

Constrained Nonlinear Least Squares: A Superlinearly Convergent Projected Structured Secant Method

Nezam Mahdavi-Amiri, Narges Bidabadi

Sharif University of Technology, Faculty of Mathematical Sciences
 Azadi Street, Tehran, Iran
 nezamm@sina.sharif.edu, n_bidabadi@mehr.sharif.ir

Abstract- Numerical solution of nonlinear least-squares problems is an important computational task in science and engineering. Effective algorithms have been developed for solving nonlinear least squares problems. The structured secant method is a class of efficient methods developed in recent years for optimization problems in which the Hessian of the objective function has some special structure. A primary and typical application of the structured secant method is to solve the nonlinear least squares problems. We present an exact penalty method for solving constrained nonlinear least-squares problems, when the structured projected Hessian is approximated by a projected version of the structured BFGS formula and give its local two-step Q -superlinear convergence. For robustness, we employ a special nonsmooth line search strategy, taking account of the least squares objective. We discuss the comparative results of the testing of our programs and three nonlinear programming codes from KNITRO on some randomly generated test problems due to Bartels and Mahdavi-Amiri. Numerical results also confirm the practical relevance of our special considerations for the inherent structure of the least squares.

Keywords: Constrained Nonlinear Least Squares, Exact Penalty Methods, Projected Hessian Updating, Structured Secant Updates, Two-Step Superlinear Convergence

© Copyright 2012 Authors - This is an Open Access article published under the Creative Commons Attribution License terms (<http://creativecommons.org/licenses/by/2.0>). Unrestricted use, distribution, and reproduction in any medium are permitted, provided the original work is properly cited.

1. Introduction

Consider the constrained nonlinear least squares (CNLS) problem,

$$\begin{aligned} \min \quad & \phi(x) = \frac{1}{2} F(x)^T F(x) \\ \text{s.t.} \quad & c_i(x) = 0, \quad i = 1, \dots, k \\ & c_i(x) \geq 0, \quad i = k + 1, \dots, k + m, \end{aligned} \quad (1)$$

where, $F(x) = [f_1(x), \dots, f_l(x)]^T$, $x \in R^n$, f_δ , $1 \leq \delta \leq l$, and c_i , $i = 1, \dots, k + m$, are functions from R^n to R , all assumed to be twice continuously differentiable. These problems arise as experimental data analysis problems in various areas of science and engineering such as electrical engineering, medical and biological imaging, chemistry, robotics, vision, and environmental sciences; e.g., see Nievergelt (2000), Golub and Pereyra (2003) and Mullen et al. (2007). The gradient and Hessian of f can be expressed as

$$\nabla \phi(x) = G(x)F(x), \quad (2)$$

and

$$\nabla^2 \phi(x) = G(x)G(x)^T + S(x), \quad (3)$$

where, $G(x)$ is the matrix whose columns are the gradients $\nabla f_\delta(x)$, and

$$S(x) = \sum_{\delta=1}^l f_\delta(x) \nabla^2 f_\delta(x). \quad (4)$$

Using this special structure for approximating the Hessian matrix has been the subject of numerous research papers; see Fletcher and Xu (1987), Dennis et al. (1989), Mahdavi-Amiri

and Bartels (1989), Li et al. (2002), Mahdavi-Amiri and Ansari (2012a), and Bidabadi and Mahdavi-Amiri (2012). Here, we consider the structured BFGS approximate formula (Dennis et al., 1989) for the structured least squares projected Hessian and a special line search scheme given by Mahdavi-Amiri and Ansari (2012c) leading to a more efficient algorithm than general nonlinear programming schemes. An exact penalty function for nonlinear programming problems is defined to be

$$\psi(x, \mu) = \mu\phi(x) + \sum_{i=1}^k |c_i(x)| - \sum_{i=k+1}^{k+m} \min(0, c_i(x)), \quad (5)$$

where, $\mu > 0$ is a penalty parameter. If x^* is a stationary point of (1) and the gradients of the active constraints at x^* are linearly independent, then there exists a real number $\mu^* > 0$ such that x^* is also a stationary point of $\psi(x, \mu)$, for each $0 < \mu \leq \mu^*$. Mahdavi-Amiri and Bartels (1989), based on a general approach for nonlinear problems (See Coleman and Conn, 1982a), proposed a special structured algorithm for minimization of ψ with a fixed value of μ in solving the CNLS problems. Let ε be a small positive number used to identify the near-active (ε -active) constraint set. The algorithm obtains search directions by using an ε -active set

$$AC(x, \varepsilon) = \{i : |c_i(x)| \leq \varepsilon, 1 \leq i \leq k+m\}, \quad (6)$$

and a corresponding merit function:

$$\begin{aligned} \psi_\varepsilon(x, \mu) = & \mu\phi(x) + \sum_{i \in VE(x, \varepsilon)} \text{sgn}(c_i(x))c_i(x) \\ & - \sum_{i \in VI(x, \varepsilon)} c_i(x), \end{aligned} \quad (7)$$

where,

$$\begin{aligned} VE(x, \varepsilon) = & \{i : |c_i(x)| > \varepsilon, 1 \leq i \leq k\}, \\ VI(x, \varepsilon) = & \{i : c_i(x) < -\varepsilon, k+1 \leq i \leq k+m\}. \end{aligned} \quad (8)$$

Step lengths are chosen by using ψ , which is ψ_ε with $\varepsilon = 0$. The optimality conditions are checked by using ψ , as well. The gradient and Hessian of ψ_ε are:

$$\begin{aligned} \nabla \psi_\varepsilon(x, \mu) = & \mu \nabla \phi(x) + E(x)\sigma(x) - I(x)e \\ = & \mu G(x)F(x) + E(x)\sigma(x) - I(x)e, \end{aligned} \quad (9)$$

and

$$\begin{aligned} \nabla^2 \psi_\varepsilon(x, \mu) = & \mu(G(x)G(x)^T + S(x)) \\ & + \sum_{i \in VE(x, \varepsilon)} \text{sgn}(c_i(x))\nabla^2 c_i(x) \\ & - \sum_{i \in VI(x, \varepsilon)} \nabla^2 c_i(x), \end{aligned} \quad (10)$$

where,

$$\begin{aligned} e = & [1 \dots 1]^T, \\ A(x) = & [\dots \nabla c_i(x) \dots]_{i \in AC(x, \varepsilon)}, \\ E(x) = & [\dots \nabla c_i(x) \dots]_{i \in VE(x, \varepsilon)}, \\ I(x) = & [\dots \nabla c_i(x) \dots]_{i \in VI(x, \varepsilon)}, \\ \sigma(x) = & [\dots \text{sgn}(c_i(x)) \dots]_{i \in VE(x, \varepsilon)}^T. \end{aligned} \quad (11)$$

It is well known that the necessary conditions for x^* to be an isolated local minimizer for ψ , under the assumptions made above on ϕ and the c_i , are that there exist multipliers, λ_i^* , for $i \in AC(x^*, 0)$, such that

$$\nabla \psi_0(x^*, \mu) = \sum_{i \in AC(x^*, 0)} \lambda_i^* \nabla c_i(x^*) = A(x^*)\lambda^*, \quad (12)$$

$$-1 \leq \lambda_i^* \leq 1, \quad i \in AC(x^*, 0) \cap \{1, \dots, k\}, \quad (13)$$

$$0 \leq \lambda_i^* \leq 1, \quad i \in AC(x^*, 0) \cap \{k+1, \dots, k+m\}.$$

A point x for which only (12) above is satisfied is called a stationary point of ψ . A minimizer x^* must be a stationary point satisfying (13). One major premise of the algorithm is that the multipliers are only worth estimating in the neighborhoods of stationary points. Nearness to a stationary point is governed by a stationary tolerance $\tau > 0$. The algorithm is considered to be in a local state, if norm of the projected or reduced gradient, i.e., $\|Z^T \nabla \psi_\varepsilon(x, \mu)\|$, is smaller than this tolerance, and it is in a global state, otherwise. Fundamental to the approach is the following quadratic problem:

$$\begin{aligned} \min \quad & \frac{1}{2} h^T H h + \nabla \psi_\varepsilon(x, \mu)^T h \\ \text{s.t.} \quad & \nabla c_i(x)^T h = 0, \quad i \in AC(x, \varepsilon), \end{aligned} \quad (14)$$

where,

$$H = \nabla^2 \psi_\varepsilon(x, \mu) - \sum_{i \in AC(x, \varepsilon)} \lambda_i \nabla^2 c_i(x), \quad (15)$$

the λ_i are the Lagrange multipliers associated with (14) in a local state (in the proximity of a stationary point) and the λ_i are taken to be zero when the algorithm is in a global state (far from a stationary point). In practice, the QR decomposition of $A(x)$ is used to solve the quadratic problem:

$$A(x) = Q \begin{bmatrix} R \\ 0 \end{bmatrix} = [Y \quad Z] \begin{bmatrix} R \\ 0 \end{bmatrix}. \quad (16)$$

If we set $h = ZW$, for some $w \in R^n$, then w is to be found by solving

$$Z^T H Z w = -Z^T \nabla \psi_\varepsilon(x, \mu). \quad (17)$$

Therefore, for solving the quadratic problem (14), we need an approximation for the projected or reduced Hessian $Z^T H Z$. We are to present a projected structured BFGS update formula for computing an approximation $B_z \approx Z^T H Z$ by providing a quasi-Newton approximation $A_z \approx Z^T S(x, \lambda) Z$, where,

$$S(x, \lambda) = \mu S(x) + \sum_{i \in VE(x, \varepsilon)} \text{sgn}(c_i(x)) \nabla^2 c_i(x) - \sum_{i \in VI(x, \varepsilon)} \nabla^2 c_i(x) - \sum_{i \in AC(x, \varepsilon)} \lambda_i \nabla^2 c_i(x), \quad (18)$$

with the setting $B_z = \mu Z^T G(x) G(x)^T Z + A_z$. We give the asymptotic convergence results of exact penalty methods using this projected structured BFGS updating scheme. Consider the asymptotic case, that is, the case that the final active set has been identified, so that for all further k , with x^* designating the optimal point, we have

$$\begin{aligned} AC(x_k, \varepsilon) &= AC(x^*, 0), \\ VE(x_k, \varepsilon) &= VE(x^*, 0), \\ VI(x_k, \varepsilon) &= VI(x^*, 0). \end{aligned} \quad (19)$$

Suppose that

$$A_{z,k} \approx Z_k^T S(x_k, \lambda^k) Z_k, \quad (20)$$

and we want to update $A_{z,k}$ to $A_{z,k+1}$, approximating

$$A_{z,k+1} \approx Z_{k+1}^T S(x_{k+1}, \lambda^{k+1}) Z_{k+1}. \quad (21)$$

Note that $A_{z,k}, A_{z,k+1} \in R^{n-t, n-t}$, where t is the number of active constraints. We assume $\text{rank}(A(x_k)) = t$, for all k . Letting $s = \bar{Z}^T (\bar{x} - x)$ and $q = \bar{Y}^T (\bar{x} - x)$, we have $\bar{x} - x = \bar{Y}q + \bar{Z}s$, where, in order to simplify notation, the presence of a bar above a quantity indicates that it is taken at iteration $k+1$, and the absence of a bar indicates iteration k . If the constraints are linear, then $q = 0$, for $k > 0$. In the nonlinear case, asymptotically it is expected that $\bar{Y}q$ be negligible and thus we obtain (Mahdavi-Amiri and Bartels, 1989):

$$\begin{aligned} \bar{Z}^T S(\bar{x}, \bar{\lambda}) \bar{Z} s &\approx \bar{Z}^T [\mu(\bar{G} - G) \bar{F} + (\bar{E} - E) \bar{\sigma} \\ &\quad - (\bar{I} - I)e + A\lambda] = y. \end{aligned} \quad (22)$$

Thus, we use the quasi-Newton formula to update A_z to \bar{A}_z according to the secant equation $\bar{A}_z s = y$, when $\bar{Y}q$ has actually become negligible, that is, when

$$\|q\| < \frac{\eta}{(k+1)^{1+\nu}} \|s\|, \quad (23)$$

for $\nu = 0.01$, where k is the iteration number and $\|\cdot\|$ denotes the Euclidean norm, as suggested by Nocedal and Overton (1985) for general constrained nonlinear programs. Therefore, an approximation \bar{B}_z to $\bar{Z}^T \bar{H} \bar{Z}$ is

$$\bar{B}_z = \bar{A}_z + Q(\bar{x}), \quad (24)$$

where,

$$Q(\bar{x}) = \mu \bar{Z}^T G(\bar{x}) G(\bar{x})^T \bar{Z}. \quad (25)$$

A general framework for exact penalty algorithm to minimize (5) is given below (see Mahdavi-Amiri and Bartels, 1989).

Algorithm 1: An Exact Penalty Method.

Step 0: Give an initial point $x, \varepsilon > 0$ and $t > 0$.

Step 1: Determine $AC(x, \varepsilon)$, $VE(x, \varepsilon)$, $VI(x, \varepsilon)$, $A(x)$. Identify the matrix Z by computing the QR decomposition of $A(x)$ and set `global=true`, `optimal=false`.

Step 2: If $\|Z^T \nabla \psi_\varepsilon(x, \mu)\| > \tau$, then obtain the search direction d that is a solution of the quadratic problem (14), where, H approximates the global Hessian $\nabla^2 \psi_\varepsilon(x, \mu)$, and go to Step 5.

Step 3: Determine the Lagrange multipliers λ_i , $i \in AC(x, \varepsilon)$, as a solution of

$$\min \|A(x)\lambda - \nabla \psi_\varepsilon(x, \mu)\|_2. \quad (26)$$

If conditions (13) are not satisfied, then choose an index r for which one of (13) is violated, and determine the search direction d that satisfies the system of equations $A(x)^T d = \sigma_r e_r$, where, e_r is the r th unit vector, and $\sigma_r = -\text{sgn}(\lambda_r)$, and go to Step 5.

Step 4: Set `global=false`. Determine the direction $d = h_A + v$, where, h_A is the solution to the quadratic problem (14), H approximates the local Hessian $\nabla^2 \psi_\varepsilon(x, \mu) - \sum_{i \in AC(x, \varepsilon)} \lambda_i \nabla^2 c_i(x)$ with the λ_i being the Lagrange multipliers associated with (14) as determined in Step 3. The vertical direction v , is the solution to the system

$$A(x)^T v = -c_{AC(x, \varepsilon)}(x + h_A), \quad (27)$$

where, $c_{AC(x, \varepsilon)}$ is the vector of the constraint functions, ordered in accordance with the columns of $A(x)$. Set the step length $\alpha = 1$ and go to Step 6.

Step 5: Determine step length α using a line search procedure on $\psi(x, \mu)$.

Step 6: Compute $\bar{x} = x + \alpha d$. If a sufficient decrease has been obtained, then set $x = \bar{x}$, else go to Step 8.

Step 7: If `global=false`, then check the optimality conditions for x . If x is optimal, then set `optimal=true` and stop, else go to Step 1.

Step 8: If `global=true` and $AC(x, \varepsilon) \neq AC(x, 0)$, then reduce ε to change $AC(x, \varepsilon)$, else reduce τ so that $\|Z^T \nabla \psi_\varepsilon(x, \mu)\|$ becomes large tested against τ .

Step 9: If (`global=true` and $AC(x, \varepsilon) = AC(x, 0)$) or ε is too small or τ is too small, then report failure and stop, else go to Step 1.

Remarks: In Step 6, we have used an appropriate nonsmooth line search strategy to determine the step length

satisfying a sufficient decrease in ψ that is characterized by the line search assumption; see Coleman and Conn (1982b), part (v), P. 152. For a recent line search strategy for CNLLS problems, see Mahdavi-Amiri and Ansari (2012a, 2012c). In Step 7, the optimality conditions are checked as follows: If $\|Z^T \nabla \psi_0(x, \mu)\|$ is small enough, then determine the multipliers λ_i , $i \in AC(x, 0)$, as the least squares solution of (26). If the conditions (13) are satisfied, then x is considered to satisfy the first order conditions, and thus x being a stationary point, the algorithm stops, as commonly practiced in optimization algorithms. Of course, second order conditions are needed to be checked to ascertain optimality of x .

In the remainder of our work, we drop the index z from A_z and B_z , for simplicity.

For approximating the projected structured Hessians in solving the quadratic problem, we make use of the ideas of Dennis et al. (1989) in a different context. We consider the structured BFGS update of A , given by

$$\begin{aligned} \bar{A} = A + & \frac{(y - As)v^T + v(y - As)^T}{v^T s} \\ & - \frac{(y - As)^T s}{(v^T s)^2} v v^T \end{aligned} \quad (28)$$

where,

$$v = y^S + \sqrt{\frac{s^T y^S}{s^T B^S s}} B^S s,$$

$$y^S = Q(\bar{x})s + y \text{ and } B^S = A + Q(\bar{x}).$$

Remark: Clearly, \bar{A} , given by (32), satisfies the secant equation $\bar{A}s = y$.

Here, we use the BFGS secant updates for approximating the projected structured Hessians in solving the constrained nonlinear least squares problem.

The remainder of our work is organized as follows. In § 2, we give the asymptotic two-step superlinear convergence of the algorithm. Competitive numerical results are reported in § 3. The results are compared with the ones obtained by the three algorithms in the KNITRO software package for solving general nonlinear programs. We conclude in § 4.

2. Local Convergence

Here, we give our local two-step superlinear convergence result. We make the following assumptions.

Assumptions:

(A1) Problem (1) has a local solution x^* . Let $D_1 = \{x: \|x - x^*\| \leq \varepsilon_1\}$.

(A2) $\{x_k\}$, generated by Algorithm 1 for minimizing $\psi(x, \mu)$ is so that $x_k \in D_1, \forall k$.

(A3) The function ϕ and $c_i, i=1, \dots, k+m$, are twice continuously differentiable on the compact set D_1 .

(A4) $\nabla^2 \phi(x)$ and $\nabla^2 c_i(x)$ are locally Lipschitz continuous at x^* .

(A5) $Q(x)$ is locally Lipschitz continuous at x^* , that is, there exist a constant $L > 0$ such that

$$\|Q(x) - Q(x^*)\| \leq L \|x - x^*\|, \forall x \in D_1. \quad (29)$$

(A6) The gradients of the active constraints at x_k , for all k , are linearly independent on D_1 .

(A7) There exist positive constants b_1 and b_2 such that

$$b_1 \|z\|^2 \leq z^T B^* z \leq b_2 \|z\|^2, \forall z \in \mathbb{R}^n \quad (30)$$

where, B^* is the projected Hessian at x^* .

We will make use of the following two theorems from Mahdavi-Amiri and Ansari (2012a) and Nocedal and Overton (1985).

Theorem 1. (Mahdavi-Amiri and Ansari, 2012a) There exists $\varepsilon_2 > 0$ such that if $\|e_k\| \leq \varepsilon_2$ and $\|e_{k+1}\| \leq \varepsilon_2$, then

$$\|y_k - A^* s_k\| \leq C_1 \max(\|e_k\|, \|e_{k+1}\|) \|s_k\| + \|Z^* B^* Y^* q_k\|, \quad (31)$$

where, $e_k = x_k - x^*$, $C_1 \geq 1$ is a constant independent of k , and if equation (8) holds, then

$$\|y_k - A^* s_k\| \leq \gamma_k \|s_k\|, \quad (32)$$

where,

$$\sigma_k \leq \gamma_k \leq C_1 (1 + \eta) \sigma_k + \eta \frac{\|Z^* B^* Y^*\|}{(1+k)^{1+\nu}}, \quad (33)$$

and $\sigma_k = \max(\|e_k\|, \|e_{k+1}\|)$.

Theorem 2. (Nocedal and Overton, 1985) Suppose that Algorithm 1 is applied with any update rule for

approximating the projected Hessian matrix. If $x_k \rightarrow x^*$, $\|B_k^{-1}\| \leq \tau_1$, for all k , and

$$w_k \equiv \frac{\|(B_k - B^*)s_k\|}{\|x_{k+1} - x_k\|} \rightarrow 0, \quad (34)$$

then $x_k \rightarrow x^*$, at a two-step Q-superlinear rate.

We now give the so-called bounded deterioration inequality for the structured BFGS update formula.

Theorem 3. Suppose that the inequality (23) and Assumptions (A5) and (A7) hold. Let B_{k+1} be the projected structured BFGS secant update formula, that is, $B_{k+1} = A_{k+1} + Q(x_{k+1})$, where, A_{k+1} is the BFGS update formula of A_k obtained by using (28). Then, there exist a positive constant α_1 and the neighborhood D_3 such that

$$\|B_{k+1} - B^*\|_{B^*} \leq \|B_k - B^*\|_{B^*} + \alpha_1 \gamma_k, \quad (35)$$

for all $x_k, x_{k+1} \in D_3$.

Now, we give a linear convergence result.

Theorem 4. Suppose that Assumptions (A1)-(A7) hold. Let B_k be the structured BFGS secant update formula, that is, $B_k = A_k + Q(x_k)$, where, A_k is the BFGS update formula of A_{k-1} obtained by using (28). Let the sequence $\{x_k\}$ be generated by Algorithm 1. For any $r \in (0, 1)$, there exist positive constants ε and δ such that if $\|e_0\| \leq \varepsilon$ and $\|B_0 - B^*\|_{B^*} \leq \delta$, then

$$\|e_{k+1}\| \leq r \|e_{k-1}\|, \quad k \geq 1, \quad (36)$$

that is, $\{x_k\}$ converge to x^* , at least at a two-step Q-linear rate.

The next theorem shows the satisfaction of (34) in Theorem 2 for the structured BFGS update formula.

Theorem 5. If $K > 0$ exists so that for every iteration $k > K$, we have

$$\|q_k\| < \frac{\eta}{(k+1)^{1+\nu}} \|s_k\|, \quad (37)$$

and we update the B_k using the structured BFGS secant formula for each $k > K$, that is, $B_k = A_k + Q(x_k)$, where,

A_k is the BFGS update formula of A_{k-1} obtained by using (28), then

$$\lim_{k \rightarrow \infty} \frac{\|(B_k - B^*)s_k\|}{\|x_{k+1} - x_k\|} = 0. \quad (38)$$

In §1, we pointed out that we use the structured quasi-Newton update formula for approximating the projected Hessian in Algorithm 1, if the inequality (23) holds (this is expected to happen when the algorithm is in its local phase with the iterate being close to a stationary point). Now, we give the superlinear convergence result for Algorithm 1.

Theorem 6. Suppose that Assumptions (A1)-(A7) hold. Let the sequence $\{x_k\}$ be generated by Algorithm 1 and B_k be obtained by

$$B_k = \begin{cases} A_k + Q(x_k), & \text{if inequality (23) holds,} \\ B_{k-1}, & \text{otherwise,} \end{cases} \quad (39)$$

where, A_k is the BFGS update formula of A_{k-1} obtained by using (28). Then, $\{x_k\}$ converges to x^* with a two step superlinear rate, that is,

$$\lim_{k \rightarrow \infty} \frac{\|x_{k+1} - x^*\|}{\|x_{k-1} - x^*\|} = 0. \quad (40)$$

3. Numerical Experiments

We coded our algorithm in MATLAB 7.6.0. In the global steps, a line search strategy is necessary. In our implementation, for the line search strategy, we used the approach specially designed for nonlinear least squares given by Mahdavi-Amiri and Ansari (2012a, b, c). We put $\eta = 1$ in (23), as suggested by Nocedal and Overton (1985) and set $\varepsilon = 10^{-2}$ and $\tau = 10^{-1}$. The initial matrix $A_{z,0}$ is set to be the identity matrix. For robustness, we followed the computational considerations provided by Mahdavi-Amiri and Bartels (1989). We tested our algorithm on 35 randomly generated test problems using the test problem generation scheme given by Bartels and Mahdavi-Amiri (1986). A simple test generating scheme in Bartels and Mahdavi-Amiri (1986) is described here. The following parameters are set for the least squares problem:

- n : number of variables;
- l : number of components in $F(x)$;
- m_e : number of equality constraints;
- m_r : number of inequality constraints;

v : number of active inequality constraints.

The random values for the components of an optimizer x^* and the corresponding Lagrange multipliers, the λ_i^* , of the active equality constraints are set between -1 and 1, while the Lagrange multipliers of the active inequality constraints are chosen between 0 and 1 (the Lagrange multipliers of the inactive constraints are set to zero). Then, the Lagrangian Hessian matrix at x^* is determined as follows:

$$L^* = \nabla^2 L(x^*, \lambda^*) = \nabla^2 \phi(x^*) - \sum_{i=1}^{m_e} \lambda_i^* \nabla^2 c_i(x^*) \quad (41)$$

$$- \sum_{i=m_e+1}^{m_r} \lambda_i^* \nabla^2 c_i(x^*) = \begin{bmatrix} \sigma U_1^T U_1 & 0 \\ 0 & U_2^T U_2 \end{bmatrix},$$

where, U_1 is a $(v + m_e) \times (v + m_e)$ upper triangular matrix. Except for $U_1[1,1]$, which is set to 2, all the components of U_1 are random values in the range -1 and 1. U_2 is also an upper triangular matrix with random components being set between -1 and 1. The scalar σ is under user control, and it dictates whether the Hessian of the Lagrangian of the generated problem is positive definite ($\sigma > 0$) or indefinite ($\sigma \leq 0$).

Next, for generating the random least squares problem, we consider $F(x) = [f_1(x), \dots, f_l(x)]^T$, where,

$$f_i(x) = \begin{cases} \sum_{j=1}^n (x_j^2 x_i) - i, & i = 1, \dots, l-2, \\ h_0^T (x - x^*) + 1, & i = l-1; \\ \frac{1}{2} (x - x^*)^T D_0 (x - x^*) + 1, & i = l. \end{cases} \quad (42)$$

The values of h_0 and D_0 are determined such that $\nabla L(x^*, \lambda^*) = 0$ and $\nabla^2 L(x^*, \lambda^*) = L^*$, to satisfy the sufficient conditions for local optimality of x^* . The constraints of the problem are set as follows:

$$c_i(x) = \sum_{j=1}^n (x_j^2 x_i) - i + h_i^T (x - x^*) + \alpha_i, \quad (43)$$

where, the constants α_i are set so that $c_i(x^*)=0$ for the active constraints and $c_i(x^*)=1$ for the inactive constraints.

For setting the h_i , the gradient of c_i at x^* is set to the i th column of the identity matrix, to have the linear independence of the active constraints at hand.

The parameters of these random problems are reported in Table 1. All random numbers needed for the random problems were generated by the function "rand" in MATLAB. We generated 35 random problems, composed of 5 problems in each one of the categories numbered in Table 1 along with the parameter settings. For the generated problem sets 1-3, 4-5 and 6-7, all quantities were exactly the same and only $\nabla^2 L(x^*, \lambda^*)$ differed in each set by having a different value of σ . A variety of problems having different number of variables and number of constraints were used. We generated problems not only having positive definite, but also indefinite Hessians of the Lagrangian (with positive definite projected Hessians).

Table 1: The parameters of random problems.

Problem Number	n	l	m_e	m_r	ν	σ
1	5	5	2	3	2	1
2	5	5	2	3	2	-1
3	5	5	2	3	2	-10
4	10	10	5	5	2	1
5	10	10	5	5	2	-1
6	20	20	8	12	2	1
7	20	20	8	12	2	-1

For comparison, we solved these random problems by the three algorithms in KNITRO 6.0 (Interior-point/Direct, Interior-point/CG and Active set algorithms). In keeping the three algorithms of KNITRO in line with our computing features, we set the parameters 'GradObj' and 'GradConstr' to 'on', so that exact gradients are used, and the other parameters were set to the default parameter values (this way, the BFGS updating rule was used for Hessian approximations).

For our comparisons, we explain the notion of a performance profile (Dolan and More', 2002) as a means to evaluate and compare the performance of the set of solvers S on a test set P . We assume that we have n_s solvers and n_p problems. We are interested in using the number of function evaluations as a performance measure. Suppose $N_{p,s}$ is the number of function evaluations required to solve problem p by algorithm s . We compare the performance on

problem p by solver s with the best performance by any solver on this problem; that is, we use the performance ratio

$$r_{p,s} = \frac{N_{p,s}}{\min_s N_{p,s}}. \quad (44)$$

We assume that a parameter R_m is chosen so that $R_m \geq r_{p,s}$, for all p and s , and we put $r_{p,s} = R_m$ if and only if algorithm s does not solve problem p . If we define

$$\rho_s(\tau) = \frac{1}{n_p} \text{size}\{p \in P : r_{p,s} \leq \tau\}, \quad (45)$$

then $\rho_s(\tau)$ is the probability for solver $s \in S$ that a performance ratio $r_{p,s}$ is within a factor $\tau \in R$ of the best possible ratio. The function ρ_s is the distribution function for the performance ratio. The value of $\rho_s(1)$ is the probability that the solver will win over the rest of the solvers (Dolan and More', 2002). Figure 1 shows the performance profiles of the five solvers. The most significant aspect of Figure 1 is that on this test set our algorithm outperforms all other solvers. The performance profile for our algorithm lies above all the others for all performance ratios. According to Figure 1, we observe that our algorithm has shown to be substantially more efficient more often than the three programs in KNITRO.

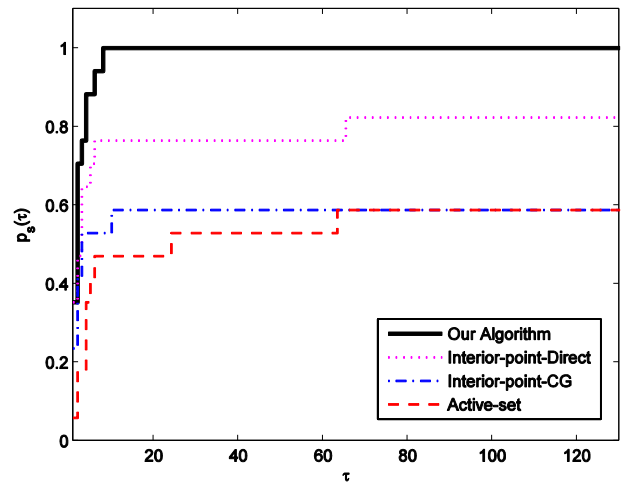


Fig. 1. Performance profile for the algorithms.

4. Conclusion

We proposed a projected structured BFGS scheme for approximating the projected structured Hessian matrix in an

exact penalty method for solving constrained nonlinear least squares problems. We established the local two-step superlinear convergence of the proposed algorithm. Comparative numerical results using the performance profile of Dolan and Moré' showed the efficiency and robustness of the algorithm.

Acknowledgements

The authors thank Iran National Science Foundation (INSF) for supporting this work under grant no. 88000800.

References

- Bartels, R. H., Mahdavi-Amiri, N. (1986). On generating test problems for nonlinear programming algorithms. *SIAM Journal of Scientific and Statistical Computing*, 7(3), pp. 769–798.
- Bidabadi, N., Mahdavi-Amiri, N. (2012). A two-step superlinearly convergent projected structured BFGS method for constrained nonlinear least squares. To appear in *Optimization*.
- Coleman, T. F., Conn, A. R. (1982a). Nonlinear programming via an exact penalty function: Asymptotic analysis. *Math. Program.*, 24, pp. 123–136.
- Coleman, T. F., Conn, A. R. (1982b). Nonlinear programming via an exact penalty function: Global analysis. *Math. Program.*, 24, pp. 137–161.
- Dennis, J. E., Martinez, H. J., Tapia R. A. (1989). Convergence theory for the structured BFGS secant method with an application to nonlinear least squares. *Journal of Optimization Theory and Applications*, 61, pp. 161-178.
- Dolan, E. D., Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Math. Program.*, 91, pp. 201-213.
- Golub, G., Pereyra, V. (2003). Separable nonlinear least squares: the variable projection method and its applications, topical review. *Inverse Problems* 19, pp. R1–R26.
- Fletcher, R., Xu, C. (1987). Hybrid methods for nonlinear least squares. *IMA Journal of Numerical Analysis*, 7, pp. 371-389.
- Li, Z. F., Osborne, M. R., Prvan, T. (2002). Adaptive algorithm for constrained least-squares problems. *Journal of Optimization Theory and Applications*, 114(2), pp. 423-441.
- Mahdavi-Amiri, N., Ansari, M. R. (2012a). Superlinearly convergent exact penalty projected structured Hessian updating schemes for constrained nonlinear least squares: asymptotic analysis. To appear in *Bulletin of the Iranian Mathematical Society*.
- Mahdavi-Amiri, N., Ansari, M. R. (2012b). Superlinearly convergent exact penalty projected structured Hessian updating schemes for constrained nonlinear least squares: global analysis. To appear in *Optimization*.
- Mahdavi-Amiri, N., Ansari, M. R. (2012c). A Superlinearly Convergent Penalty Method with Nonsmooth Line Search for Constrained Nonlinear Least Squares. *SQU Journal for Science*, 17(1), pp. 103-124.
- Mahdavi-Amiri, N., Bartels, R. H. (1989). Constrained nonlinear least squares: An exact penalty approach with projected structured quasi-Newton updates. *ACM Transactions on Mathematical Software*, 15(3), pp. 220–242.
- Mullen, K. M., Vengris, M., van Stokkum, I. H. M. (2007). Algorithms for separable nonlinear least squares with application to modeling time-resolved spectra. *Journal of Global Optimization*, 38, pp. 201–213
- Nievergelt, Y. (2000). A tutorial history of least squares with applications to astronomy and geodesy. *Journal of Computational and Applied Mathematics*, 121, pp. 37-72.
- Nocedal, J., Overton, M. L. (1985). Projected Hessian updating algorithms for nonlinearly constrained optimization. *SIAM Journal of Numerical Analysis*, 22(5), pp. 821–850.