = G_k, u = d, c = g_k, \bar{b} = b - Ax_k$, problem (3) is equivalent to

$$\begin{cases} \text{Minimize } \frac{1}{2}u^T G u + c^T u \\ \text{s.t. } Au \leq \bar{b} \end{cases} \tag{43}$$

where $A \in R^{m\times n}$ and G is symmetric and positive definite.

Define the Lagrangian function:

$$L(u, y) = \frac{1}{2}u^T G u + c^T u + y^T(Au - \bar{b}).$$

The dual problem of (43) (see [22], Section 9.5) is

$$\begin{cases} \text{Maximize}_{u,y} \;\; L(u, y) \\ \text{s.t. } \; Gu + c + A^T y = 0 \\ \quad y \geq 0 \end{cases} \tag{44}$$

It follows from the first constraint of (44) that

$$u = -G^{-1}(c + A^T y). \tag{45}$$

So, if (u, y) satisfies the first constraint of (44),

$$L(u, y) = -\frac{1}{2}(c + A^T y)^T G^{-1}(c + A^T y) - \bar{b}^T y.$$

Therefore, (44) is equivalent to

$$\begin{cases} \text{Minimize } \frac{1}{2}(c + A^T y)^T G^{-1}(c + A^T y) + \bar{b}^T y \\ \text{s.t.} \quad y \geq 0. \end{cases} \tag{46}$$

The main properties of the dual approach are:

(a) If $\bar{u}$ is a feasible point of (43) and (u, y) is a feasible point of (44), then

$$L(u, y) \leq \frac{1}{2}\bar{u}^T G\bar{u} + c^T \bar{u}.$$

(b) If $y \in I\!R^m_+$, $u = -G^{-1}(c + A^T y)$ and u is a feasible point of (43) (so, $Au \leq \bar{b}$) then u is the unique solution of (43).

5 Obtaining a trial point for the IVM iteration

Making the proper substitutions, and eliminating the constant term in the objective function, problem (46) has the form

$$\text{Minimize } \frac{1}{2}y^T By + \hat{b}^T y \quad \text{s.t.} \quad y \geq 0, \tag{47}$$

where $B = AG^{-1}A^T$ and $\hat{b} = \bar{b} + AG^{-1}c$.

Assume, for a moment, that we have an algorithm for non-negative constrained convex quadratic minimization that is able to generate a (finite or infinite) sequence converging to a solution of (47). For each iteration y_ℓ of this box-constrained quadratic solver we compute, using (45),

$$u_\ell = -G^{-1}(c + A^T y_\ell).$$

Then, either u_ℓ is feasible for some finite ℓ, therefore being the solution of (43), or the sequence $\{u_\ell\}$ converges to a solution of (43). Therefore, for all ℓ each u_ℓ we have a lower bound $C_\ell = L(u_\ell, y_\ell)$ that converges to the optimal functional value of (43). Recall the remark after the definition of Algorithm 2.1.

These properties suggest a practical procedure for obtaining a point that satisfies (4). For implementing Algorithm 5.1 below we assume that

$$Int(\Omega) \neq \emptyset.$$

Algorithm 5.1 *Computation of the search direction d_k.*

Assume that the iterate x_k of the IVM method (Algorithm 2.1) is *interior*, that is, $Ax_k < b$. Choose $\beta \in (0, 1), \eta' \in (0, 1)$ and define $\eta = \beta\eta'$. Set $\ell \leftarrow 0$.

Step 1. *Check exact solution.*

Compute y_ℓ, the $\ell-$th iterate of the box-constrained quadratic solver mentioned above. Compute

$$u_\ell = -G^{-1}(c + A^T y_\ell).$$

If $Au_\ell \leq \bar{b}$ go to Step 3. (In this case, y_ℓ is a solution of (47) and u_ℓ is the exact solution of (43).)

Step 2. *Check inexact solution.*

Compute

$$\alpha_{\max}^\ell = \max\{\alpha \geq 0 \mid x_k + \alpha u_\ell \in \Omega\}.$$

(In this case $\alpha_{\max}^\ell < 1$ since u_ℓ is infeasible.)

If

$$Q_k(\beta \alpha_{\max}^\ell u_\ell) \leq \eta L(u_\ell, y_\ell) \tag{48}$$

set

$$d_k = \beta \alpha_{\max}^\ell u_\ell$$

and terminate the execution of the algorithm.

If (48) does not hold, set $\ell \leftarrow \ell + 1$ and go to Step 1.

Step 3. *Keep the new trial point interior enough*

Compute

$$\alpha_{\max}^\ell = \max\{\alpha \geq 0 \mid x_k + \alpha u_\ell \in \Omega\}.$$

Define

$$d_k = \min\{1, \beta \alpha_{\max}^\ell\} u_\ell$$

and terminate the execution of the algorithm.

The justification of this procedure is based on the following lemma, proved in [9].

Lemma 5.1 *Assume that Ω is a closed and convex set, $x_k \in Int(\Omega)$ and $\{u_\ell\} \subset I\!R^n$ is a sequence such that*

$$\lim_{\ell \to \infty} u_\ell = \bar{d}_k,$$

where $x_k + \bar{d}_k \in \Omega$.

For all $\ell \in I\!N$ we define

$$\alpha_{\max}^\ell = \max\{\alpha \geq 0 \mid x_k + \alpha u_\ell \in \Omega\}$$

and

$$\bar{u}_\ell = \min(\alpha_{\max}^\ell, 1) u_\ell.$$

Then,
$$\lim_{\ell\to\infty} \bar{u}_\ell = \bar{d}_k.$$

Theorem 5.1 *The direction d_k computed at Algorithm 5.1 satisfies (4).*

Proof. Assume first that Algorithm 5.1 terminates at Step 3. Then $u_\ell = \bar{d}_k$ and $\alpha^\ell_{\max} \geq 1$. This implies (4) in the case that $1 \leq \beta\alpha^\ell_{\max}$. If $\beta\alpha^\ell_{\max} < 1$ then, by the convexity of Q_k,

$$Q_k(\beta\alpha^\ell_{\max}u_\ell) \leq \beta\alpha^\ell_{\max}Q_k(\bar{d}_k) \leq \beta Q_k(\bar{d}_k) \leq \eta Q_k(\bar{d}_k).$$

Therefore, (4) is satisfied.

Assume now that Algorithm 5.1 does not terminate at Step 3. Therefore, $x_k + u_\ell \notin \Omega$ for all ℓ. So, $\alpha^\ell_{\max} < 1$ for all ℓ. By Lemma 5.1, since the box-constrained solver is globally convergent,
$$\lim_{\ell\to\infty} \alpha^\ell_{\max}u_\ell = \bar{d}_k.$$
Therefore,
$$\lim_{\ell\to\infty} Q_k(\alpha^\ell_{\max}u_\ell) = Q_k(\bar{d}_k). \tag{49}$$
By the convexity of Q_k,
$$Q_k(\beta\alpha^\ell_{\max}u_\ell) \leq \beta Q_k(\alpha^\ell_{\max}u_\ell).$$
So, by (49), since $\eta < \beta$ and the box-constrained solver is globally convergent, we have that, for ℓ large enough,
$$Q_k(\beta\alpha^\ell_{\max}u_\ell) \leq \eta L(u_\ell, y_\ell).$$
Thus, (48) is satisfied. This completes the proof. □

6 Nonnegatively constrained minimization

In the previous sections we saw that, for solving the quadratic programming problem (43) using the dual approach, we need to minimize a (not necessarily strictly) convex quadratic with nonnegativity constraints. So, in this section we assume that the problem to be solved is (47). From now on, we define:

$$q(y) = \frac{1}{2}y^T By + \hat{b}^T y.$$

The solutions of (47) are the Lagrange multipliers associated to the unique solution of (43). Moreover, since the feasible set of (43) is a polytope with nonempty interior, the Mangasarian-Fromovitz constraint qualification is satisfied at every feasible point of (43). This implies (see [27]) that the set of Lagrange multipliers is compact. So, the problem (47) has a nonempty and compact solution set. We are going to prove that this implies that the level sets of the objective function in the nonnegative orthant are also compact. This is a crucial step for ensuring the effectivity of iterative algorithms for solving this problem.

Theorem 6.1. *Let y_* be a solution of (47), $c \geq q(y_*)$. Define*

$$S_c = \{y \in I\!R^m_+ \mid q(y) \leq c\}.$$

Then, S_c is bounded.

Proof. Assume, by contradiction, that S_c is unbounded. Then, there exists a sequence $\{y_k\} \subset S_c$ such that

$$\lim_{k\to\infty} \|y_k\| = \lim_{k\to\infty} \|y_k - y_*\| = \infty. \tag{50}$$

By the definition of S_c and the convexity of q, we have that, for all $k \in I\!N$,

$$\frac{c - q(y_*)}{\|y_k - y_*\|} \geq \frac{q(y_k) - q(y_*)}{\|y_k - y_*\|} \geq \frac{\nabla q(y_*)^T (y_k - y_*)}{\|y_k - y_*\|} \geq 0. \tag{51}$$

Moreover, by (50),

$$\lim_{k\to\infty} \frac{c - q(y_*)}{\|y_k - y_*\|} = 0. \tag{52}$$

Let K_1 be an infinite subset of indices such that

$$\lim_{k\in K_1} \frac{y_k - y_*}{\|y_k - y_*\|} = v. \tag{53}$$

By (51), (52) and (53) we obtain:

$$0 \leq \nabla q(y_*)^T v = \lim_{k\in K_1} \nabla q(y_*)^T \frac{y_k - y_*}{\|y_k - y_*\|} = 0. \tag{54}$$

Now,

$$\frac{q(y_k) - q(y_*) - \nabla q(y_*)^T (y_k - y_*)}{\|y_k - y_*\|^2} = \frac{1}{2} \frac{(y_k - y_*)^T}{\|y_k - y_*\|} B \frac{(y_k - y_*)}{\|y_k - y_*\|}. \tag{55}$$

So, by (50), (54) and (55),

$$v^T B v = 0.$$

Therefore, by (54),

$$q(y_* + tv) = q(y_*) + t\nabla q(y_*)^T v + \frac{1}{2} t^2 v^T B v = q(y_*) \text{ for all } t \in I\!R. \tag{56}$$

Since the set of solutions is bounded and $v \neq 0$, (56) implies that there exists $\bar{t} > 0$ such that

$$y_* + \bar{t} v \notin I\!R^m_+.$$

So, there exists $k_0 \in K_1$ such that for all $k \in K_1, k \geq k_0$,

$$\frac{\bar{t}}{\|y_k - y_*\|} < 1$$

and

$$\left(1 - \frac{\bar{t}}{\|y_k - y_*\|}\right) y_* + \frac{\bar{t}}{\|y_k - y_*\|} y_k = y_* + \frac{\bar{t}}{\|y_k - y_*\|}(y_k - y_*) \notin I\!R^m_+.$$

Thus, a convex combination of y_k and y_* does not belong to $I\!R^m_+$. This implies that either y_k or y_* do not belong to $I\!R^m_+$, which is a contradiction. □

The algorithm used to solve (47) is inspired in the papers [5, 12, 25]. The main difference with the problems addressed in those papers is that in (47) the feasible region is unbounded. However, as we proved above, the level sets of the objective function are bounded. So, the convergence results of [5, 12, 25] hold exactly in the same way as in those papers. The proposed method is an active-set method with a special procedure for dropping constraints.

As in [12, 25], let us divide the feasible set Ω into disjoint faces. For all $I \subset \{1, \dots, m\}$, we define

$$F_I = \{y \in I\!R^m_+ \mid y_i = 0 \text{ if } i \in I, \; y_i > 0 \text{ otherwise}\}.$$

The closure of F_I is denoted by $\bar{F}_I$. Let $[F_I]$ denote the smallest linear manifold that contains F_I, and S_I denote the subspace obtained by the parallel translation of $[F_I]$. As in [12], $-\nabla q(y)$ will be called the *antigradient.*

Given $y \in F_I$, the orthogonal projection of $-\nabla q(y)$ on S_I will be called the *internal antigradient* and denoted by $g_I(y)$. The *chopped antigradient* (see [19, 25]) $g_C(y)$ is defined for $y \in I\!R^m_+$ as follows

$$[g_C(y)]_i = \begin{cases} -\frac{\partial q}{\partial y_i}(x), & \text{if } \; [y]_i = 0 \; \text{ and } \; \frac{\partial q}{\partial y_i}(y) < 0, \\ 0, & \text{otherwise}, \end{cases}$$

where $i = 1, \dots, m$. Observe that $[g_C(y)]_i = 0$ if $[y]_i > 0$.

Since $g_C(y) \perp S_I$, we have that

$$g_I(y) \perp g_C(y).$$

Denote $g_P(y) = g_I(y) + g_C(y)$. The vector $g_P(y)$ will be called the *projected antigradient.* Note that $y \in I\!R^m_+$ is a stationary point of problem (47) if and only if $g_P(y) = 0$. In general, the mapping $g_P(y)$ is not continuous, nevertheless, $y_k \to y$ and $g_P(y_k) \to 0$ imply that $g_P(y) = 0$ (see [13]).

Given $y_k \in I\!R^m_+$, a sub-algorithm is available that may compute a new iterate y_{k+1}. We assume that the sub-algorithm has the following properties (see [12]):

P1. $q(y_{k+1}) < q(y_k)$.

P2. If $y_k \in F_I$ then $y_{k+1} \in \bar{F}_I$.

P3. If $\{y_k, y_{k+1}, y_{k+2}, \dots\} \subset F_I$ is a set of infinitely many iterates generated by the sub-algorithm, then $g_I(y_k) \to 0$.

The sub-algorithm is essentially an unconstrained minimization method that handles the free variables in the face F_I.

Below we present our main model algorithm.

Algorithm 6.1. Assume that $y_0 \in I\!R^m_+$ is an arbitrary initial point, $TOL \in (0,1)$, Let F_I be the face that contains the current iterate y_k. The new iterate y_{k+1} is computed as follows.

Step 1. *Decide whether the current face must be abandoned or not.*

If $\|g_P(y_k)\| = 0$ (y_k is a stationary point), stop.

If

$$\frac{\|g_C(y_k)\|}{\|g_P(y_k)\|} \geq TOL,$$

compute y_{k+1} at Step 2, else compute y_{k+1} using the sub-algorithm.

Step 2. *Abandoning the current face.*

Define $\alpha_k > 0$ such that

$$q(y_k + \alpha_k g_C(y_k)) \leq q(y_k + \alpha g_C(y_k)) \;\; \forall \;\; \alpha \geq 0.$$

Set $y_{k+1} = y_k + \alpha_k g_C(y_k)$.

Slight variations of the arguments used in [12] allow one to prove the following theorem.

Theorem 6.1. *Algorithm 6.1 is well defined and, given any $\varepsilon > 0$ there exists $k \in I\!N$ such that*

$$\|g_P(y_k)\| \leq \varepsilon.$$

Remark. The definition of Algorithm 6.1 must be completed by the introduction of the internal algorithm used within the faces. In some cases, as we will see in the numerical experiments, the structure of the problem is such that the conjugate-gradient method is the best choice. In other cases the damped proximal-point method, which will be introduced in the next section, will be preferred. In practice, and due to availability reasons, in the cases in which the conjugate-gradient method is used inside the faces, the nonnegatively constrained minimizer used will be a quadratic adaptation of GENCAN (see [6]). Due to the bounded level set property, the convergence established in Theorem 6.1 holds for GENCAN as well.

7 Damped proximal-point minimization of a convex quadratic

The internal algorithm used within the faces is, essentially, an unconstrained minimization method. Therefore, this algorithm might generate an iterate not belonging to the nonnegative orthant. In this case, we need that the objective function value at the "breakpoint" on the boundary of the current face should be smaller than the objective function value at the previous iterate. In order to guarantee this essential property, the unconstrained optimization algorithm will be based on descent directions and the new iterate will be defined as the minimizer along the half-line determined by some search direction, when this minimizer exists. If the one-dimensional minimizer does not exist, the search direction must be such that the objective function tends monotonically to $-\infty$ along the generated half-line. In this way, the internal minimization algorithm will generate a monotonic sequence of points within the current face or will be able to stop on the boundary of the nonnegative orthant with a decrease of the objective function value.

In this section we define an adequate algorithm for our purposes. It will be called the DPP (Damped proximal-point) method. In the context of Algorithm 6.1 the DPP method modifies only the free variables of F_I. For simplicity of exposition we call m the number of free variables of F_I and we define the quadratic objective function as:

$$q(y) = \frac{1}{2} y^T B y + \hat{b}^T y, \tag{57}$$

where $\hat{b}, y \in I\!R^m$ and $B \in I\!R^{m \times m}$ is symmetric and positive semidefinite.

The problem of minimizing $q(y)$ may have solutions or not. The unconstrained minimizers of this quadratic function are the zeros of the linear system $\nabla q(y) = 0$. So, at an unconstrained minimizer of $q(y)$ we have that

$$By = -\hat{b}. \tag{58}$$

Algorithm 7.1. *Damped proximal-point method*

Given $\sigma > 0$, $y_0 \in I\!R^n$ set $k \leftarrow 0$. Execute the following steps.

Step 1. *Finite convergence criterion*

If $\nabla q(y_k) = 0$, terminate the execution of the algorithm. The point y_k is a solution.

Step 2. *Search direction*

Compute $d_k \in I\!R^m$ by:

$$(B + \sigma I) d_k = -\nabla q(y_k). \tag{59}$$

Step 3. *Test if one-dimensional minimum exists*

If $d_k^T B d_k = 0$ terminate the execution of the algorithm. The function is linear along the half-line $\{y_k + t d_k\}_{t>0}$ and tends monotonically to $-\infty$.

Step 4. *One-dimensional minimizer*

Compute

$$\alpha_k = -\frac{\nabla q(y_k)^T d_k}{d_k B d_k}. \tag{60}$$

Step 5. *Next iterate*

Compute

$$y_{k+1} = y_k + \alpha_k d_k.$$

Set $k \leftarrow k+1$ and go to Step 1.

Proposition 7.1. *If d_k is defined by (59), $\nabla q(y_k) \neq 0$ and α_k is defined by (60), then*

$$q(y_k + t d_k) < q(y_k) \;\; \forall \;\; t \in (0, \alpha_k) \tag{61}$$

Proof. The inequality (61) holds because d_k is a descent direction, the objective function is a parabola and α_k is its minimizer. □

When $t_k \equiv 1$, the iterative process defined by

$$y_{k+1} = y_k + t_k d_k, \;\; d_k = -[B + \sigma I_m]^{-1}[B y_k + \hat{b}], t_k > 0 \tag{62}$$

is the *proximal point method* [39] for minimizing $q(y)$.

Lemma 7.1. *Assume that B is symmetric and positive definite, and λ_0, ρ are the smallest and largest eigenvalues of B respectively. Then, for all $\sigma \geq 0, v \neq 0$,*

$$\frac{(v^T(B+\sigma I)^{-1}v)^2}{(v^T B^{-1} v)(v^T(B+\sigma I)^{-1} B (B+\sigma I)^{-1} v)} \geq \frac{4\lambda_0 \rho(\rho+\sigma)(\lambda_0+\sigma)}{(2\lambda_0\rho + \sigma(\rho+\lambda_0))^2}. \tag{63}$$

Proof. Since $\sigma \geq 0$, $B + \sigma I$ is symmetric and positive definite. Hence, there exists a nonsingular matrix C such that $B + \sigma I = CC^T$. Defining $w = C^{-1} v \neq 0$ and $Q = C^{-1} B C^{-T}$, we obtain:

$$\frac{(v^T(B+\sigma I)^{-1}v)^2}{(v^T B^{-1} v)(v^T(B+\sigma I)^{-1} B (B+\sigma I)^{-1} v)} = \frac{(w^T w)^2}{(w^T C^{-1} B C^{-T} w)(w^T C^T B^{-1} C w)}$$

$$= \frac{(w^T w)^2}{(w^T Q w)(w^T Q^{-1} w)}.$$

So, by the Kantorovich Inequality ([33], p. 151),

$$\frac{(w^T w)^2}{(w^T Q w)(w^T Q^{-1} w)} \geq \frac{4\tau\gamma}{(\tau+\gamma)^2}, \tag{64}$$

where τ and γ are the smallest and largest eigenvalues of Q respectively. Since Q is similar to $(B+\sigma I)^{-1} B$, we have

$$\tau = \frac{\lambda_0}{\lambda_0 + \sigma}, \text{ and } \gamma = \frac{\rho}{\rho+\sigma}. \tag{65}$$

Substituting (65) into (64), we obtain (63). So, the proof is complete. $\square$

Theorem 7.1. *Assume that a solution of (58) exists. Let y_* be the unique solution of the problem*

$$\text{Minimize } \|y - y_0\| \ \ s.t. \ \ By = -\hat{b}. \tag{66}$$

Let $\{y_k\}$ be the sequence generated by Algorithm 7.1. Then, one of the following possibilities hold:

(a) The sequence terminates at Step 1 with $y_k = y_*$;

(b) The sequence $\{y_k\}$ converges to $\{y_*\}$ at a linear rate in an appropriate norm.

Proof. Since $By_* = b$, we have that

$$\nabla q(y_k) = By_k + \hat{b} = B(y_k - y_*).$$

By (59), for all k such that $\nabla q(y_k) \neq 0$ we have that

$$d_k \neq 0 \text{ and } (B + \sigma I)d_k = -B(y_k - y_*).$$

So,

$$\sigma d_k = -B(y_k - y_*) - Bd_k.$$

This implies that d_k belongs to the column-space of B. Therefore, $d_k^T B d_k > 0$ and, so, termination at Step 3 of the algorithm is not possible.

Moreover, $\alpha_0 d_0 + \ldots + \alpha_{k-1} d_{k-1}$ belongs to the column-space of B for all k, therefore $y_k - y_0$ is in the column-space of B for all k.

Thus, if the algorithm terminates at Step 1 with $\nabla q(y_k) = 0$, y_k must be the closest point to y_0 that satisfies $By + \hat{b} = 0$.

Consider the case in which the algorithm generates an infinite sequence. Assume first that B is positive definite.

Define $E(y) = \frac{1}{2}(y - y_*)^T B(y - y_*)$, $v_k = y_k - y_*$, and $g_k = Bv_k$. Since

$$\alpha_k = \frac{d_k^T (B + \sigma I) d_k}{d_k^T B d_k} = \frac{g_k^T (B + \sigma I)^{-1} g_k}{g_k^T (B + \sigma I)^{-1} B (B + \sigma I)^{-1} g_k},$$

we have that

$$E(y_k) - E(y_{k+1}) = \frac{1}{2} v_k^T B v_k - \frac{1}{2} v_{k+1}^T B v_{k+1}$$

$$= \alpha_k g_k^T (B + \sigma I)^{-1} g_k - \frac{1}{2} \alpha_k^2 g_k^T (B + \sigma I)^{-1} B (B + \sigma I)^{-1} g_k$$

$$= \frac{(g_k^T (B + \sigma I)^{-1} g_k)^2 E(y_k)}{(g_k^T (B + \sigma I)^{-1} B (B + \sigma I)^{-1} g_k)(g_k^T B^{-1} g_k)}.$$

Then,

$$E(y_{k+1}) = \{1 - \frac{(g_k^T (B + \sigma I)^{-1} g_k)^2}{(g_k^T (B + \sigma I)^{-1} B (B + \sigma I)^{-1} g_k)(g_k^T B^{-1} g_k)}\} E(y_k).$$

Thus, by Lemma 7.1,

$$E(y_{k+1}) \leq \{1 - \frac{4\lambda_0\rho(\rho+\sigma)(\lambda_0+\sigma)}{[2\lambda_0\rho+\sigma(\rho+\lambda_0)]^2}\}E(y_k) = \frac{(\rho-\lambda_0)^2\sigma^2}{[2\lambda_0\rho+\sigma(\rho+\lambda_0)]^2}E(y_k).$$

Since

$$\frac{(\rho-\lambda_0)^2\sigma^2}{[2\lambda_0\rho+\sigma(\rho+\lambda_0)]^2} < 1,$$

we deduce that $E(y_k)$ converges to zero. Since B is symmetric and positive definite, y_k converges to y_*.

It remains to discuss the case in which B is singular. Since B is symmetric and positive semidefinite, there exists an orthogonal matrix U such that

$$B = U\begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix}U^T, \tag{67}$$

where $D \in R^{l\times l}$ is positive definite and $l = \text{rank}(B)$.

Pre-multiplying (62) by U^T and replacing t_k by α_k, we obtain

$$\begin{aligned} z_{k+1} = U^T y_{k+1} &= U^T y_k - \alpha_k[\begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} + \sigma I_m]^{-1}[\begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix}U^T y_k + U^T\hat{b}] \\ &= z_k - \alpha_k[\begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} + \sigma I_m]^{-1}[\begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix}z_k + \begin{pmatrix} b_1 \\ 0 \end{pmatrix}]. \end{aligned} \tag{68}$$

So,

$$\begin{pmatrix} z_{k+1}^{(1)} \\ z_{k+1}^{(2)} \end{pmatrix} = \begin{pmatrix} [D+\sigma I_l]^{-1}\{[(1-\alpha_k)D + \sigma I_l]z_k^{(1)} - \alpha_k b_1\} \\ z_k^{(2)} \end{pmatrix}. \tag{69}$$

Formula (69) describes the behavior of the iterative process in the space z. In particular, observe that $z_k^{(2)} = z_0^{(2)}$ for all k. Applying Lemma 7.1 with $B = D$, we can prove that $z_k^{(1)}$ converges to $z_*^{(1)}$. Since $z^{(2)}$ is fixed, it follows from (69) that $\{z_k\}$ is convergent. So, y_k converges to the unique solution of (66) at a linear rate in the norm $\sqrt{E(y)}$. □

Theorem 7.2. *Assume that a solution of (58) does not exist. Then the sequence generated by Algorithm 7.1 satisfies:*

$$\lim_{k\to\infty} \|y_k\| = \infty.$$

Proof. Since (58) has no solution, B must be singular. Moreover, B has the form (67) and $\hat{b}$ does not belong to the its column-space. Since B is symmetric and positive semidefinite, substituting (67) in (58), we obtain

$$U\begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix}U^T y = -\hat{b}.$$

Let $U^T y = z = \begin{pmatrix} z^{(1)} \\ z^{(2)} \end{pmatrix}$, $-U^T\hat{b} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$. Then,

$$\begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} z^{(1)} \\ z^{(2)} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}.$$

Since $\hat{b}$ does not belong to the column-space of B we have that $b_2 \neq 0$. Pre-multiplying (62) by U^T and replacing t_k by α_k, we obtain

$$z_{k+1} = \begin{pmatrix} z_{k+1}^{(1)} \\ z_{k+1}^{(2)} \end{pmatrix} = \begin{pmatrix} (D+\sigma I_l)^{-1}([(1-\alpha_k)D + \sigma I_l]z_k^{(1)} + \alpha_k b_1) \\ z_k^{(2)} + \frac{1}{\sigma} b_2 \end{pmatrix}$$

$$= \begin{pmatrix} (D+\sigma I_l)^{-1}([(1-\alpha_k)D + \sigma I_l]z_k^{(1)} + \alpha_k b_1) \\ z_0^{(2)} + \frac{k+1}{\sigma} b_2 \end{pmatrix},$$

where $z_0 = U^T y_0$. Without loss of the generality, we can set $z_0^{(2)} = 0$. Then,

$$\|y_k\| = \|U z_k\| = \|z_k\| \geq \|z_k^{(2)}\| = \frac{k}{\sigma}\|b_2\| \to \infty$$

as k tends to ∞. □

We finish this section showing how to compute the search direction in Algorithm 7.1 when the matrix has the form

$$B = AG^{-1}A^T,$$

where $A \in I\!R^{m\times n}$, and $G \in I\!R^{n\times n}$ is symmetric and positive definite. So, defining $z = d_k, v = -\nabla q(y_k)$, (59) takes the form

$$[AG^{-1}A^T + \sigma I]z = v. \tag{70}$$

In principle, one solution of (70) involves Cholesky factorizations of G and $[AG^{-1}A^T + \sigma I]$. The final factorization can be dense even in cases that G and A are very sparse. For this reason we decomposed the solution of (70) in order to make it more suitable for sparse calculations.

Let us define

$$r = G^{-1}A^T z.$$

So,

$$-A^T z + Gr = 0$$

and, by (70),

$$\sigma z + Ar = v.$$

Therefore (70) is equivalent to:

$$\begin{pmatrix} \sigma I & A \\ -A^T & G \end{pmatrix} \begin{pmatrix} z \\ r \end{pmatrix} = \begin{pmatrix} v \\ 0 \end{pmatrix}.$$

So, by elimination,

$$\begin{pmatrix} \sigma I & A \\ 0 & G + A^T A/\sigma \end{pmatrix} \begin{pmatrix} z \\ r \end{pmatrix} = \begin{pmatrix} v \\ A^T v/\sigma \end{pmatrix}. \tag{71}$$

Since G is positive definite, $G + A^T A/\sigma$ is positive definite too. Therefore, r can be obtained from (71) solving the system

$$(\sigma G + A^T A)r = A^T v. \tag{72}$$

The solution of (72) may be obtained using the Cholesky decomposition of $\sigma G + A^T A$. Finally,

$$z = (v - Ar)/\sigma.$$

When the inverse of G can be easily computed there are no special difficulties with the computation of the search direction using (70) as it stands. In these cases, we can use the quadratic version of GENCAN [6] for solving the nonnegatively constrained quadratic minimization problem.

8 Numerical experiments

We implemented the Inexact Variable Metric (Algorithm 2.1) in the way described below.

8.1 Hessian approximation

Two different definitions for G_k were employed:

(a) **Hessian choice.** A positive-definite modification of the true Hessian as described by Eskow and Schnabel in [21].

(b) **ISPG choice.** The diagonal spectral approximation given by $G_k = (1/\lambda_k^{\text{spg}})I$, where

$$\lambda_k^{\text{spg}} = \min\{\lambda_{\max}, \max\{\lambda_{\min}, \frac{\|x_k - x_{k-1}\|^2}{(x_k - x_{k-1})^T(g_k - g_{k-1})}\}\}$$

if $(x_k - x_{k-1})^T(g_k - g_{k-1}) > 0$, whereas $\lambda_k^{\text{spg}} = \lambda_{\max}$ otherwise.

The arbitrary initial spectral steplength $\lambda_0 \in [\lambda_{\min}, \lambda_{\max}]$ was computed as

$$\lambda_0^{\text{spg}} = \min\{\lambda_{\max}, \max\{\lambda_{\min}, \frac{\|\bar{x} - x_0\|^2}{(\bar{x} - x_0)^T(g(\bar{x}) - g_0)}\}\}$$

if $(\bar{x} - x_0)^T(g(\bar{x}) - g_0) > 0$, and $\lambda_k^{\text{spg}} = \lambda_{\max}$ otherwise. Here $\bar{x} = x_0 - t_{\text{small}} g(x_0)$ and $t_{\text{small}} = \max(\epsilon_{\text{rel}}\|x\|_\infty, \epsilon_{\text{abs}})$.

8.2 Solution of the subproblem at Step 1

The approximate minimization of $Q_k(d)$ subject to $A(x_k + d) \leq b$ was performed using the dual approach. The dual problem becomes the minimization of a convex quadratic on the nonnegative orthant. In the case in which the matrix G_k is an exact or modified Hessian the nonnegative quadratic minimization is done using the active set method described in Section 6. We use the DPP method described in Section 7 within the faces. In the case in which the Hessian approximation is the spectral diagonal matrix described above, the nonnegative quadratic minimization problem is solved using a quadratic-oriented version of GENCAN [6].

The initial point (dual variables) in the nonnegatively constrained quadratic minimization problems is the set of Lagrange multipliers estimated at the previous iteration.

The trial point that satisfies (4) is computed using the procedure described in Section 5. This means that, at each iteration of the nonnegative quadratic solver, we check feasibility of u_ℓ and we test the inequality (48). In this way, a trial point that satisfies (4) is necessarily obtained.

8.3 Computation of the steplength

The computation of $\alpha_{\rm new}$ in (5) uses safeguarded one-dimensional quadratic interpolation. We take $\alpha_{\rm new} \leftarrow \alpha/2$ when the minimizer of the one-dimensional quadratic lies outside $[\sigma_1, \sigma_2\alpha]$.

8.4 Practical convergence criterion

In the first set of (CUTE) problems we terminate the execution of the algorithm when

$$\|d_k\|_\infty \le \epsilon_1.$$

In the set of large-scale location problems, we terminate when

$$\min\{\alpha^k_{\max}, 1\}\|d_k\|_\infty \le \epsilon_1,$$

where, as usually,

$$\alpha^k_{\max} = \max\{t \ge 0 \mid x_k + td_k \in \Omega\}.$$

8.5 Parameters

We used $\eta = \beta\eta'$, $\gamma = 10^{-4}$, $\sigma_1 = 0.1$, $\sigma_2 = 0.9$, $M = 10$, $\epsilon_1 = 10^{-6}$, $\epsilon_2 = 10^{-6}$, $\epsilon_{\rm rel} = 10^{-7}$, $\epsilon_{\rm abs} = 10^{-10}$ for Algorithm 2.1; $\beta = 0.85$, and η'=0.8 for Algorithm 5.1 and $TOL = 0.9$ for Algorithm 6.1. In Algorithm 7.1, σ was chosen as follows:

$$\sigma = \tau \frac{\max_{ij} |[A^t A]_{ij}|}{\max_{ij} |[G_k]_{ij}|}.$$

In general, we use $\tau = 10^{-6}$. However, we have some evidence that in the iterations in which G_k is the true Hessian or a very small modification of the true Hessian, $\tau = 0.1$ is better.

8.6 Problems in the CUTE collection

We considered all the problems of type (1)-(2) for which an interior initial point is provided in the CUTE collection [10]. Box constraints were considered as ordinary linear inequality constraints.

The computations were done on an 1.9Ghz Intel Pentium IV Computer with 1Gb of RAM. Codes are in Fortran and the compiler used was GNU Fortran 2.95.4, with the optimization option "–O4".

Tables 1 and 2 show the performance of the Hessian version and the ISPG version of IVM on this set of problems. In these tables n is the number of variables, m is the number of constraints, OUTIT is the number of iterations, FE is the number of function and gradient evaluations, HM is the number of times the Hessian was not positive definite and was modified by the Eskow-Schnabel subroutine, INNIT is the number of iterations performed by the nonnegative quadratic minimization solver, TIME is the CPU time in seconds and $f(x^*)$ is the objective function value at solution found. In Table 1 some comparison with two well-known algorithms is also given. The algorithms are LOQO (see [40, 41]) and FSQP (see [35, 42, 43]). The code of FQSP was provided by AEM Design [1] and the experiments of LOQO were taken from [3]. In the case of LOQO, since the experiments were done on a different computer environment, we report the number of function/gradient evaluations as performance parameter. In some problems (indicated by NA) this information was not available. We proceed in the same way with respect to FSQP. In the case of FSQP, since the experiments were run by us in the same conditions as we did in the case of the IVM method, computer times are available. They are not reported here for economy of space and because they are proportional to the number of functional evaluations, so they add nothing to the comparison. In these problems FSQP seems to be more efficient than the IVM method. This is probably due to the efficient way in which FSQP discards constraints that are strongly satisfied at a current iterate. We also observe that, in this limited number of examples, the number of function/gradient evaluations used by LOQO is, on average, similar to the one used by the IVM method. The final functional values were always the same for the three methods.

Clearly, the IVM-Hessian algorithm is much more efficient than the ISPG method when the true Hessian is positive definite everywhere (problems HS35, HS35I, HS76 and HS76I). In the cases in which the Eskow-Schnabel algorithm modifies the Hessian, the performance of both algorithms is more or less the same. Problems KSIP and SIPOXX are Linear Programming problems. In these cases the Hessian is null, therefore the Eskow-Schnabel algorithm modifies this matrix at every iteration. The ISPG method also replaces the Hessian by a small diagonal matrix in this case, and this is the reason why both methods behave very similarly.

It is interesting to observe that in all the iterations of all these problems the first trial point was accepted, so decreasing the steplength was never necessary and the number of functional evaluations coincides with the number of iterations. This is probably an effect of the nonmonotone strategy.

8.7 Location problems

We consider the family of *location* problems introduced in [8]. In [8] the 2-dimensional case, in which the projections onto the feasible set are easy to compute, was considered. Here we deal with the 3-dimensional case.

Given a set of *npol* disjoint polyhedra $P_1, P_2, \ldots, P_{\text{npol}}$ in $I\!R^3$ we wish to find the point $z^{\text{npol}} \in P_{\text{npol}}$ that minimizes the sum of the distances to the other polyhedra. Therefore, the problem is

$$\text{Minimize}_{\,(z^1,\ldots,z^{\text{npol}})} \frac{1}{\text{npol}-1} \sum_{i=1}^{\text{npol}-1} \|z^i - z^{\text{npol}}\|_2$$

Problem	n	m	OUTIT	FE	FE(LOQO)	FE(FSQP)	HM	INNIT	TIME	$f(x^*)$
HS24	2	3	13	14	13	5	14	29	0.03	-1.000
HS35	3	1	8	9	10	7	0	1	0.00	0.1111
HS35I	3	1	8	9	NA	7	0	1	0.00	0.1111
HS36	3	1	10	11	16	2	11	16	0.01	$-3300.$
HS37	3	2	46	47	11	11	47	50	0.01	$-3456.$
HS44NEW	4	6	10	11	12	5	11	33	0.00	-15
HS76	4	3	8	9	11	7	0	5	0.01	-4.682
HS76I	4	3	8	9	NA	7	0	5	0.00	-4.682
KSIP	20	1001	12	13	47	47	13	669	2.92	0.5758
SIPOW1	2	2000	15	16	NA	6	16	390	0.74	-1.000
SIPOW1M	2	2000	17	18	NA	6	18	387	0.75	-1.000
SIPOW2	2	2000	14	15	NA	6	15	254	0.48	-1.000
SIPOW2M	2	2000	15	16	NA	6	16	257	0.44	-1.000
SIPOW3	4	2000	9	10	NA	3	10	308	0.70	0.5347
SIPOW4	4	2000	9	10	NA	3	10	603	1.30	0.2724
SIPOW1	2	10000	19	20	17	6	20	1019	13.15	-1.000
SIPOW1M	2	10000	20	21	16	6	21	1113	14.77	-1.000
SIPOW2	2	10000	18	19	NA	6	19	681	8.05	-1.000
SIPOW2M	2	10000	19	20	NA	6	20	730	9.21	-1.000
SIPOW3	4	10000	9	10	19	3	10	686	12.24	0.5357
SIPOW4	4	10000	9	10	19	3	10	1437	26.33	0.2728

Table 1: IVM-Hessian ($\eta' = 0.8$, $\tau = 10^{-6}$), LOQO and FSQP on CUTE problems.

Problem	n	m	OUTIT	FE	INNIT	TIME	$f(x^*)$
HS24	2	3	11	12	17	0.00	-1.000
HS35	3	1	17	18	15	0.00	0.1111
HS35I	3	1	17	18	15	0.01	0.1111
HS36	3	1	12	13	30	0.01	-3300
HS37	3	2	28	29	35	0.00	-3456
HS44NEW	4	6	13	14	29	0.00	-15
HS76	4	3	14	15	19	0.00	-4.682
HS76I	4	3	14	15	19	0.02	-4.682
KSIP	20	1001	25	26	1086	4.19	0.5758
SIPOW1	2	2000	15	16	641	1.07	-1.000
SIPOW1M	2	2000	13	14	656	1.10	-1.000
SIPOW2	2	2000	11	12	349	0.53	-1.000
SIPOW2M	2	2000	12	13	323	0.48	-1.000
SIPOW3	4	2000	11	12	1355	3.51	0.5347
SIPOW4	4	2000	11	12	1398	4.18	0.2724
SIPOW1	2	10000	15	16	2039	23.73	-1.000
SIPOW1M	2	10000	15	16	2463	29.32	-1.000
SIPOW2	2	10000	15	16	1485	14.72	-1.000
SIPOW2M	2	10000	14	15	1535	15.25	-1.000
SIPOW3	4	10000	13	14	4711	81.44	0.5357
SIPOW4	4	10000	13	14	5176	104.21	0.2728

Table 2: ISPG on CUTE problems.

$$\text{subject to} \quad z^i \in P_i,\ i = 1, \ldots, \text{npol}.$$

This problem has $3 \times \text{npol}$ variables. The number of (linear inequality) constraints is $\sum_{i=1}^{\text{npol}} \nu_i$, where ν_i is the number of constraints that define the polyhedron P_i. Writing $x = (z_1^1, z_2^1, z_3^1, \ldots, z_1^{\text{npol}}, z_2^{\text{npol}}, z_3^{\text{npol}})$, the Hessian of this problem, which is singular and positive semidefinite, has the form

$$H(x) = \left(\begin{array}{cccccc|c} H_1 & & & & & & -H_1 \\ & H_2 & & & & & -H_2 \\ & & H_3 & & & & -H_3 \\ & & & H_4 & & & -H_4 \\ & & & & \ddots & & \vdots \\ & & & & & H_{\text{npol}-1} & -H_{\text{npol}-1} \\ \hline -H_1 & -H_2 & -H_3 & -H_4 & \ldots & -H_{\text{npol}-1} & \sum_{i=1}^{\text{npol}-1} H_i \end{array}\right) \tag{73}$$

Each H_i is a 3×3 matrix which depends on z^i and z^{npol}. In order to obtain a positive definite matrix B_k at each iteration of the algorithm, it is sufficient to add a small positive number to the diagonal of $H(x)$. Note that the pattern (73) does not produce fill-in in the Cholesky factorization. Moreover, the matrix $\sigma B + A^T A$ used in (72) exhibits the same pattern.

We generated 54 problems of this class, varying *npol* and choosing randomly the location of the polygons and the number of half-spaces that define each polytope. The test problems were solved with IVM. As in [9], the interior initial guess for the primal variables is generated together with the problem data. The initial guess for the vector of Lagrange multipliers (variables of the nonnegatively constrained quadratic minimization problems) was the null vector.

Since we know that the matrices B_k are tiny modifications of the Hessian and the factorization work required for an IVM iteration is similar to the one required by the ISPG iteration, we only used IVM for solving this set of problems.

Table 3 displays the performance of two versions of the IVM-Hessian algorithm: solving the problems accurately ($\eta = 0.90$, $\theta = 0.95$) and inaccurately ($\eta = 0.50$, $\theta = 0.75$). The columns mean: *Problem*, problem number; n, number of variables; m, number of constraints; OUTIT, iterations of IVM needed to satisfy the convergence criterion; INNIT, total number of iterations of the quadratic solver; TIME, CPU time in seconds; $f(x^*)$, function value at the solution found. The practical convergence criterion was satisfied in all the cases. In this table we also report the performance of FSQP on some problems. In the last column of the table we give the number of function/gradient evaluations used by FSQP to achieve convergence. The final functional value was always the same for the three methods. We do not report computer time of FSQP because, in this case, it would be quite unfair. In fact, since FSQP does not use sparse factorization techniques, the numerical algebra overhead is very high and do not reflect the performance that this algorithm would achieve if sparsity were exploited. For example, in the last problem reported here, FSQP used almost six hours to converge. On the other hand, functional evaluations provide a fair performance measure for comparison with the IVM method. It is quite impressive that in all the problems FSQP used much more functional evaluations than the IVM algorithms thus confirming the observations made in [8] for the 2-dimensional case.

As expected, when the subproblems are less accurately solved, the number of outer iterations increases, and the average number of inner iterations needed to solve each subproblem decreases. As a whole, the accurate version was more efficient in 34 problems whereas the inaccurate version won in 20 problems.

9 Conclusions

The fine performance of the SPG method in very large problems [8] motivated the introduction of the inexact spectral projected gradient method [9]. This method should be useful when exact projections are difficult to compute.

Sometimes, the structure of the problem suggests a particular procedure for computing inexact projections. This is the case of the matricial problems considered in [9]. In other cases, using a general quadratic solver seems to be recommendable. The strategy chosen in this paper for solving the quadratic subproblems fits the theoretical requirements of the IVM method because it produces lower bounds of the objective function value at the quadratic solution. This implies that the fulfillment of the inexact criterion for stopping iterations of the subproblem is theoretically guaranteed.

In many cases the computer work required by the ISPG iteration is considerably less than the computer work required by Newton or quasi-Newton iterations. When this is not the case, it seems reasonable to use a Newton-like IVM method. We observed that, for convex problems, the advantage of the Newtonian IVM is clear, but for nonconvex problems it is better to use the ISPG method. Moreover, we showed that, exploiting the sparsity pattern of a positive semidefinite Hessian, to solve very large location problems is possible.

The superiority of the ISPG method over the Newton-like IVM method in nonconvex problems deserves some explanation. When, at the current iteration, the Hessian is not positive semidefinite the Newtonian IVM method adds positive terms to the diagonal in order to obtain the positive definite matrix B_k. This is a common practical procedure in unconstrained minimization and, roughly speaking, it is in this way that trust-region steps are obtained. In unconstrained minimization one replaces a subproblem without solution (minimizing a nonconvex quadratic) by the problem of minimizing the Taylor approximation on a restricted region. Probably, this is the best thing we can do. However, when constraints are present, the original quadratic subproblem (with indefinite or negative definite Hessian) may be solvable and the solution of its diagonal modification might be very different from its true solution. It is not surprising, therefore, that no meaningful differences appear between slight and rough positive definite modifications of the Hessian in constrained problems. Due to its low cost, the ISPG strategy may be even better.

Further research is necessary to exploit all the possible variations of linear constraints for an efficient implementation of IVM and ISPG. The implementation of different quasi-Newton versions of the IVM algorithm is also necessary.

Problem		IVM with accurate subproblems				IVM with inaccurate subproblems				FSQP
n	m	OUTIT	FE	INNIT	TIME	OUTIT	FE	INNIT	TIME	FE
33	10,050	7	7	65	1.28	16	16	61	1.35	68
78	21,868	9	9	618	20.34	20	20	558	21.30	156
84	21,903	9	9	604	20.14	24	24	542	18.03	168
99	25,333	8	8	203	11.41	21	21	279	16.07	198
90	20,841	9	9	428	16.48	21	21	403	16.32	180
141	34,683	9	9	1138	58.77	23	23	936	55.13	282
132	31,154	8	8	526	28.30	21	21	518	29.55	264
174	46,504	9	9	905	75.78	22	22	969	82.99	348
177	45,001	9	9	559	44.81	20	20	740	66.71	354
225	58,807	10	10	1112	116.37	23	23	829	88.77	450
276	69,483	9	9	1297	148.60	24	24	1065	131.69	552
345	86,595	10	10	530	93.58	23	23	563	111.07	
339	88,505	11	11	1291	168.18	23	23	863	132.06	
336	80,235	10	10	3195	346.16	24	24	2789	320.86	
411	109,512	10	10	871	182.06	22	22	867	243.17	
504	138,872	11	11	678	176.38	26	26	971	347.63	
636	160,509	11	11	746	245.54	25	25	522	242.22	
738	196,046	12	12	1045	388.15	25	25	832	395.27	
837	211,050	12	12	1201	446.30	30	30	1387	600.04	
897	235,127	12	12	749	344.22	27	27	526	319.14	
966	263,094	14	14	1071	582.65	32	32	1186	754.96	
978	267,744	12	12	646	428.11	28	28	602	454.03	
1158	306,201	13	13	1134	730.02	30	30	1077	801.92	
1245	335,266	12	12	2119	1413.51	27	27	3142	2197.49	
1353	365,464	12	12	1555	1161.41	30	30	893	787.18	
1416	379,009	18	18	1523	1173.03	37	37	2562	1926.32	
1479	387,914	15	15	1075	1012.29	28	28	827	887.87	
1623	433,965	20	20	1766	1760.34	38	38	1890	1990.88	
1671	445,415	21	21	2605	2551.67	56	56	2846	2736.35	
1761	471,237	20	20	2302	2412.56	46	46	2680	3029.78	
1848	499,401	17	17	1631	1999.98	33	33	1046	1647.56	
1860	481,435	19	19	8096	7757.22	48	48	11395	11128.81	
2133	592,996	17	17	3240	4496.14	44	44	3150	4629.66	
2223	613,562	19	19	1834	3141.99	53	53	1859	3246.47	
2259	590,251	21	21	2389	3683.71	45	45	2019	3459.02	
2337	630,390	25	25	9131	12990.15	40	40	5517	8546.61	
2448	648,757	16	16	976	2163.47	39	39	1044	2193.88	
2562	676,535	17	17	2222	4291.25	44	44	2090	4186.96	
2625	703,565	20	20	1521	3361.72	31	31	1562	3446.29	
2688	724,318	17	17	1532	3487.54	29	29	1207	3300.02	
2823	763,514	20	20	1537	3613.95	45	45	4093	8865.39	
2865	753,101	29	29	3940	8161.02	46	46	3195	7082.61	
2991	795,177	22	22	2692	6452.56	42	42	2600	6455.50	
3048	834,315	21	21	1398	3943.43	68	68	1639	4575.67	
3123	815,656	20	20	2911	7039.55	41	41	1751	4703.06	
3222	877,593	15	15	2473	6580.20	39	39	3599	9540.93	
3327	873,605	25	25	5309	14163.70	39	39	5663	14972.71	
3429	893,267	23	23	3931	11126.43	44	44	2174	6505.21	
3537	946,827	17	17	1552	5444.67	69	69	2833	9013.79	
3621	970,713	13	13	555	2844.54	31	31	1122	4333.27	
3621	1,571,323	13	13	659	4429.44	22	22	565	4072.57	
3621	2,196,476	12	12	788	6736.31	29	29	3003	16044.66	
3621	2,845,050	12	12	1119	10754.51	29	29	1744	13414.26	
3621	3,361,256	12	12	955	10648.48	45	45	2950	22439.15	

Table 3: Performance of IVM and FSQP in the 3D location problems.

Acknowledgements

We are indebted to two anonymous referees whose comments helped a lot to improve this paper.

References

[1] http://www.aemdesign.com/

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

[3] H. Y. Benson, D. F. Shano and R. J. Vanderbei, Interior-point methods for nonconvex nonlinear programming: jamming and comparative numerical testing, Operations Research and Financial Engineering, Princeton University, ORFE-00-02, 2000. (A detailed table with the numerical experiments of this work was found at http://www.princeton.edu/ rvdb/bench.html).

[4] D. P. Bertsekas, On the Goldstein-Levitin-Polyak gradient projection method, *IEEE Transactions on Automatic Control* 21, pp. 174–184, 1976.

[5] R. H. Bielschowsky, A. Friedlander, F. M. Gomes, J. M. Martínez and M. Raydan, An adaptive algorithm for bound constrained quadratic minimization, *Investigación Operativa* 7 pp. 67–102, 1998.

[6] E. G. Birgin and J. M. Martínez, Large-scale active-set box-constrained optimization method with spectral projected gradients, *Computational Optimization and Applications* 23, pp. 101–125, 2002.

[7] E. G. Birgin, J. M. Martínez and M. Raydan, Nonmonotone spectral projected gradient methods on convex sets, *SIAM Journal on Optimization* 10, pp. 1196–1211, 2000.

[8] E. G. Birgin, J. M. Martínez and M. Raydan, Algorithm 813: SPG - Software for convex-constrained optimization, *ACM Transactions on Mathematical Software* 27, pp. 340–349, 2001.

[9] E. G. Birgin, J. M. Martínez and M. Raydan. Inexact Spectral Projected Gradient methods on convex sets, *IMA Journal on Numerical Analysis* 23, pp. 539–559, 2003.

[10] I. Bongartz, A. R. Conn, N. I. M. Gould and Ph. L. Toint. CUTE: constrained and unconstrained testing environment. *ACM Transactions on Mathematical Software* 21, pp. 123–160 (1995).

[11] J. P. Boyle and R. L. Dykstra, A method for finding projections onto the intersection of convex sets in Hilbert spaces, *Lecture Notes in Statistics* 37, pp. 28–47, 1986.

[12] O. Burdakov, J. M. Martínez and E. A. Pilotta, A limited memory multipoint secant method for bound constrained optimization, *Annals of Operations Research* 117, pp. 51–70, 2002.

[13] P. H. Calamai and J. J. Moré, Projected gradient methods for linearly constrained problems, *Mathematical Programming* 39, pp. 93–116, 1987.

[14] A. R. Conn, N. I. M. Gould and Ph. L. Toint. *Trust Region Methods*, MPS SIAM Series on Optimization, Philadelphia, USA, 2000.

[15] Y. H. Dai, On nonmonotone line search, to appear in *Journal of Optimization Theory and Applications* .

[16] Y. H. Dai and L. Z. Liao, R-linear convergence of the Barzilai and Borwein gradient method, *IMA Journal on Numerical Analysis* 22, pp. 1–10, 2002.

[17] J. E. Dennis Jr. and J. J. Moré, A characterization of superlinear convergence and its application to quasi-Newton methods, *Mathematics of Computation* 28, pp. 549-560, 1974.

[18] J. E. Dennis Jr. and R. B. Schnabel *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, 1983.

[19] Z. Dostál, Box constrained quadratic programming with proportioning and projections, *SIAM Journal on Optimization* 7, pp. 871–887, 1997.

[20] R. L. Dykstra, An algorithm for restricted least-squares regression, *Journal of the American Statistical Association* 78, pp. 837–842, 1983.

[21] E. Eskow and R. B. Schnabel, Algorithm 695 – Software for a new modified Cholesky factorization, *ACM Transactions on Mathematical Software* 17, pp. 306–312, 1991.

[22] R. Fletcher, *Practical Methods of Optimization*, 2nd Edition, John Wiley and Sons Ltd, Reading, MA, USA, 1990.

[23] R. Fletcher, Low storage methods for unconstrained optimization, *Lectures in Applied Mathematics (AMS)* 26, pp. 165–179, 1990.

[24] R. Fletcher, On the Barzilai-Borwein method, *Department of Mathematics, University of Dundee* NA/207, Dundee, Scotland, 2001.

[25] A. Friedlander and J. M. Martínez, On the maximization of a concave quadratic function with box constraints, *SIAM Journal on Optimization* 4, pp. 177–192, 1994.

[26] A. Friedlander, J. M. Martínez and M. Raydan, A new method for large-scale box constrained convex quadratic minimization problems, *Optimization Methods and Software* 5, pp. 57–74, 1995.

[27] J. Gauvin, A necessary and sufficient condition to have bounded multipliers in nonconvex programming, *Mathematical Programming* 12, 136–138, 1977.

[28] P. E. Gill, W. Murray and M. H. Wright, *Practical Optimization*, Academic Press, NY, USA, 1981.

[29] A. A. Goldstein, Convex Programming in Hilbert Space, *Bulletin of the American Mathematical Society* 70, pp. 709–710, 1964.

[30] L. Grippo, F. Lampariello and S. Lucidi, A nonmonotone line search technique for Newton's method, *SIAM Journal on Numerical Analysis* 23, pp. 707–716, 1986.

[31] S. P. Han, A successive projection method, *Mathematical Programming* 40, pp. 1–14, 1988.

[32] E. S. Levitin and B. T. Polyak, Constrained Minimization Problems, *USSR Computational Mathematics and Mathematical Physics* 6, pp. 1–50, 1966.

[33] D. G. Luenberger, *Introduction to Linear and Nonlinear Programming*, Reading, Mass.: Addison-Wisely Publishing Company, 1973.

[34] B. A. Murtagh and M. A. Saunders, Large-scale linearly constrained optimization, *Mathematical Programming* 14, pp. 41–72, 1978.

[35] E. Panier and A. L. Tits. On combining feasibility, descent and superlinear convergence in inequality constrained optimization. *Mathematical Programming* 59, pp. 261–276, 1993.

[36] M. Raydan, On the Barzilai and Borwein choice of steplength for the gradient method, *IMA Journal of Numerical Analysis* 13, pp. 321–326, 1993.

[37] M. Raydan, The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem, *SIAM Journal on Optimization* 7, pp. 26–33, 1997.

[38] M. Raydan and P. Tarazaga, Primal and polar approach for computing the symmetric diagonally dominant projection, *Numerical Linear Algebra with Applications* 9, pp. 333–345, 2002.

[39] R. T. Rockafellar. Monotone operators and the proximal point algorithm, *SIAM Journal on Control and Optimization* 14, pp. 877–898, 1976.

[40] R. J. Vanderbei, An interior point code for quadratic programming, *Optimization Methods & Software* 12, pp. 451–484, 1999.

[41] R. J. Vanderbei and D. F. Shanno, An interior point algorithm for nonconvex nonlinear programming, *Computational Optimization and Applications* 13, pp. 231–252, 1999.

[42] J. L. Zhou and A. L. Tits. Nonmonotone line search for minimax problems. *Journal of Optimization Theory and Applications* 76, pp. 455–476, 1993.

[43] J. L. Zhou and A. Tits. User's Guide for FFSQP Version 3.7: A Fortran code for solving optimization programs, possibly minimax, with general inequality constraints and linear equality constraints, generating feasible iterates. Technical Report SRC-TR-92-107r5, Institute for Systems Research, University of Maryland, 1997.