On the solution to the subproblems of a globally convergent SQP algorithm for nonlinear programming *

F. A. M. Gomes †       S. A. Santos ‡       R. C. A. Thomé §

December 17, 2003

Abstract

This work introduces a bound-constrained-based strategy for dealing with the quadratic subproblems of the sequential quadratic programming (SQP) algorithm proposed by Gomes, Maciel and Martínez (1999). Two approaches for choosing the Hessian of the quadratic model are suggested. Numerical experiments are presented that illustrate the use of the proposed ideas.

Keywords: SQP algorithm, quadratic subproblems, bound-constrained minimization, augmented Lagrangian merit function, non-monotone penalty parameter, trust region.

AMS: 65K05, 90C55

---

*This work was supported by CAPES, CNPq, FINEP and FAEP-UNICAMP.

†Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas SP, Brazil. E-mail: chico@ime.unicamp.br

‡Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas SP, Brazil. E-mail: sandra@ime.unicamp.br

§Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas SP, Brazil. E-mail: rcthome@ime.unicamp.br
1 Introduction

Our aim is to solve nonlinear programming problems in the general form

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad h(x) = 0, \\
& \quad l \leq x \leq u,
\end{align*}
\]

where \( f : \mathbb{R}^n \to \mathbb{R} \) and \( h : \mathbb{R}^n \to \mathbb{R}^m \) are smooth functions (\( C^1 \) or \( C^2 \)) and \( m \leq n \). Components of vectors \( l \) and \( u \) might be \(-\infty\) or \(+\infty\).

There are basically three approaches for solving general problems like (1): (i) maintain feasibility, trying to deal with the nonlinear constraints (elimination and feasible direction methods), (ii) incorporate the constraints into the objective function (penalization methods), and (iii) linearize the feasible set and build a (quadratic) model to the objective function. It is in the latter category that the sequential quadratic programming (SQP) methods fit.

Extension of Newton’s method for constrained optimization, the SQP approach is considered one of the most effective methods for nonlinearly constrained optimization. The steps are generated by the solution of quadratic subproblems. It can be used within both line search and trust-region globalization frameworks, and it is appropriate for small or large problems. Recent researches point at a particular interest towards SQP methods, from a theoretical perspective [15, 21, 33, 39, 45], in the large-scale scenario [7, 8, 23, 27, 31, 41, 43] and in applications like control, dynamic systems, among others [16, 22, 28, 30, 32, 40, 44].

In [25], the authors propose a model algorithm based on the SQP method for solving (1). The strategy for obtaining global convergence is based on the trust-region approach, using an augmented Lagrangian merit function and a non monotone updating scheme for the penalty parameter.

Here we focus on the quadratic subproblems of the algorithm introduced in [25], that are recast as equivalent bound-constrained problems without any additional parameter. An essential ingredient to the equivalency is convexity, which guides the choices for the Hessian of the quadratic model. Two possibilities are addressed, namely a quasi-Newton approximation to the Hessian of the augmented Lagrangian function and a quasi-Newton reduced Hessian. Numerical experiments illustrate the performance of these choices for a small set of problems.

This paper is structured as follows: Section 2 contains the SQP algorithm, together with the used notation and assumptions. Section 3 presents the equivalence result that will turn the quadratic subproblems into bound-constrained ones. The subproblems of the SQP algorithm are defined in Section 4, together with specific details on finding the required directions or steps, mostly approximate solutions of
the stated subproblems. The choices for the Hessian of the quadratic model are detailed in Section 5. Section 6 contains the description of the numerical experiments, and their results. Conclusions and ideas for future research are presented in Section 7.

2 The SQP algorithm

For the reader’s convenience, in this section we recall the algorithm proposed in [25], herein denoted Algorithm GMM. First, some notation should be specified, together with the main ingredients used in the algorithm. We denote the derivatives by

\[ g(x) = \nabla f(x) \quad \text{and} \quad (h'(x))^T = (\nabla h_1(x) \cdots \nabla h_m(x)) \in \mathbb{R}^{n \times m}, \]

and, the original bound-constrained set by \( \Omega = \{ x \in \mathbb{R}^n \mid l \leq x \leq u \} \).

The merit function is

\[ \Phi(x, \lambda, \theta) = \theta \ell(x, \lambda) + (1 - \theta) \varphi(x), \quad (2) \]

where \( \theta \in [0, 1] \) is given, \( \lambda \in \mathbb{R}^m \), \( \ell(x, \lambda) \) is the Lagrangian function

\[ \ell(x, \lambda) = f(x) + h(x)^T \lambda \]

and

\[ \varphi(x) = \frac{1}{2} \| h(x) \|_2^2. \]

We say that \( z \in \Omega \) is \( \varphi \)-stationary if it satisfies the first order optimality conditions of

\[ \min \varphi(x) \quad \text{s.t.} \quad x \in \Omega. \]

A point \( z \) is feasible if \( z \in \Omega \) and \( h(z) = 0 \). A feasible point \( z \) is regular if the gradients of the active constraints at \( z \) are linearly independent.

The quadratic approximation to \( \ell(x + s, \lambda) \) is

\[ Q(s) \equiv Q(B, x, \lambda, s) = \frac{1}{2} s^T Bs + \nabla_x \ell(x, \lambda)^T s + \ell(x, \lambda), \quad (3) \]

where \( B \in \mathbb{R}^{m \times n} \) is a symmetric matrix.

The actual reduction and the predicted reduction of the merit function from \( (x, \lambda) \) to \( (x + s, \lambda + \Delta \lambda) \) are given, respectively, by

\[ A_{\text{red}}(x, \lambda, s, \Delta \lambda, \theta) = \theta[\ell(x, \lambda) - \ell(x + s, \lambda + \Delta \lambda)] + (1 - \theta)[\varphi(x) - \varphi(x + s)] \quad (4) \]
and

\[ \text{Pred}(x, \lambda, s, \Delta \lambda, \theta) = \theta \left[ Q(0) - Q(s) - [h'(x)s + h(x)]^T \Delta \lambda \right] + \\
(1 - \theta) \left[ \frac{1}{2} \|h(x)\|^2_2 - \frac{1}{2} \|h'(x)s + h(x)\|^2_2 \right]. \tag{5} \]

Given \( x_k \), an estimate of the solution to (1), \( \lambda_k \), an estimate of the Lagrange multipliers of (1) and a symmetric matrix \( B_k \), the algorithm that follows describes the steps to obtain the next iterate \( x_{k+1} \) and the next estimate of the vector of Lagrange multipliers \( \lambda_{k+1} \).

There are some parameters that should be commented. At the beginning of each iteration, a lower bound \( \Delta_{\text{min}} > 0 \) to the trust-region radius is used, so the iteration begins with \( \Delta \geq \Delta_{\text{min}} \). There are also upper bounds \( L_1 \) and \( L_2 \) to the norm of the Lagrange multiplier estimates and to the norm of the Hessian of the quadratic model, respectively. Finally, the value \( N > 0 \) defines the non-monotonicity of the penalty parameter.

### 2.1 Algorithm GMM

Let \( x_k \in \Omega, \lambda_k \in \mathbb{R}^m, \|\lambda_k\| \leq L_1, \Delta \geq \Delta_{\text{min}}, B_k = B_k^T \in \mathbb{R}^{m \times n}, \|B_k\| \leq L_2 \) and \( N \geq 0 \). If \( x_k \) is a stationary point of (1), \( x_k \) is \( \varphi \)-stationary but not feasible, or if \( x_k \) is feasible but not regular, then the algorithm stops. Otherwise, the steps for obtaining the next iterate are the following:

**Step 0.** Compute a feasible descent direction for the constraints.

If \( x_k \) is not \( \varphi \)-stationary, compute \( d_{\text{nor}} \in \mathbb{R}^n \) such that

\[ l \leq x_k + d_{\text{nor}}(x_k) \leq u, \tag{6} \]

and

\[ d_{\text{nor}}(x_k)^T \nabla \varphi(x_k) < 0. \tag{7} \]

If \( x_k \) is feasible, regular and non-stationary for problem (1), then define \( d_{\text{nor}}(x_k) = 0 \) and \( s_{\text{nor}} = s_{\text{nor}}(x_k, \Delta) = 0. \)

**Step 1.** Compute the decrease step for the constraints.

If \( x_k \) is not feasible, compute \( s_{\text{nor}} = s_{\text{nor}}(x_k, \Delta) \in \mathbb{R}^n \) by solving

\[
\begin{align*}
\text{min} & \quad \frac{1}{2} \|h'(x_k)s + h(x_k)\|_2^2 \\
\text{s.t.} & \quad \|s\| \leq 0.8 \Delta \\
& \quad l \leq x_k + s \leq u \\
& \quad s = td_{\text{nor}}(x_k), \quad t \geq 0.
\end{align*}
\]

**Step 2.** Compute the normal step.
If $x_k$ is not feasible, compute $s_{\text{nor}} = s_{\text{nor}}(x_k, \Delta) \in \mathbb{R}^n$ such that
\[
l \leq x_k + s_{\text{nor}} \leq u, \quad ||s_{\text{nor}}|| \leq 0.8\Delta
\]
and
\[
||h(x_k)||^2_2 - ||h'(x_k)s_{\text{nor}} + h(x_k)||^2_2 \geq 0.9 \left( ||h(x_k)||^2_2 - ||h'(x_k)s_{\text{nor}} + h(x_k)||^2_2 \right). \quad (8)
\]

**Step 3.** Compute the decrease step for the Lagrangian function on the tangent space.

If $x_k$ is feasible, set $d_{\text{tan}} = d_{\text{tan}}(B_k, x_k, \lambda_k, \Delta) = d_{\text{tan}}(x_k)$. Otherwise, compute $d_{\text{tan}}(x_k) = d_{\text{tan}}(B_k, x_k, \lambda_k, \Delta) \in \mathbb{R}^n$ such that
\[
l \leq x_k + s_{\text{nor}} + d_{\text{tan}}(x_k) \leq u, \quad (9)
\]
\[
h'(x_k)d_{\text{tan}}(x_k) = 0 \quad (10)
\]
and
\[
d_{\text{tan}}(x_k)^T \nabla Q(B_k, x_k, \lambda_k, s_{\text{nor}}) < 0. \quad (11)
\]

If a vector satisfying (9)-(11) does not exist, define $d_{\text{tan}} = 0$. Otherwise (which includes the case $s_{\text{nor}} = 0$, i.e. $x_k$ feasible), define $s_{\text{dec}} = s_{\text{dec}}(B_k, x_k, \lambda_k, \Delta)$ the solution of
\[
\begin{align*}
\min & \quad Q_k(s_{\text{nor}} + s) \\
\text{s.t.} & \quad ||s_{\text{nor}} + s|| \leq \Delta \\
& \quad l \leq x_k + s_{\text{nor}} + s \leq u \\
& \quad s = td_{\text{tan}}(x_k), \quad t \geq 0,
\end{align*}
\]
where $Q_k(s) \equiv Q(B_k, x_k, \lambda_k, s)$.

**Step 4.** Compute the tangent step.

Compute $s_{\text{tan}} = s_{\text{tan}}(B_k, x_k, \lambda_k, \Delta) \in \mathbb{R}^n$ such that
\[
h'(x_k)s_{\text{tan}} = 0; \quad l \leq x_k + s_{\text{nor}} + s_{\text{tan}} \leq u; \quad ||s_{\text{nor}} + s_{\text{tan}}|| \leq \Delta \quad (12)
\]
and
\[
Q_k(s_{\text{nor}}) - Q_k(s_{\text{nor}} + s_{\text{tan}}) \geq 0.9[Q_k(s_{\text{nor}}) - Q_k(s_{\text{nor}} + s_{\text{dec}})]. \quad (13)
\]

**Step 5.** Compute the update of the Lagrange multipliers and the current step.

Compute $\Delta\lambda \in \mathbb{R}^m$ such that $||\lambda_k + \Delta\lambda|| \leq L_1$ and define
\[
s_c = s_c(B_k, x_k, \lambda_k, \Delta) = s_{\text{nor}} + s_{\text{tan}}.
\]

**Step 6.** Compute the penalty parameter $\theta \in [0, 1]$.

Compute $\theta_k^{\text{min}}$, $\theta_k^{\text{large}}$, $\theta_k^{\text{sup}}$ such that
if \( k = 0 \) then \( \theta_k^{\min} = 1 \), otherwise \( \theta_k^{\min} = \min\{1, \theta_0, \cdots, \theta_{k-1}\} \);

\[ \theta_k^{\text{large}} = \left[1 + \frac{N}{(k+1)^\gamma}\right] \theta_k^{\min}; \]

\[ \theta_k^{\sup} = \min\left\{1, \frac{1}{2}[Q_k(s_c) - Q_k(0) + |h'(x_k)s_c + h(x_k)|^T \Delta \lambda + \frac{1}{2}||h'(x_k)||^2 - \frac{1}{2}||h'(x_k)s_c + h(x_k)||^2] \right\}. \]

If \( \Delta \) is the first trust-region radius tested at the current iteration, define

\[ \theta' = \theta'(x_k, \Delta) = \theta_k^{\text{large}}. \]

Otherwise, set

\[ \theta' = \theta'(x_k, \Delta) = \theta(x_k, \Delta'), \]

where \( \Delta' \) is the trust-region radius tested immediately before \( \Delta \) at the current iteration.

Set \( \theta = \min\{\theta_k^{\sup}, \theta'\} \).

**Step 7.** Accept or reject the current step \( s_c \).

If

\[ \text{Ared}(x_k, \lambda_k, s_c, \Delta \lambda, \theta) \geq 0.1 \text{Pred}(x_k, \lambda_k, s_c, \Delta \lambda, \theta) \quad \text{(14)} \]

define

\[ x_{k+1} = x_k + s_c, \quad \lambda_{k+1} = \lambda_k + \Delta \lambda, \quad \theta_k = \theta. \]

Otherwise, choose new \( \Delta \in [0.1 \Delta, 0.9 \Delta] \) and go to step 1.

It was proved in [25] that the Algorithm GMM is well defined under conditions (7), (8) and (11). In other words, if it does not terminate at the current point, the next iterate satisfying (14) will be found after repeating steps 1-7 a finite number of times.

As far as global convergence is concerned, assuming continuity of the direction \( d_{nor} \) at step 0 with respect to the current point, bounded variation of the first derivatives, and compactness of the generated sequence, it was proved in [25] that every limit point of Algorithm GMM is \( \varphi \)-stationary. Furthermore, under two suitable algorithmic assumptions, the authors have proved that there exists a limit point of Algorithm GMM that is a stationary point of (1).

### 3 An equivalence result

There are many efficient algorithms available to the solution of bound-constrained minimization problems, like trust-region methods [12, 13, 18] and spectral-projected-gradient-based methods [6], among others.
Moreover, an approach for minimizing linearly constrained convex problems by means of an equivalent reformulation was proposed in [17]. The equivalent problem is nonconvex, bound-constrained, and under mild assumptions, its stationary points are global minimizers. This equivalent approach has given rise to several reformulations in different contexts of complementarity, variational inequalities, mathematical problems with equilibrium constraints, etc. (see [1, 2, 3, 4, 19, 20]).

Here we apply the aforementioned equivalence result to the quadratic subproblems of the Algorithm GMM. To this purpose, let us consider the general quadratic programming problem

\[
\min \frac{1}{2} x^T B x + g^T x \\
\text{s.t.} \quad A x = c, \quad \bar{l} \leq x \leq \bar{u},
\]

where \( B = B^T \in \mathbb{R}^{n \times n} \), \( A \in \mathbb{R}^{m \times n}, g, x, \bar{l}, \bar{u} \in \mathbb{R}^n \), \( c \in \mathbb{R}^m \), and define the set \( \mathcal{W} = \{ x \in \mathbb{R}^n \mid A x = c, \bar{l} \leq x \leq \bar{u} \} \). The first-order optimality conditions of (15) motivate the definition of problem

\[
\min \Lambda(x, v, z, w) \\
\text{s.t.} \quad \bar{l} \leq x \leq \bar{u}, \quad v \in \mathbb{R}^m, \quad z \geq 0, \quad w \geq 0
\]

where

\[
\Lambda(x, y, z, w) = \frac{1}{2} \left( \| B x + g + A^T v - z + w \|^2 + \| A x - c \|^2 \right) \\
\quad + \left( (x - \bar{l})^T z \right)^2 + \left( (\bar{u} - x)^T w \right)^2.
\]

The objective function of problem (16) is not convex and typical algorithms for bound-constrained minimization are convergent to stationary points, not necessarily global minimizers. However, under the hypothesis that (15) is convex, the result stated below guarantees that finding stationary points of (16) is equivalent to solving (15).

**Theorem 1** If the feasible set \( \mathcal{W} \) is nonempty and bounded, and the Hessian matrix \( B \) is positive semidefinite then problem (16) admits at least one stationary point and every stationary point \((x^*, v^*, z^*, w^*)\) of (16) is such that \( \Lambda(x^*, v^*, z^*, w^*) = 0 \).

**Proof:** It is a slight adaptation of the proof of Theorem 2.1 of [17] (cf. [42]).

It is worth mentioning that the hypothesis of boundedness of \( \mathcal{W} \) can be removed due to the quadratic nature of the objective function of problem (15). This follows from Theorem 2 of [3].
4 Subproblems of the Algorithm GMM

In this section we address the quadratic subproblems of Algorithm GMM by means of the equivalence result stated in Theorem 1.

At step 0, to compute \( d_{\text{nor}} \) satisfying (6)-(7), it is enough to set \( d_{\text{nor}} = y^* - x_k \), where \( y^* \) is the solution to

\[
\min \frac{1}{2} \| x_k - \alpha \nabla \varphi(x_k) - y \|_2^2 \\
\text{s.t.} \quad l \leq y \leq u,
\]

with \( \alpha > 0 \) a fixed parameter. Problem (17) is quite simple, and its solution can be easily computed by projecting \( x_k - \alpha \nabla \varphi(x_k) \) on the box \( \{ y \in \mathbb{R}^n \mid l \leq y \leq u \} \).

At step 2, to compute \( s_{\text{nor}} \) satisfying (8), we set \( s_{\text{nor}} = \tilde{s} \), where \( \tilde{s} \) is an approximate solution to

\[
\min \frac{1}{2} \| h'(x_k)s + h(x_k) \|_2^2 \\
\text{s.t.} \quad \| s \|_\infty \leq 0.8 \Delta \\
\quad l \leq x_k + s \leq u
\]

that verifies inequality (8). Problem (18) consists of minimizing a bound-constrained quadratic and any algorithm that computes an approximate solution to this problem can be used (e.g. [5, 6, 18]).

At step 3, to obtain \( d_{\text{tan}} \) such that conditions (9)-(11) hold we set \( d_{\text{tan}} = \tilde{d} \), where \( \tilde{d} \) is the solution to

\[
\min \frac{1}{2} \| \eta \nabla Q_k(s_{\text{nor}}) + d \|_2^2 \\
\text{s.t.} \quad h'(x_k)d = 0 \\
\quad l \leq x_k + s_{\text{nor}} + d \leq u,
\]

with fixed \( \eta > 0 \). Since the objective function of problem (19) is a convex quadratic and \( 0 \in \mathbb{R}^n \) is feasible for this problem, its solution \( \tilde{d} \) will be found by solving the following equivalent problem, as guaranteed by Theorem 1,

\[
\min \quad \bar{\Lambda}(d, v, z, w) \\
\text{s.t.} \quad l \leq x_k + s_{\text{nor}} + d \leq u, \quad v \in \mathbb{R}^m, \quad z \geq 0, \quad w \geq 0,
\]

where

\[
\bar{\Lambda}(d, v, z, w) = \frac{1}{2} \left( \| d + \eta \nabla Q_k(s_{\text{nor}}) + h'(x_k)^T v - z + w \|_2^2 + \| h'(x_k)d \|_2^2 \\
+ \left( (d + x_k + s_{\text{nor}} - l)^T z \right)^2 + \left( (u - d - x_k - s_{\text{nor}})^T w \right)^2 \right).
\]
Finally, let us analyze step 4. To obtain \(s_{tan}\) satisfying (12)-(13), we consider the approximate solution to problem

\[
\min Q_k(s) \\
\text{s.t.} \\
h'(x_k)s = 0 \\
\|s_{nor} + s\|_\infty \leq \Delta \\
l \leq x_k + s_{nor} + s \leq u, 
\]

(21)

that verifies (13). The feasible set of problem (21) is nonempty (0 \(\in\) \(\mathbb{R}^n\) is feasible) and bounded. Moreover, \(\|s_{nor} + s\|_\infty \leq \Delta\) and \(l \leq x_k + s_{nor} + s \leq u\) can be rewritten as \(\tilde{l} \leq s \leq \tilde{u}\), where the components of \(\tilde{l}\) and \(\tilde{u}\) are respectively

\[
\tilde{l}_i = \max\{-\Delta - (s_{nor})_i, \, l_i - (x_k)_i - (s_{nor})_i\} 
\]

(22)

and

\[
\tilde{u}_i = \min\{\Delta - (s_{nor})_i, \, u_i - (x_k)_i - (s_{nor})_i\}, 
\]

(23)

where \(i = 1, \ldots, n\). The equivalent formulation is

\[
\min \tilde{\Lambda}(s, v, z, w) \\
\text{s.t.} \quad \tilde{l} \leq s \leq \tilde{u}, \quad v \in \mathbb{R}^m, \quad z \geq 0, \quad w \geq 0, 
\]

(24)

where

\[
\tilde{\Lambda}(s, v, z, w) = \frac{1}{2} \left( \|B_k s + \nabla_x f(x_k, \lambda_k) + h'(x_k)^T v - z + w\|_2^2 + \|h'(x_k)s\|_2^2 \right) \\
+ \left( (s - \tilde{l})^T z \right)^2 + \left( (\tilde{u} - s)^T w \right)^2. 
\]

In order to apply the result established by Theorem 1, it is necessary that matrix \(B_k\) be positive semidefinite. This requirement will be addressed in detail in the next section.

5 Choices for the Hessian of the quadratic model

Let us consider first, for simplicity, the equality-constrained optimization problem

\[
\min f(x) \\
\text{s.t.} \quad h(x) = 0. 
\]

(25)

The equivalence between SQP and Newton’s method applied to the optimality conditions of (25) is based on the choice of matrix \(B_k\) as the Hessian of the Lagrangian, that is,

\[
\nabla_{xx}^2 f(x_k, \lambda_k) = \nabla_{xx}^2 f(x_k) + \sum_{i=1}^m [\lambda_k]_i \nabla_{xx}^2 h_i(x_k). 
\]

(26)
In spite of the good local properties of this choice, that often produces fast progress when the iterates are far from the solution (see e.g. [26, 35]), the reasons for not adopting \( B_k = \nabla^2_x \ell(x_k, \lambda_k) \) are twofold. First, the Hessian of the Lagrangian is made up of second derivatives of the objective function and constraints, which may not be easy to compute. Second, in general, this matrix may not be positive definite along the generated sequences \( \{x_k\}, \{\lambda_k\} \), not even on the constraint null space. Therefore, alternative choices for \( B_k \) should be devised.

### 5.1 Quasi-Newton approximations to the Hessian of the augmented Lagrangian

Although model (3) is a quadratic approximation to the Lagrangian, it might be that \( B_k = \nabla^2_x \ell(x_k, \lambda_k) \) has negative eigenvalues and, thus, model (3) with this choice will not be convex. To overcome this difficulty and, at the same time, take advantage of the related second-order optimality condition, we shall consider the Hessian of the merit function (2):

\[
\nabla^2_{xx} \Phi(x, \lambda, \theta) = \theta \nabla^2_{xx} \ell(x, \lambda) + (1 - \theta) \left( \sum_{i=1}^{m} h_i(x) \nabla^2_{xx} h_i(x) + h'(x)^T h'(x) \right). \tag{27}
\]

Using the change of variables \( \bar{\rho} = (1 - \theta)/\theta \) for the penalty parameter, the merit function (2) can be seen as the usual augmented Lagrangian. The classic theory of augmented Lagrangian ensures that, at a minimizer \( (x_*, \lambda_*) \) of (1) satisfying second-order sufficiency conditions, matrix (27) becomes

\[
\nabla^2_{xx} \Phi(x_*, \lambda_*, \theta) = \theta \nabla^2_{xx} \ell(x_*, \lambda_*) + (1 - \theta) h'(x_*)^T h'(x_*), \tag{28}
\]

which is positive definite for \( \theta \in (0, \theta^*) \) (see, e.g. [34, chap. 13]). Therefore, we could use directly (27) as the Hessian of the quadratic model (3), or a quasi-Newton approximation to it.

A popular quasi-Newton formula is the BFGS one, for which the update is given by

\[
B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k}, \tag{29}
\]

where \( s_k = x_{k+1} - x_k \) and \( y_k = \nabla_x \Phi(x_{k+1}, \lambda_{k+1}, \theta) - \nabla_x \Phi(x_k, \lambda_{k+1}, \theta) \) (see [37, p. 542]).

If \( B_k \) is positive definite, as in the unconstrained case, the curvature condition \( s_k^T y_k > 0 \) ensures that \( B_{k+1} \) is positive definite as well. However, it might be that \( s_k \)
and $y_k$ do not satisfy this condition. An alternative, so-called damped BFGS, was proposed by Powell [38] and consists of replacing $y_k$ by

$$
\bar{y}_k = \nu y_k + (1 - \nu)B_k s_k,
$$

(30)

where

$$
\nu = \left\{ \begin{array}{ll}
1, & s_k^T y_k \geq 0.2 s_k^T B_k s_k \\
0.8 s_k^T B_k s_k, & s_k^T y_k < 0.2 s_k^T B_k s_k.
\end{array} \right.
$$

(31)

It is worth mentioning that the choice (27), or a quasi-Newton approximation to it, are not free from difficulties. The threshold value $\theta^*$ for which matrix (28) is positive definite depends on bounds on the second derivatives of the problem, usually not known. A too small choice of $\theta$ might result in the domination of the last term of (28), whereas if $\theta$ is too large, the Hessian of the augmented Lagrangian may not be positive definite. As a consequence, the curvature condition at the corresponding quasi-Newton approximation $B_k$ may not be satisfied.

A variant to Powell’s approach (30)-(31) is suggested by Nocedal and Wright [37, p. 542], and comes from the fact that near the solution there is a maximum value for $\theta$ that guarantees uniform positiveness of $s_k^T y_k$. It rests upon the relationship

$$
y_k = \nabla_x \Phi(x_{k+1}, \lambda_{k+1}, \theta) - \nabla_x \Phi(x_k, \lambda_{k+1}, \theta)
= \theta y_k^\ell + (1 - \theta) h'(x_{k+1})^T h(x_{k+1}),
$$

where $y_k^\ell = \nabla_x \ell(x_{k+1}, \lambda_{k+1}) - \nabla_x \ell(x_k, \lambda_k)$, as long as the first-order update for the Lagrange multipliers is used $\theta \lambda_{k+1} = \theta \lambda_k + (1 - \theta) h(x_k)$. Thus, the idea is to select $\theta$ adaptively to satisfy a positivity criterion upon the curvature condition whenever $y_k$ is used in the update (29).

### 5.2 Quasi-Newton reduced-Hessian approximations

In the previous approach we adopted full matrices $B_k \in \mathbb{R}^{n \times n}$. Now we will consider approximations to the reduced-Hessian of the Lagrangian, namely,

$$
Z_k^T \nabla_{xx}^2 \ell(x_k, \lambda_k) Z_k,
$$

(32)

where the columns of matrix $Z_k \in \mathbb{R}^{n \times p}$ span the null space of $h'(x_k)$). Procedures for computing a smoothly varying sequence of null-space matrices $Z_k$ are described in [10, 24]. Convergence results for reduced-Hessian quasi-Newton methods have been proved in [9].
The Hessian (32) has smaller dimension than (27) and, under the standard assumptions, is positive definite in a neighborhood of the solution.

To use matrix (32) in problem (21) we will proceed as follows. Assuming that \( h'(x_k) \) has full rank \( m \), and that matrix \( Z_k \in \mathbb{R}^{n \times (n - m)} \) has been determined, the change of variables \( s = Z_k v, v \in \mathbb{R}^{n-m} \) turns (21) into

\[
\begin{align*}
\min & \quad \frac{1}{2} v^T Z_k^T \nabla^2 \ell(x_k, \lambda_k) Z_k v + v^T Z_k^T \nabla_x \ell(x_k, \lambda_k) + \ell(x_k, \lambda_k) \\
\text{s.t.} & \quad \|s_{nor} + Z_k v\|_\infty \leq \Delta \\
& \quad l \leq x_k + s_{nor} + Z_k v \leq u.
\end{align*}
\]

(33)

The feasible set of (33) can be rewritten as

\[
\tilde{l} \leq Z_k v \leq \tilde{u},
\]

(34)

where the components of the bounds \( \tilde{l} \) and \( \tilde{u} \) are given, respectively, by (22) and (23), so that we have a quadratic problem with linear inequality constraints. Adding slacks to (34), it becomes

\[
\begin{align*}
\begin{cases}
Z_k v \leq \tilde{u} \\
Z_k v \geq \tilde{l}
\end{cases} \implies \begin{cases}
Z_k v + w = \tilde{u} \\
0 \leq w \leq \tilde{u} - \tilde{l},
\end{cases}
\end{align*}
\]

where \( w \in \mathbb{R}^n \). Thus, (33) can be rewritten as

\[
\begin{align*}
\min & \quad \frac{1}{2} v^T Z_k^T \nabla^2 \ell(x_k, \lambda_k) Z_k v + v^T Z_k^T \nabla_x \ell(x_k, \lambda_k) + \ell(x_k, \lambda_k) \\
\text{s.t.} & \quad (Z_k \ I) \begin{pmatrix} v \\ w \end{pmatrix} = \tilde{u} \\
& \quad 0 \leq w \leq \tilde{u} - \tilde{l}.
\end{align*}
\]

(35)

Problem (35) has \( 2n - m \) variables, \( n \) linear equality constraints and \( n \) simple bounds. Besides, it is convex in a neighborhood of a minimizer, its feasible set is nonempty (observe that \( Z_k v = 0 \) is feasible to (33) and \( \tilde{l} \leq 0 \leq \tilde{u} \leq \tilde{u} - \tilde{l} \) and, by the equivalence between (33) and (35), the feasible set of (35) is also bounded. Therefore, Theorem 1 can be applied to (35), and the equivalent formulation is given by

\[
\begin{align*}
\min & \quad \tilde{\lambda}_R(v, w, \tilde{z}, \tilde{w}) \\
\text{s.t.} & \quad v \in \mathbb{R}^{n-m}, \ 0 \leq w \leq \tilde{u} - \tilde{l}, \ \tilde{z} \geq 0, \ \tilde{w} \geq 0,
\end{align*}
\]

(36)

where

\[
\tilde{\lambda}_R(v, w, \tilde{z}, \tilde{w}) = \frac{1}{2} \left( \|Z_k^T \nabla^2 \ell(x_k, \lambda_k) Z_k v + Z_k^T \nabla_x \ell(x_k, \lambda_k) + Z_k^T (\tilde{z} - \tilde{w})\|^2 \right.
\]

\[
\left. + \|Z_k v + w - \tilde{u}\|^2 + \left( w^T \tilde{z} \right)^2 + \left( (\tilde{u} - \tilde{l} - w)^T \tilde{w} \right)^2 \right)
\]
Next we provide the expressions to approximate the Hessian (32) in the quasi-Newton context. Denoting the approximation matrix by $M_k = Z_k^T \nabla^2 \ell(x_k, \lambda_k) Z_k$ and following [37, p.553], its BFGS update is given by

$$M_{k+1} = M_k - \frac{M_k s_k s_k^T M_k}{s_k^T M_k s_k} + \frac{\hat{y}_k \hat{y}_k^T}{s_k^T \hat{y}_k},$$

(37)

with $s_k = v_k^*$, where vector $v_k^*$ is the solution to (33), and

$$\hat{y}_k = Z_k^T (\nabla_x \ell(x_{k+1}, \lambda_{k+1}) - \nabla_x \ell(x_k, \lambda_{k+1})).$$

(38)

There are other possibilities for defining vector $\hat{y}_k$ with similar properties, such as

$$\hat{y}_k = Z_k^T (\nabla f(x_{k+1}) - \nabla f(x_k)),$$

(39)

or

$$\hat{y}_k = Z_k^T \left( \nabla f(x_k + Z_k v_k^*) + h'(x_k)^T \lambda_k - \nabla f(x_k) \right).$$

(40)

Choice (40) was inspired in [11] and consists of a projected updating scheme along the tangent space of the constraints, whenever used together with the least-squares update for the Lagrange multiplier vector, namely

$$\lambda_{k+1} = \left( h'(x_{k+1}) h'(x_{k+1})^T \right)^{-1} h'(x_{k+1}) \nabla f(x_{k+1}).$$

(41)

As before, in case the curvature condition $s_k^T \hat{y}_k > 0$ does not hold, some safeguard should be adopted, like the damped BFGS update. A common practical strategy is to skip the update, that is, set $M_{k+1} = M_k$ whenever $s_k^T \hat{y}_k \leq 0$.

6 Numerical experiments

In this section, we present some preliminary numerical results for the reformulation of the GMM algorithm proposed above. Some implementation details deliberately suppressed when the algorithm was presented are listed below.

- **Quacan**, a subroutine for solving bound-constrained quadratic problems (see [5]), is used to compute $s_{nor}$ as in (18).

- In our numerical tests, $d_{tan}$, the decrease step for the Lagrangian function on the tangent space, was never actually computed. The tangent step $s_{tan}$ is obtained by solving problem (36) by means of a bound-constrained nonconvex minimization algorithm based on [18]. This algorithm, developed at the University of Campinas, is available as a subroutine called **Box**. As the initial values for problem (36) we use $(v_0, w_0, z_0, w_0) = (0, \bar{u} - \bar{l}, 0, 0)$. 

13
At each iteration of the Box algorithm, a quadratic approximation of the objective function is minimized. The approximate solution of the resulting bound-constrained nonconvex quadratic problem is given by Quacan.

Instead of using \( s_{tan} = Z_kv \), we define \( s_{tan} = \bar{u} - w \), due to the second term from the objective function of problem (36).

One of the most time consuming steps of the new algorithm is the computation of matrix \( Z_k \). A reasonable alternative for generating this matrix, based on the LU-factorization, is described below.

Denoting \( A = h'(x_k) \) and writing \( A \) in the form \( A = [B 
 N] \), where \( B \) is invertible, \( Z \) is given by

\[
Z = \begin{bmatrix}
-B^{-1}N \\
I
\end{bmatrix}.
\]

Since it is not trivial to select the columns of \( A \) that will form matrix \( B \) and the computation of \( B^{-1} \) is impractical, we use the umfpack [14] routine to compute an LU factorization of \( A \), with row and column permutations. Therefore, writing \( PAP^T = LU \), we have

\[
Z = \bar{P}^T \begin{bmatrix}
-U_1^{-1}U_2 \\
I
\end{bmatrix}.
\]

To compute vector \( \hat{y}_k \), the following reformulation of (40) is used:

\[
\hat{y}_k = Z_k^T (\nabla f(x_k + Z_kv^*_k) + \nabla \ell(x_k) - 2\nabla f(x_k)).
\]

Instead of using (41), we use

\[
\lambda_{k+1} = (h'(x_{k+1})(I - I_\ell)h'(x_{k+1})^T)^{-1}h'(x_{k+1})\nabla f(x_{k+1}),
\]

where \( I_\ell \) is the diagonal matrix with \( I_{ii} = 1 \) if \( [x_{k+1}]_i = \ell_i \) or \( [x_{k+1}]_i = u_i \), and \( I_{ii} = 0 \) otherwise.

To analyze the performance of the new algorithm, a set of 22 very small problems extracted from the Hock and Schittkowski test library [29] was used. A FORTRAN implementation of the GST algorithm was build to compare it to the original GMM method. The results are presented in table 1.

Table 1 shows the total number of outer iterations required by the GST algorithm, and the number of Box and Quacan iterations used to compute the tangent
steps. For the GMM code, the number of outer iterations and the number of Minos [36] iterations required to compute $s_{\text{tan}}$ are given. The symbol (*) indicates that the algorithm failed to attain the correct value for the objective function. The number of variables and constraints are provided in columns $n$ and $m$, respectively.

The time spent by both algorithms was omitted here due to the fact that the GMM algorithm uses true Hessians, whereas the GST algorithm approximates the Hessians using a BFGS formula, so the values would not be comparable.

It can be noticed from the table that the number of outer iterations spent by the GST algorithm is not unreasonably greater than the GMM figures, especially if we take in account that it uses a quasi-Newton Hessian approximation. On the other hand, the Box routine usually takes too much iterations to solve the tangent
problem. Besides, Quacan also has to struggle to compute each step of the Box algorithm.

This bad behavior of the GST algorithm is related to the presence of the complementarity terms \((w^T \bar{z})^2\) and \(((\bar{u} - \bar{l} - w)^T \bar{w})^2\) in the objective function of problem (36).

7 Final remarks

The results presented in the last section suggest that, although theoretically convergent, the new algorithm cannot rival the performance of the GMM algorithm.

We also implemented the quasi-Newton approximation to the Hessian of the augmented Lagrangian (depicted in section 5.1) and several different strategies for computing \(\Delta \lambda\) and \(\hat{y}_k\), but the results were worse than those shown in table 1. Therefore, the performance of the original GMM algorithm seems to be unbeatable by an algorithm that uses a bound constrained method to solve the tangent subproblem.

Acknowledgments. We are indebted to R. Andreani and J. M. Martínez for their helpful comments.

References


[38] Powell, M. J. D. - The convergence of variable metric methods for nonlinearly
constrained optimization calculations, in *Nonlinear Programming 3* (Mangasarian,

[39] Qi, L. & Wei, Z. - On the Constant Positive Linear Dependence Condition and
pp.963-981.

[40] Sachs, E. W. - Optimization Methods in Food Processing, *Presentation at 16th
International Symposium on Mathematical Programming*, Lausanne, Switzerland,

[41] Sargent, R. W. H. & Ding, M. - A New SQP Algorithm for Large-Scale Non-

[42] Thomé, R. C. A. - Construção e análise de um algoritmo PQS globalmente
convergente. Master Thesis, IMECC - Unicamp, State University of Campinas,
Campinas (SP) Brazil, March 2001.

[43] Tomlin, J. - Implementation of a Filter Algorithm for Regularized SQP Solu-
tion of Nonlinear Programming, *Mini-Symposium at sixth SIAM Conference on

of Semilinear Parabolic Equations. *SIAM Journal on Control and Optimization*,

[45] Yuan, Y. & Chen, X. - Optimality Conditions for the CDT Subproblem, in *Nu-
merical Linear Algebra and Optimization*, Yuan, Y. ed, Science Press, Beijing,