A primal-dual interior-point algorithm for nonlinear least squares constrained problems

M. Fernanda P. Costa
Departamento de Matemática para a Ciência e Tecnologia
Universidade do Minho, 4810 Guimarães, Portugal
email: mfc@mct.uminho.pt

Edite M. G. P. Fernandes
Departamento de Produção e Sistemas
Universidade do Minho, 4710-057 Braga, Portugal
email: emgpf@dps.uminho.pt

Abstract

This paper extends prior work by the authors on solving nonlinear least squares unconstrained problems using a factorized quasi-Newton technique. With this aim we use a primal-dual interior-point algorithm for nonconvex nonlinear programming. The factorized quasi-Newton technique is now applied to the Hessian of the Lagrangian function for the transformed problem which is based on a logarithmic barrier formulation. We emphasize the importance of establishing and maintaining symmetric quasi-definiteness of the reduced KKT system. The algorithm then tries to choose a step size that reduces a merit function, and to select a penalty parameter that ensures descent directions along the iterative process. Computational results are included for a variety of least squares constrained problems and preliminary numerical testing indicates that the algorithm is robust and efficient in practice.

Key Words: Least squares, factorized quasi-Newton methods, primal-dual interior-point method

AMS subject classification: 90C30, 49M37

1   Introduction

In nonlinear least squares unconstrained problems, the objective function has the following special form

$$F(x) = \frac{1}{2} \sum_{j=1}^{m} (f_j(x))^2, \qquad (1.1)$$

where $f_j : \mathbb{R}^n \to \mathbb{R}$ for $j = 1, \ldots, m$ are twice continuously differentiable functions and $m \geq n$. The mathematical formulation (1.1) includes nonlinear data fitting and parameter estimation problems. Researchers in numerical analysis and optimization have been able to devise efficient and robust minimization algorithms by exploiting the special structure in F and its derivatives. Assembling the individual components f_j of F into a *residual vector* $f : \mathbb{R}^n \to \mathbb{R}^m$ defined by $f(x)= (f_1(x),\, f_2(x),\, \ldots,\, f_m(x))^T$, we can see why the special form of F often makes the least squares problem easier to solve than general unconstrained minimization problems. The derivatives of F can be expressed in terms of the Jacobian of f, which is the $m \times n$ matrix of first partial derivatives defined by

$$J(x) = [\partial f_j/\partial x_i]_{j=1,2,\ldots,m,\ i=1,2,\ldots,n}. \tag{1.2}$$

We have then

$$\nabla F(x) = \sum_{j=1}^{m} f_j(x) \nabla f_j(x) = J(x)^T f(x), \tag{1.3}$$

$$\begin{aligned}\nabla^2 F(x) &= \sum_{j=1}^{m} \nabla f_j(x) \nabla f_j(x)^T + \sum_{j=1}^{m} f_j(x) \nabla^2 f_j(x) \\ &= J(x)^T J(x) + \Sigma(x). \end{aligned} \tag{1.4}$$

In many applications, it is possible to calculate the first partial derivatives that make up the Jacobian matrix $J(x)$ explicitly. This could be used to calculate the gradient $\nabla F(x)$ as written in formula (1.3). However, the distinctive feature of least squares problems is that by knowing the Jacobian we can compute the first part of the Hessian $\nabla^2 F(x)$ for free. Moreover, the term $J(x)^T J(x)$ is often more important than the second summation term in (1.4), either because of near-linearity of the model near the solution or because of small residuals. Most algorithms for nonlinear least squares exploit these structural properties of the Hessian. They are based on Newton and quasi-Newton approaches. Two well known variants include the Gauss-Newton method (Dennis and Schnabel (1983), Kelly (1999)

and Wolfe (1978)) and the Levenberg-Marquardt method (Dennis and Schnabel (1983) and Moré (1978)). Since these methods neglect the second part of the Hessian matrix, the local convergence analysis establishes conditions for a q-quadratic convergence for linear and zero residual problems (Corollary 10.2.2 and Theorem 10.2.6 in Dennis and Schnabel (1983)). For the other problems, the convergence becomes q-linear (Theorem 10.2.1 of Dennis and Schnabel (1983)). Another class is the structured quasi-Newton methods. Since the term $J(x)^T J(x)$ of $\nabla^2 F(x)$ is available, only the second part of the Hessian matrix is approximated by the matrix R_k $(J(x_k)^T J(x_k) + R_k)$, which is updated by quasi-Newton formulae. Some references in this domain are Dennis et al. (1981), Gill and Murray (1976), McKeown (1972) and Nazareth (1980). Since the updating formulae for the matrix R_k, in practice, may not generate positive definite matrices and the search direction would not be a descent direction for F, it seems reasonable to define updating formulae which preserve, without conditions, positive definiteness along the iterative process. As Levenberg-Marquardt is a very simple positive definite correction of Gauss-Newton, it is reasonable to expect that more sophisticated positive definite corrections of Gauss-Newton work properly. So, we proposed in Costa and Fernandes (2003) a scheme similar to the Levenberg-Marquardt implementation of Wolfe (1978) (a scaled trust-region strategy) combined with a quasi-Newton technique. Our aim was to generate a positive definite correction matrix, in a factorized form $L^T L$, to overcome a possible singularity of $J^T J$ and to obtain q-superlinear convergence when applied to problems where $\Sigma(x)$ is symmetric and positive definite at the solution, x^*. In this context, we have deduced two structured and factorized quasi-Newton updates for the matrix L, therein denoted by Type L and Type A, which have proved to be very successful. The updates for L_k are as follows:

$$L_{k+1} = A_k + \frac{A_k \overline{s}_k}{a_{2k}} \left(\sqrt{\frac{a_{2k}}{a_{1k}}} \, \overline{w}_k - A_k^T A_k \overline{s}_k \right)^T \qquad (1.5)$$

where the matrix A_k is defined by $L_k + J(x_k) - J(x_{k+1})$ for Type A formula and by L_k for Type L formula. a_{1k} and a_{2k} are scalars defined

by $a_{1k} = \overline{s}_k^T \overline{w}_k$ and $a_{2k} = \overline{s}_k^T A_k^T A_k \overline{s}_k$, where

$$\overline{s}_k = x_{k+1} - x_k, \tag{1.6}$$

$$\overline{w}_k = \overline{z}_k - J(x_{k+1})^T J(x_{k+1})\overline{s}_k \tag{1.7}$$

and

$$\overline{z}_k = \overline{y}_k = \nabla F(x_{k+1}) - \nabla F(x_k) \tag{1.8}$$

or

$$\overline{z}_k = \overline{v}_k + J(x_{k+1})^T J(x_{k+1})\overline{s}_k. \tag{1.9}$$

The vector $\overline{v}_k$ is given by

$$\overline{v}_k = \left(J(x_{k+1})^T - J(x_k)^T\right) f(x_{k+1}).$$

Type A and Type L updating formulae for L come from the minimization, with respect to L_{k+1}, of $\left\|(J(x_{k+1}) + L_{k+1})^T - (J(x_k) + L_k)^T\right\|_F$ and of $\left\|L_{k+1}^T - L_k^T\right\|_F$ respectively, both subject to $L_{k+1}^T \overline{h} = \overline{w}_k$ and $L_{k+1}\overline{s}_k = \overline{h}$, for any unknown m-dimensional vector $\overline{h}$, such that $\overline{h}^T \overline{h} = \overline{s}_k^T \overline{w}_k$. In this context (Costa and Fernandes (2003)), the updating formulae were deduced under the conditions that $\overline{s}_k$ and $\overline{w}_k$ satisfy $\overline{s}_k^T \overline{w}_k > 0$ and that $A_k \overline{s}_k \neq 0$ for all k.

The local convergence analysis therein presented for the factorized quasi-Newton formulae uses very well known results of Broyden et al. (1973) and Dennis and Moré (1974). It is based on the bounded deterioration property of the sequence $\{J(x_k)^T J(x_k) + L_k^T L_k\}$ of approximations to $\nabla^2 F(x^*)$ and on the q-linear convergence of the sequence $\{x_k\}$ to the minimizer x^*. For the sake of completeness, we include here the main theorems. Let D be an open and convex subset of $\mathbb{R}^n$ that contains x^*. The assumptions made to prove the theorems are the following:

(A1) There exist positive constants ξ_1, ξ_2 and p such that

$$\left\|\nabla^2 F(u) - \nabla^2 F(x^*)\right\| \leq \xi_1 \left\|u - x^*\right\|^p$$

for any u in D, and

$$\|J(u_1) - J(u_2)\| \le \xi_2 \|u_1 - u_2\|^p$$

for any u_1 and u_2 in D.

(A2) $\nabla^2 F$ is symmetric and positive definite at x^*.

Theorem 1.1. *(Theorem 5 in Costa and Fernandes (2003)) Suppose that the Assumptions (A1) and (A2) are satisfied. Let the matrix L_k be updated by formula (1.5), where $\overline{z}_k$ is given by (1.8) or (1.9). Let the sequence $\{x_k\}$ be generated by*

$$x_{k+1} = x_k - \left(J(x_k)^T J(x_k) + L_k^T L_k\right)^{-1} J(x_k)^T f(x_k). \qquad (1.10)$$

Then, for any $r \in (0,1)$, there exist positive constants $\varepsilon(r)$ and $\delta(r)$ such that if $\|x_1 - x^*\| \le \varepsilon(r)$ and $\|B_1 - \nabla^2 F(x^*)\|_{F,M} \le \delta(r)$, and if $\Sigma(x^*)$ is a symmetric positive definite matrix, the sequence $\{x_k\}$ is well defined and converges q-linearly to x^* with

$$\|x_{k+1} - x^*\| \le r \|x_k - x^*\|, \quad k \ge 1.$$

Furthermore, the sequences $\{\|B_k\|\}$, $\{\|B_k^{-1}\|\}$, $\left\{\left\|B_k^{\#}\right\|\right\}$ and $\left\{\left\|\left(B_k^{\#}\right)^{-1}\right\|\right\}$ are uniformly bounded and the matrix B_{k+1} satisfies the bounded deterioration property with respect to $\nabla^2 F(x^*)$, with B_{k+1} given by

$$B_{k+1} = B_k^{\#} + \frac{\overline{w}_k \overline{w}_k^T}{\overline{w}_k^T \overline{s}_k} - \frac{A_k^T A_k \overline{s}_k \overline{s}_k^T A_k^T A_k}{\overline{s}_k^T A_k^T A_k \overline{s}_k}$$

where

$$B_k^{\#} = J(x_{k+1})^T J(x_{k+1}) + A_k^T A_k$$

and

$$B_k = J(x_k)^T J(x_k) + L_k^T L_k.$$

Theorem 1.2. *(Theorem 6 in Costa and Fernandes (2003)) Suppose that the hypotheses of Theorem 1.1 hold. Then the sequence $\{x_k\}$ generated by (1.10) converges q-superlinearly to x^*, that is,*

$$\lim_{k \to \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0.$$

In some large-scale applications, it is more practical to compute the gradient $\nabla F(x)$ directly using computational techniques (see Chapter 7 of Nocedal and Wright (2000)) than to compute the first partial derivatives (1.2) separately to form $J(x)^T f(x)$ from (1.3). Since, for these problems, we do not have access to $J(x)$, we can not exploit the special least squares structure, and therefore the algorithm therein proposed does not apply.

As some constrained optimization problems have objective functions of the form (1.1), this paper aims to extend the theory of the structured and factorized quasi-Newton techniques to constrained problems. In particular, the factorized formulae are now used to approximate the terms of the Lagrangian Hessian matrix which involve the second derivatives information of the residual functions and of the constraint functions (see more details in Section 3).

Taking into account the success of the software package LOQO - an interior-point code for general (smooth) nonlinear optimization problems - the proposed extension will be analyzed and implemented in a similar interior-point framework context.

We consider now the problem of minimizing a sum of squared nonlinear functions subject to constraints. In a general form, the problem is

$$
\begin{aligned}
\operatorname*{minimize}_{x \in R^n} \ F(x) \quad &\equiv \quad \frac{1}{2} \sum_{j=1}^{m} \left(f_j(x)\right)^2 \\
\text{subject to} \quad b \quad &\leq \quad h(x) \leq b + r, \\
l \quad &\leq \quad x \leq u, \quad\quad\quad\quad (1.11)
\end{aligned}
$$

where $h_k : \mathbb{R}^n \to \mathbb{R}$ for $k = 1, \ldots, \overline{m}$ are twice continuously differentiable functions. r is the vector of ranges on the constraints $h(x)$, u and l are the vectors of the upper and lower bounds on the variables respectively and b is assumed to be a finite real vector. Elements of the vectors r, l and u are real numbers subject to the following limitations: $0 \leq r_k \leq \infty$; $-\infty \leq l_i, u_i \leq \infty$ for $k = 1, \ldots, \overline{m}$, $i = 1, \ldots, n$. Constraints of the form $b \leq h(x) \leq b + r$ are denoted by range constraints. When upper and lower bounds on the x variables do not

exist, the vector x is considered free. Equality constraints are treated as range constraints with $r = 0$.

The algorithm herein presented is a particular version of Vanderbei and Shanno (1999) algorithm for nonlinear least squares problems.

This algorithm is called an *infeasible interior-point method*. The term *infeasible* refers to the fact that primal feasibility is not required by the algorithm at the beginning although it is enforced throughout the process. The term *interior-point* refers to the fact that the slack variables are required to be positive at the beginning and maintained as so. Other interior-point methods have been proposed to solve nonlinear problems. Some references in this domain are Armand et al. (2000), Byrd et al. (1999), Byrd et al. (2000), El-Bakry et al. (1996), Forsgren and Gill (1998) and Gay et al. (1997).

2   The interior-point paradigm

We begin with a brief description of the interior-point approach to solve (1.11). We refer to Vanderbei and Shanno (1999) for details. Adding slack variables w, p, g and t, (1.11) becomes

$$
\begin{aligned}
\text{minimize} \quad F(x) \;&\equiv\; \frac{1}{2}\sum_{j=1}^{m}\left(f_j\left(x\right)\right)^2 \\
\text{subject to} \quad h(x) - w \;&=\; b, \\
h(x) + p \;&=\; b + r, \\
x - g \;&=\; l, \\
x + t \;&=\; u, \qquad\qquad (2.1)
\end{aligned}
$$

$$
w, p, g, t \geq 0.
$$

The nonnegativity constraints are then eliminated by incorporating them in logarithmic barrier terms in the objective function, transfor-

ming (2.1) into

minimize $b_\mu(x, w, p, g, t)$
subject to $h(x) - w = b;\ \ h(x) + p = b + r;\ \ x - g = l;\ \ x + t = u,$

$$(2.2)$$

where the objective function

$$b_\mu(x, w, p, g, t) \ = \ F(x) - \mu \sum_{j=1}^{\overline{m}} \ln(w_j) - \mu \sum_{j=1}^{\overline{m}} \ln(p_j) - \mu \sum_{i=1}^{n} \ln(g_i)$$

$$-\mu \sum_{i=1}^{n} \ln(t_i)$$

is the classical logarithmic barrier function (Fiacco and McCormick (1990)) with μ a positive parameter. Note that, we could have introduced the simple bounds directly in barrier terms, but the presented approach makes the Karush-Kuhn-Tucker (KKT) conditions easier to write. From the Lagrangian function for problem (2.2)

$$\begin{aligned} \mathbb{L}(x, w, p, g, t, v, q, z, s) \ = \ & b_\mu(x, w, p, g, t) - v^T(h(x) - w - b) \\ & -q^T(r + b - h(x) - p) - z^T(x - g - l) \\ & -s^T(u - x - t), \end{aligned}$$

we may produce the standard primal-dual system

$$\begin{aligned} \nabla F(x) - \nabla h(x)^T y - z + s &= 0, \\ y + q - v &= 0, \\ -\mu e_1 + WVe_1 &= 0, \\ -\mu e_1 + PQe_1 &= 0, \\ -\mu e_2 + GZe_2 &= 0, \\ -\mu e_2 + TSe_2 &= 0, \\ h(x) - w - b &= 0, \\ w + p - r &= 0, \\ x - g - l &= 0, \\ x + t - u &= 0, \end{aligned} \qquad (2.3)$$

where $V = diag(v_1, \ldots, v_{\overline{m}})$, $Q = diag(q_1, \ldots, q_{\overline{m}})$, $Z = diag(z_1, \ldots, z_n)$, $S = diag(s_1, \ldots, s_n)$, $W = diag(w_1, \ldots, w_{\overline{m}})$, $G = diag(g_1, \ldots, g_n)$, $P = diag(p_1, \ldots, p_{\overline{m}})$, $T = diag(t_1, \ldots, t_n)$, $e_1 = (1, 1, \ldots, 1)^T$ and $e_2 = (1, 1, \ldots, 1)^T$ are $\overline{m}$ and n vectors respectively, $\nabla h(x)$ is the Jacobian matrix of the constraints $h(x)$ and $y = v - q$. The first two equations define the conditions of dual feasibility, the next four equations are the complementarity conditions and the last four equations define the primal feasibility.

When applying Newton's method to obtain the solution for (2.3), the resulting system is not symmetric, but it can be symmetrized giving,

$$
\left[
\begin{array}{ccccc|ccccc}
-T^{-1}S & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -I & 0 \\
0 & -G^{-1}Z & 0 & 0 & 0 & 0 & 0 & -I & 0 & 0 \\
0 & 0 & -P^{-1}Q & 0 & 0 & 0 & -I & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -I & -I & 0 & 0 & I \\
0 & 0 & 0 & 0 & -H & \nabla h^T & 0 & I & -I & 0 \\
\hline
0 & 0 & 0 & -I & \nabla h & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -I & -I & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -I & 0 & 0 & I & 0 & 0 & 0 & 0 & 0 \\
-I & 0 & 0 & 0 & -I & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & I & 0 & 0 & 0 & 0 & 0 & V^{-1}W
\end{array}
\right] \triangle = a
$$

(2.4)

where the vectors $\triangle$ and a are given by $(\triangle t, \triangle g, \triangle p, \triangle w, \triangle x, \triangle y, \triangle q, \triangle z, \triangle s, \triangle v)$ and $(-\gamma_s, -\gamma_z, -\gamma_q, \beta, \sigma, \rho, -\alpha, \upsilon, -\tau, \gamma_w)$ respectively,

$$
H \equiv \nabla_{xx}^2 \mathbb{L}(x, y) = J(x)^T J(x) + \sum_{j=1}^{m} f_j(x) \nabla^2 f_j(x) - \sum_{j=1}^{\overline{m}} y_j \nabla^2 h_j(x),
$$

(2.5)

and

$$
\begin{aligned}
\gamma_s &= \mu T^{-1} e_2 - s \\
\gamma_z &= \mu G^{-1} e_2 - z \\
\gamma_q &= \mu P^{-1} e_1 - q \\
\beta &= y + q - v \\
\sigma &= \nabla F(x) - \nabla h(x)^T y - z + s
\end{aligned}
$$

$$
\begin{aligned}
\rho &= w + b - h(x) \\
\alpha &= r - w - p \\
\upsilon &= l - x + g \\
\tau &= u - x - t \\
\gamma_w &= \mu V^{-1} e_1 - w.
\end{aligned}
$$

Note that ρ, α, υ and τ measure *primal infeasibility* and β and σ measure *dual infeasibility*. The symmetry of this system suggests a systematic process of elimination which brings us to the so-called reduced KKT system:

$$
\left[\begin{array}{c|c} -(H+D) & \nabla h(x)^T \\ \hline \nabla h(x) & E \end{array} \right] \left[\begin{array}{c} \triangle x \\ \hline \triangle y \end{array} \right] = \left[\begin{array}{c} \sigma - ZG^{-1}\widehat{\upsilon} - ST^{-1}\widehat{\tau} \\ \hline \rho - E\left(\widehat{\beta} - QP^{-1}\widehat{\alpha}\right) \end{array} \right],
$$

where

$$
\begin{aligned}
E &= \left(VW^{-1} + QP^{-1}\right)^{-1} \\
D &= ZG^{-1} + ST^{-1}
\end{aligned}
$$

and

$$
\begin{aligned}
\widehat{\beta} &= \beta - VW^{-1}\gamma_w \\
\widehat{\alpha} &= \alpha - PQ^{-1}\gamma_q \\
\widehat{\upsilon} &= \upsilon + GZ^{-1}\gamma_z \\
\widehat{\tau} &= \tau - TS^{-1}\gamma_s.
\end{aligned}
$$

Once the reduced system has been solved for $\triangle x$ and $\triangle y$, the other $\triangle$ variables that were eliminated are recuperated by

$$
\begin{aligned}
\triangle w &= -E\left(\widehat{\beta} - QP^{-1}\widehat{\alpha} + \triangle y\right) \\
\triangle q &= P^{-1}Q\left(\triangle w - \widehat{\alpha}\right) \\
\triangle z &= G^{-1}Z\left(\widehat{\upsilon} - \triangle x\right) \\
\triangle s &= T^{-1}S\left(\triangle x - \widehat{\tau}\right) \\
\triangle t &= TS^{-1}\left(\gamma_s - \triangle s\right)
\end{aligned}
$$

$$\begin{aligned}
\triangle g &= GZ^{-1}\left(\gamma_z - \triangle z\right) \\
\triangle p &= PQ^{-1}\left(\gamma_q - \triangle q\right) \\
\triangle v &= VW^{-1}\left(\gamma_w - \triangle w\right).
\end{aligned}$$

Let

$$N = H + D + \nabla h(x)^T E^{-1} \nabla h(x) \qquad (2.6)$$

denote the dual normal matrix.

Theorem 2.1. *(Similar to Theorem 1 in Vanderbei and Shanno (1999))*
If N is nonsingular, then system (2.4) has a unique solution.

Proof. Theorem 5.1 in Costa (2002). $\qquad\qquad\qquad\square$

From an initial point $(x_1, w_1, p_1, g_1, t_1, y_1, q_1, z_1, s_1, v_1)$, the algorithm then proceeds iteratively as follows:

$$\begin{aligned}
x_{k+1} &= x_k + \overline{\alpha}_k \triangle x_k, & y_{k+1} &= y_k + \overline{\alpha}_k \triangle y_k, \\
w_{k+1} &= w_k + \overline{\alpha}_k \triangle w_k, & q_{k+1} &= q_k + \overline{\alpha}_k \triangle q_k, \\
p_{k+1} &= p_k + \overline{\alpha}_k \triangle p_k, & z_{k+1} &= z_k + \overline{\alpha}_k \triangle z_k, \\
g_{k+1} &= g_k + \overline{\alpha}_k \triangle g_k, & s_{k+1} &= s_k + \overline{\alpha}_k \triangle s_k, \\
t_{k+1} &= t_k + \overline{\alpha}_k \triangle t_k, & v_{k+1} &= v_k + \overline{\alpha}_k \triangle v_k,
\end{aligned} \qquad (2.7)$$

using the search directions described above, where $\overline{\alpha}_k$ gives the step length and k is the iteration counter.

2.1   The merit function

The idea of a merit function is to ensure that joint progress is made both toward a local minimizer and toward feasibility. The step length $\overline{\alpha}_k$ in (2.7) should be chosen to ensure that the slack and the dual variables are maintained positive and that an appropriate merit function is reduced. The chosen merit function is the penalty function studied by Fiacco and McCormick (1990), which has the form

$$\Psi_{\overline{\beta},\,\mu}\left(x,w,p,g,t\right) = b_\mu(x,w,p,g,t) + \frac{\overline{\beta}}{2}\left\|\rho\left(x,w\right)\right\|_2^2 + \frac{\overline{\beta}}{2}\left\|\alpha\left(w,p\right)\right\|_2^2$$

$$+ \frac{\overline{\beta}}{2}\left\|\upsilon\left(x,g\right)\right\|_2^2 + \frac{\overline{\beta}}{2}\left\|\tau\left(x,t\right)\right\|_2^2 \tag{2.8}$$

when applied to problem (2.2), where $\overline{\beta} \geq 0$ is the penalty parameter.

In practice, the choice of an $\overline{\alpha}_{\max}$ to ensure the positivity of the slack and the dual variables means that, for $\triangle w_j < 0$, $\triangle p_j < 0$, $\triangle g_i < 0$, $\triangle t_i < 0$, $\triangle y_j < 0$, $\triangle q_j < 0$, $\triangle z_i < 0$, $\triangle s_i < 0$, $\triangle v_j < 0$

$$\overline{\alpha}_{\max} \;=\; \min\left\{1, 0.95\min\left(-\frac{w_j}{\triangle w_j}, -\frac{p_j}{\triangle p_j}, -\frac{g_i}{\triangle g_i}, -\frac{t_i}{\triangle t_i}, -\frac{y_j}{\triangle y_j}, \right.\right.$$

$$\left.\left. -\frac{q_j}{\triangle q_j}, -\frac{z_i}{\triangle z_i}, -\frac{s_j}{\triangle s_j}, -\frac{v_j}{\triangle v_j}\right)\right\}$$

with $i = 1,\ldots,n$ and $j = 1,\ldots,\overline{m}$. The interval $(0,\overline{\alpha}_{\max}]$ is then searched by backtracking, for an $\overline{\alpha}_k$ that produces a reduction in the function $\Psi_{\overline{\beta},\,\mu}$, using merit function evaluations only.

2.2 Choice of the $\overline{\beta}$ parameter

The following theorem shows, among other things, that for large enough values of $\overline{\beta}$, the search directions, corresponding to the primal variables of (2.2), defined by (2.4) are descent directions for the merit function (2.8) whenever the problem is strictly convex.

Theorem 2.2. *(Similar to Theorem 2 in Vanderbei and Shanno (1999)) Suppose that the dual normal matrix N (2.6) is positive definite. Then, the search directions have the following properties:*

1. *If $\rho = 0$, $\alpha = 0$, $\upsilon = 0$ and $\tau = 0$, then*

$$\begin{bmatrix} \nabla_x b_\mu \\ \nabla_w b_\mu \\ \nabla_p b_\mu \\ \nabla_g b_\mu \\ \nabla_t b_\mu \end{bmatrix}^T \begin{bmatrix} \triangle x \\ \triangle w \\ \triangle p \\ \triangle g \\ \triangle t \end{bmatrix} \leq 0.$$

2. There exists $\overline{\beta}_{\min} \geq 0$ such that, for every $\overline{\beta} > \overline{\beta}_{\min}$,

$$
\begin{bmatrix}
\nabla_x \Psi_{\overline{\beta},\,\mu} \\
\nabla_w \Psi_{\overline{\beta},\,\mu} \\
\nabla_p \Psi_{\overline{\beta},\,\mu} \\
\nabla_g \Psi_{\overline{\beta},\,\mu} \\
\nabla_t \Psi_{\overline{\beta},\,\mu}
\end{bmatrix}^T
\begin{bmatrix}
\triangle x \\
\triangle w \\
\triangle p \\
\triangle g \\
\triangle t
\end{bmatrix}
\leq 0.
$$

In both cases, equality holds if and only if (x, w, p, g, t) satisfies (2.3) for some (y, q, z, s, v).

Proof. Theorem 5.2 in Costa (2002). $\square$

In practice, $\overline{\beta}$ is initialized to 0 and is unchanged as long as $(\triangle x, \triangle w, \triangle p, \triangle g, \triangle t)$ is a descent direction for $\Psi_{\overline{\beta},\,\mu}$. When $(\triangle x, \triangle w, \triangle p, \triangle g, \triangle t)$ fails as a descent direction, then $\overline{\beta}$ is calculated using $\overline{\beta} = 10\overline{\beta}_{\min}$.

3   Algorithm modifications for nonconvex optimization

For convex optimization, the use of the merit function described in the Subsection 2.1 to guide the selection of a step length yields an efficient and robust algorithm. However, for nonconvex optimization problems, the matrix N (2.6) may fail to be positive definite. In this section, we describe the algorithm changes that we made to address this issue.

Theorem 2.2 shows that the direction $(\triangle x, \triangle w, \triangle p, \triangle g, \triangle t)$ has desirable descent properties for both the barrier function and the merit function provided that N is positive definite. When N is indefinite, the algorithm might converge to a point which is not a local minimum.

For the positive definiteness of the dual normal matrix, we need to guarantee that H is positive definite, to be able to ensure that the direction $(\triangle x, \triangle w, \triangle p, \triangle g, \triangle t)$ is a descent direction for $\Psi_{\overline{\beta},\,\mu}$. For

our problem (recall (2.5)), the Jacobian matrix $J(x)$ can be computed analytically or numerically in a expedite form and the term $J(x)^T J(x)$ of H is available at an acceptable cost and is positive definite if $J(x)$ is full rank. Our aim is to surpass the calculation of the second derivatives of the residual functions and of the constraint functions, and at same time to guarantee that the H matrix is approximated by a positive definite matrix. Since in an unconstrained least squares context (Costa and Fernandes (2003)), we have deduced two factorized quasi-Newton formulae and they have proved to be very successful, we try now to extend them to this case as well. So, using a similar argument, the second order terms of H are to be approximated by a positive definite matrix.

Our proposal considers a factorized quasi-Newton approximation of the form $L^T L$ to the last two terms of H. So, at each iteration k,

$$H_k \approx J(x_k)^T J(x_k) + L_k^T L_k.$$

The updates for L_k (see (1.5)) that result from the step from iterate k to iterate $k+1$ will make use of the vectors, $\overline{s}_k$, $\overline{z}_k$, $\overline{w}_k$ and $\overline{v}_k$, which are defined as follows:

$$\overline{z}_k = \overline{y}_k$$

or

$$\overline{z}_k = \overline{v}_k + J(x_{k+1})^T J(x_{k+1})\overline{s}_k$$

where $\overline{y}_k$ and $\overline{v}_k$ are now given by

$$
\begin{aligned}
\overline{y}_k &= \nabla F(x_{k+1}) - \nabla h(x_{k+1})^T y_{k+1} - \nabla F(x_k) + \nabla h(x_k)^T y_k - z_{k+1} \\
&\quad + z_k + s_{k+1} - s_k \\
\overline{v}_k &= \left(J(x_{k+1})^T - J(x_k)^T \right) f(x_{k+1}) - \left(\nabla h(x_{k+1})^T - \nabla h(x_k)^T \right) y_{k+1}.
\end{aligned}
$$

The vector $\overline{v}_k$ comes easily from

$$
\begin{aligned}
L_{k+1}^T L_{k+1} \overline{s}_k &= \left[\sum_{j=1}^m f_j\left(x_{k+1}\right) \nabla^2 f_j\left(x_{k+1}\right) - \sum_{i=1}^{\overline{m}} y_{i,k+1} \nabla^2 h_i(x_{k+1}) \right] \overline{s}_k \\
&\cong \sum_{j=1}^m f_j\left(x_{k+1}\right) \left[\nabla f_j\left(x_{k+1}\right) - \nabla f_j\left(x_k\right) \right] \\
&\quad - \sum_{i=1}^{\overline{m}} y_{i,k+1} \left[\nabla h_i(x_{k+1}) - \nabla h_i(x_k) \right] \\
&= \left(J_{k+1}^T - J_k^T \right) f_{k+1} - \left(\nabla h_{k+1}^T - \nabla h_k^T \right) y_{k+1} = \overline{v}_k.
\end{aligned}
$$

If $\nabla_{xx}^2 \mathbb{L}$ is not positive definite, the condition $\overline{s}_k^T \overline{w}_k > 0$ may not hold when $\overline{s}_k$ and $\overline{w}_k$ are defined by (1.6) and (1.7), even when the iterates are close to the solution. To overcome this difficulty, whenever the condition $\overline{s}_k^T \overline{w}_k > 0$ failed we used $\left| \overline{s}_k^T \overline{w}_k \right|$ instead. A *skipping* strategy is also included, whenever the scalars that appear in the denominators of the updates became too small, i.e., we do *skip* the update formulae if

$$
\overline{s}_k^T \overline{w}_k > \gamma \text{ and/or } \overline{s}_k^T A_k^T A_k \overline{s}_k > \gamma
$$

are not satisfied, where γ is a positive parameter (10^{-12}, say).

In Yamashita and Yabe (1996), among other things, the authors prove local and superlinear convergence of a primal-dual interior-point method for constrained optimization, when a quasi-Newton approximation is used to approximate the Hessian of the Lagrangian and a single step size rule is implemented for the primal and the dual variables. Besides the conditions on the objective function, constraint functions, barrier parameter and on the parameter in the step size rule, to prove superlinear convergence they also assume that the sequence of quasi-Newton updates satisfies the bounded deterioration property. In our view, the analysis can be extended to our factorized quasi-Newton updates as long as we are able to prove the bounded deterioration property of the sequence $\{B_k\}$ with respect to $\nabla_{xx}^2 \mathbb{L}(x^*, y^*)$. This is now under investigation.

4   Details of implementation

This section provides implementation details of the algorithm that we
have coded and tested.

4.1   The initial point

In an interior-point code, to start the algorithm we need to provide
initial values for all slack and dual variables. It is important to provide
these values to satisfy at least some of the equations. Similarly to what
is done in Vanderbei (1998), the algorithm initializes the variables as
follows. First, giving a starting point x_0 of the problem, x and y are
found as solutions to the following system:

$$\left[\begin{array}{c|c} -\left(J^T J + I\right) & \nabla h^T \\ \hline \nabla h & I \end{array}\right] \left[\begin{array}{c} x \\ y \end{array}\right] = \left[\begin{array}{c} c \\ b \end{array}\right]$$

where J and ∇h are computed at x_0 and $c = \nabla F(0)$. In practice, if
∇F is not defined at the point 0 we use $x = x_0$ and $y = 1$. The other
variables are set as follows:

$$g = \max\left(abs(x - l), \theta\right)$$
$$z = \max((abs(x), \theta)$$
$$t = \max\left(abs(u - x), \theta\right)$$
$$s = \max(abs(x), \theta)$$
$$w = \max\left(abs\left(h(x) - b\right), \theta\right)$$
$$p = \max\left(abs(r - w), \theta\right)$$
$$q = \max\left(abs(y), \theta\right)$$
$$v = \max(abs(y), \theta),$$

where $\max()$ and $abs()$ denote componentwise maximum and absolute
value, respectively. The parameter θ is used to guarantee that all the
variables constrained to be nonnegative are at least as large as θ. $\theta = 1$
has proved to be a good choice.

4.2   Stopping rule

The stopping rule used in the algorithm considers both primal and dual feasibilities. The level of primal infeasibility is measured by

$$primal\ inf. = \frac{\sqrt{\|\rho\|^2 + \|\tau\|^2 + \|\alpha\|^2 + \|v\|^2}}{\|b\| + 1},$$

and the dual infeasibility by

$$dual\ inf. = \frac{\sqrt{\|\sigma\|^2 + \|\beta\|^2}}{\|c\| + 1}.$$

A solution is declared primal/dual feasible if these measures are less than 10^{-6} (Vanderbei and Shanno (1999)). In practice, if ∇F is not defined at the point 0 we use $c = 0$.

4.3   The barrier parameter

The duality measure parameter μ of the objective function in (2.2) is computed, at each iteration, by

$$\mu = \gamma(z^T g + v^T w + s^T t + p^T q)/(2\overline{m} + 2n)$$

where γ is a scale factor ($0 \leq \gamma < 1$), with default value equals to 0.1. We refer to Vanderbei (1998) for details.

4.4   Handling infinities

In our problem formulation (1.11) the variables have two sided bounds. But in real-world problems we may have some variables with two-sided bounds, others with only one-sided bounds (either upper or lower) and yet others without bounds (so-called *free variables*). Missing bounds are handled as follows. We simply omit the slack variable associated with the missing bound as well as the corresponding dual variable.

If the x variables are *free*, the technique replaces each one with the difference of two nonnegative variables, which means adding the constraints $x - g + t = 0$, g, $t \geq 0$ to the problem.

If we have a pure inequality constraint, which means that the i-th constraint has infinite range, then we leave out the slack variable p_i, associated with this constraint, and the corresponding dual variable q_i.

5   Computational experiments

To compare our factorized formulae (Type L and Type A) with the BFGS-like formula of Yabe and Takahashi (1991) in this primal-dual interior-point framework we selected 32 least squares constrained problems from Hock and Schittkowski (1981) collection (HS). The tests were done in double precision arithmetic with a Pentium II and Fortran 90 (LINUX). The initial matrix L_1 is always set to $[10^{-4}I\,|\,0]^T$.

We present in Table 1 some ideas about the performance of the factorized quasi-Newton updates Type A and BFGS against the Type L formula, using Al-Baali rule (1991). The performance of these factorized formulae is measured by computing the numbers (TI,RI), (TFE,RFE) and TF. These numbers are related to the total number of iterations (TI), total number of merit function evalutions (TFE) and failures (TF) required to solve the set of problems. A value of RI (similarly for RFE) gives the average of $N = 32$ ratios of the form

$$r_i = \begin{cases} \dfrac{C_{2,i}}{C_{1,i}} & \text{if } C_{2,i} \leq C_{1,i} \\[2mm] 2 - \dfrac{C_{1,i}}{C_{2,i}} & \text{if } C_{2,i} > C_{1,i} \end{cases}$$

where $C_{1,i}$ and $C_{2,i}$, for $i = 1, 2, \ldots, N$, denote the costs required to solve problem i by two methods (1 and 2, respectively). It includes all problems even if they fail. r_i is equal to 1 when the two methods fail, r_i is equal to 0 if method 1 fails and r_i is equal to 2 if method 2 fails.

Thus, the RI ratios take all kind of termination into account and

Formula	TI	RI	TFE	RFE	TF
Type L	429	1	2532	1	0
Type A	396	0.988	1788	1.008	0
BFGS	469	1.013	2025	1.033	1

Table 1: Performance of Type A and BFGS-like formulae against Type L.

always belong to [0,2]. A value of RI < 1 indicates that method 2 improves over method 1 by $100(1- \text{RI})\%$. If RI > 1 then method 1 improves $100(\text{RI} -1)\%$ over method 2. RI = 1 means that the two methods are comparable to each other.

As conclusion, Table 1 shows that Type A formula improves over Type L by 1.2% and Type L improves over BFGS by 1.3%, as far as number of iterations is concerned. For the number of function evaluations, Type L formula improves over Type A by 0.8% and over BFGS by 3.3%. The large total number of function evaluations required by Type L formula is due to a single problem. The advantage of this rule is that it is not sensitive to large values which may occur in a very small number of problems in the set.

For the second group of experiments, we compare our structured and factorized quasi-Newton interior-point framework with LOQO (Vanderbei and Shanno (1999), Shanno and Vanderbei (2000) and Benson et al. (2004)). In these papers, exact second derivatives are computed, the least squares structure is not taken in consideration and to maintain positive definiteness of the dual normal matrix, the authors replace the H matrix with a diagonal perturbation thereof $\widehat{H} = H + \lambda I, \lambda \geq 0$.

In Table 2, we report iteration counts on the 32 HS problems obtained by our Type L and Type A (L/A) factorized updates. In this table, n is the number of variables and nc is the total number of constraints which include range constraints, simple bounds on the variables and constraints due to the treatment of free variables. The iteration counts of the column LOQO were taken from Benson et al. (http://www.princeton.edu/~rvdb/cute_table.pdf) (see reference [1]

Problem(n/nc)	L/A	LOQO	Problem(n/nc)	L/A	LOQO
HS01(2/1)	14	33	HS31 (3/7)	13	13
HS02 (2/1)	14	32	HS32 (3/5)	16*	23
HS06 (2/1)	11	11	HS42 (4/2)	9	12
HS14 (2/2)	9	13	HS46 (5/2)	15*/18*	20
HS15 (2/3)	22	25	HS48 (5/2)	8	8
HS16 (2/5)	11	18	HS49 (5/2)	11*	24
HS17 (2/5)	18	27	HS50 (5/3)	9	16
HS18 (2/6)	15	18	HS51 (5/3)	9	8
HS20 (2/5)	18*	24	HS52 (5/3)	9	8
HS22 (2/2)	13	9	HS53 (5/13)	15	11
HS23 (2/9)	13*	18	HS57 (2/3)	17	16
HS25 (3/6)	20/17	26	HS60 (3/7)	35/18	18
HS26 (3/1)	14*	15	HS65 (3/7)	12	21
HS27 (3/1)	23/18	58	HS70 (4/9)	54/49	27
HS28 (3/1)	8	8	HS77 (5/2)	24*/29*	16
HS30 (3/7)	20	9	HS79 (5/3)	13/11	9

Table 2: Iteration counts on Hock-Schittkowski problems.

in Benson et al. (2004)).

For the initial approximations, x_0, we considered the Hock and Schittkowski (1981) published values as well.

We test our framework on other three problems with variable size (n=30, 100, 200) taken from the CUTE collection (Bongartz et al. (1995)). Table 3 reports the iteration counts. The numbers in the last column of the table were obtained running LOQO V4.01.

As in LOQO, the stopping rule includes now *primal inf.* and *dual inf.* $< 10^{-6}$ and 8 digits of agreement between the primal and dual objective functions. For some problems, this level of accuracy was unattainable by our framework. They are identified in the tables with the character $*$. The iteration counts shown in the tables were obtained by relaxing the parameters in the stopping rule. Table 4 contains details of the relaxed parameters. In the table, *sigfig* gives the number of digits of agreement between primal and dual objective functions attained at the solution found.

Problem (n)	L/A	LOQO V4.01
argtrig (30)	6	7
argtrig (100)	6	10
argtrig (200)	6	8
brownal (30)	13/12	12
brownal (100)	13	14
brownal (200)	15	13
broydn3d (30)	13/17	11
broydn3d (100)	15/47*	11
broydn3d (200)	16/59*	11

Table 3: Iteration counts on three larger CUTE problems.

Problem	relaxations
HS20	(L/A -*sigfig*=0)
HS23	(L/A -*sigfig*=0)
HS26	(L/A -*sigfig*=6)(A -*dual inf.*$< 10^{-5}$)
HS32	(L/A -*sigfig*=7)
HS46	(L/A -*sigfig*=3)(L -*primal inf.*$< 10^{-5}$)(L/A -*dual inf.*$< 10^{-3}$)
HS49	(L/A -*sigfig*=5)
HS77	(L -*sigfig*=5, *dual inf.*$< 10^{-5}$)(A -*sigfig*=2, *dual inf.*$< 10^{-2}$)
broydn3d	(n=100) (A -*sigfig*=5, *dual inf.*$< 10^{-5}$)
broydn3d	(n=200) (A -*sigfig*=7)

Table 4: Problems that required relaxation of the parameters.

6   Conclusions

This paper presents a particular version of Vanderbei and Shanno (1999) interior-point algorithm for constrained nonlinear least squares problems.

Factorized quasi-Newton updates that exploit the special least squares structure of the problem are implemented in order to approximate the second order terms of the Lagrangian Hessian. The new updates also guarantee positive definite approximations to the H matrix so ensuring descent directions in practice. Numerical experiments that were carried out on a well known set of small problems indicate that the efficiency of this quasi-Newton approach is comparable to that of a general interior-point algorithm that uses second derivative information and its use may present advantages in those cases when this second-order information is expensive to compute.

References

Al-Baali M. (1991). A rule for comparing two methods in practical optimization. Report N ^{o} 119. University of Damascus.

Armand P., Gilbert J. C. and Jan-Jégou S. (2000). A feasible BFGS interior point algorithm for solving convex minimization problems. *SIAM Journal on Optimization* 11, 199-222.

Benson H. Y., Shanno D. F. and Vanderbei R. J. (2004). Interior-point methods for nonconvex nonlinear programming: Jamming and numerical testing. *Mathematical Programming* Ser. A, 99(1), 35-48.

Bongartz I., Conn A. R., Gould N. and Toint Ph. L. (1995). CUTE: Constrained and Unconstrained Testing Environment. *ACM Transactions on Mathematical Software* 21(1), 123-160.

Broyden C. G., Dennis J. E. and Moré J. J. (1973). On the local and superlinear convergence of Quasi-Newton methods. *Journal of Institute of Mathematics and its Applications* 12, 223-245.

Byrd R. H., Hribar M. E. and Nocedal J. (1999). An interior point algorithm for large-scale nonlinear programming. *SIAM Journal on Optimization* 9, 877-900.

Byrd R. H., Gilbert J. C. and Nocedal J. (2000). A trust region method based on interior point techniques for nonlinear programming. *Mathematical Programming* 89, 149-185.

Costa M. F. P. (2002). Técnicas Quasi-Newton Factorizadas em Problemas de Mínimos Quadrados Não Lineares. Ph.D. Thesis, Universidade do Minho.

Costa M. F. P. and Fernandes E. M. G. P. (2003). Factorized Quasi-Newton formulae for least squares problems. Technical Report (to be published). University of Minho.

Dennis J. E. and Moré J. J. (1974). A characterization of superlinear convergence and its application to Quasi-Newton methods. *Mathematics of Computation* 28(126), 549-560.

Dennis J. E., Gay D. M and Welsch R. E. (1981). An adaptive nonlinear least-squares algorithm. *ACM Transactions on Mathematical Software* 7(3), 348-368.

Dennis J. E. and Schnabel R. B. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, NJ.

El-Bakry A. S., Tapia R. A., Tsuchiya T. and Zhang Y. (1996). On the formulation and theory of Newton interior-point method for nonlinear programming. *Journal of Optimization Theory and Applications* 89(3), 507-541.

Fiacco A. V. and McCormick G. P. (1990). *Nonlinear Programming. Sequential Unconstrained Minimization Techniques.* 2nd. ed.. SIAM.

Forsgren A. L. and Gill P. E. (1998). Primal-dual interior methods for nonconvex nonlinear programming. *SIAM Journal on Optimization* 8, 1132-1152.

Gay D. M., Overton M. L. and Wright M. H. (1997). A primal-dual interior

method for nonconvex nonlinear programming. Technical Report 97-4-08, Computing Sciences Research, Bell Laboratories, Murray Hill, NJ.

Gill P. E. and Murray W. (1976). Algorithms for the solution of nonlinear least-squares problems. Technical Report 71, Division of Numerical Analysis and Computing, National Physical Laboratory.

Hock W. and Schittkowski K. (1981). *Test Examples for Nonlinear Programming Codes.* Lecture Notes in Economics and Mathematical Systems 187. Springer-Verlag.

Kelly C. T. (1999). *Iterative Methods for Optimization.* SIAM.

McKeown J. J. (1972). Experience with Peckham's algorithm for minimising sums of squared functions. Technical Report 29, Numerical Optimisation Centre, the Haltfield Polytechnic.

Moré J. J. (1978). The Levenberg-Marquardt algorithm implementation and theory. In: Watson G. A. (ed.), *Numerical Analysis Proceedings*, Lectures Notes in Mathematics 630, Springer-Verlag, 105-116.

Nazareth L. (1980). Some recent approaches to solving large residual nonlinear least squares problems. *SIAM Review* 24(1), 1-11.

Nocedal J. and Wright S. J. (2000). *Numerical Optimization.* 2nd. ed., Springer Series in Operations Research.

Shanno D. F. and Vanderbei R. J. (2000). Interior-point methods for nonconvex nonlinear programming: Orderings and higher-order methods. *Mathematical Programming* Ser. B 87, 303-316.

Vanderbei R. J. (1998). LOQO: An interior-point code for quadratic programming. Technical Report SOR 94-15 (revised). Princeton University.

Vanderbei R. J. and Shanno D. F. (1999). An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications* 13, 231-252.

Wolfe M. A. (1978). *Numerical Methods for Unconstrained Optimization.* Van Nostrand Reinhold Company.

Yabe H. and Takahashi T. (1991). Factorized Quasi-Newton methods for nonlinear least-squares problems. *Mathematical Programming* 51, 75-100.

Yamashita H. and Yabe H. (1996). Superlinear and quadratic convergence of some primal-dual interior point methods for constrained optimization. *Mathematical Programming* 75, 377-397.