

# A Superlinearly Convergent Penalty Method with Nonsmooth Line Search for Constrained Nonlinear Least Squares

Nezam Mahdavi-Amiri\* and Mohammad Reza Ansari\*\*

Faculty of Mathematical Sciences, Sharif University of Technology, Tehran, Iran,  
\*Email: nezamm@sina.sharif.edu, \*\*Email: ansari.mr@gmail.com.

**ABSTRACT:** Recently, we have presented a projected structured algorithm for solving constrained nonlinear least squares problems, and established its local two-step Q-superlinear convergence. The approach is based on an adaptive structured scheme due to Mahdavi-Amiri and Bartels of the exact penalty method. The structured adaptation also makes use of the ideas of Nocedal and Overton for handling the quasi-Newton updates of projected Hessians and appropriates the structuring scheme of Dennis, Martinez and Tapia. Here, for robustness, we present a specific nonsmooth line search strategy, taking account of the least squares objective. We also discuss the details of our new nonsmooth line search strategy, implementation details of the algorithm, and provide comparative results obtained by the testing of our program and three nonlinear programming codes from KNITRO on test problems (both small and large residuals) from Hock and Schittkowsky, Lukšan and Vlček and some randomly generated ones due to Bartels and Mahdavi-Amiri. The results indeed affirm the practical relevance of our special considerations for the inherent structure of the least squares.

**KEYWORDS:** Constrained nonlinear programming, Exact penalty method, Nonlinear least squares. Nonsmooth line search, Projected structured Hessian update.

طريقة جزاء فائقة التقارب مع خط بحث غير قابل للاشتقاق لمسائل مربعات صغرى غير خطية ومقيدة

نظام مهداوي ومحمد رضا أنصاري

**ملخص:** قدمنا حديثاً خوارزمية مركبة لحل مسائل المربعات الصغرى غير الخطية المقيدة، وأكدنا تقاربها الموضوعي الفائق بخطوتين. وقد تم ذلك بالاستناد إلى اقتباس خطة مهداوي أميري و بارتلز لطريقة الجزاء الدقيقة. ذلك الاقتباس يستند أيضاً على أفكار نوسيدال وأفيرتون في كيفية تصحيح ممتاثلة نيوتن لإسقاطات هس وعلى خواص الخطة التركيبية لدنيس ومارتنيز وتابية. ولتأكيد هذه النتيجة، نعرض هنا خط بحث معين غير قابل للاشتقاق، بالاستناد إلى هدف المربعات الصغرى. كذلك ناقش تفاصيل استراتيجية هذا الخط البحثي، والتفاصيل التنفيذية للخوارزمية، وتجهيز نتائج للمقارنة تم الحصول عليها باختيار برنامجنا مع ثلاثة حالات من برنامج نايترو (KNITRO) للبرمجة غير الخطية على مسائل نموذجية (ذات أخطاء متبقية صغيرة وكبيرة) تنتمي لمسائل هوك-شيتكوسكي ولوكشان-فلك وبعض المسائل المختارة عشوائياً بواسطة بارتلز ومهداوي أميري. تؤكد تلك النتائج العلاقة التطبيقية لافتراضاتنا الخاصة حول تركيبة المربعات الصغرى.

## 1. Introduction

An exact penalty method is a sequential unconstrained minimization approach for solving general nonlinear programming (NLP) problems. Some attractive features of the exact penalty methods are:

- They overcome the difficulties posed by inconsistent constraint linearization (Fletcher, 1987).
- They are successful in solving certain classes of problems in which standard constraint qualifications are not satisfied (Anitescu, 2005; Benson *et al.*, 2006; Leyffer *et al.*, 2006).
- They are useful to enhance the robustness and ensure the feasibility of the subproblems produced in the iterations of the algorithm (Byrd *et al.*, 2004; Gill *et al.*, 2002).

Coleman and Conn (1982a, 1982b, 1984) presented an effective exact penalty algorithm for solving generally constrained nonlinear minimization problems. Mahdavi-Amiri and Bartels (1989) adapted the approach to develop an algorithm using projected structured Hessian for solving constrained least squares problems. Although computational results given in (Mahdavi-Amiri and Bartels, 1989) showed a promising two-step superlinear asymptotic convergence, the authors did not provide an analytical proof. We presented a variant of the approach in (Mahdavi-Amiri and Bartels, 1989) and established both the global and local superlinear convergence (Mahdavi-Amiri and Ansari, 2011a, 2011b). Our algorithm appropriates the ideas of Dennis *et al.* (1989) in the unconstrained case to the projected structured Hessian updating. Here, we present the details of our new structurally adapted nonsmooth line search strategy, implementation details and more extensive computational results.

The remainder of our work is organized as follows. Section 2 gives the notation and relevant background on constrained nonlinear least squares problems, the exact penalty approach and projected structured Hessian updating scheme for constrained nonlinear least squares problems. Section 3 discusses the new nonsmooth line search procedure accounting for the structure in the nonlinear least squares objective. Section 4 is devoted to the global and the local two-step superlinear convergence results. The practical issues relating to implementation are discussed in Section 5. Computational results in Section 6 substantiate the theoretical results and show that the new algorithm is competitive as compared to the best methods tested in the collection of Hock and Schittkowski (1981) and three algorithms developed in KNITRO (Byrd *et al.*, 2006) integrated package. There, we also show the effectiveness of the algorithm on some larger test problems taken from (Lukšan and Vlček, 1999) and randomly generated test problems due to Bartels and Mahdavi-Amiri (1986). Finally, we conclude in Section 7.

Throughout,  $\|\cdot\|$  denotes the  $l_2$  norm for vectors or matrices.

## 2. Constrained nonlinear least squares

The problem to be solved is:

$$\begin{aligned} \min_x \quad & \phi(x) = \frac{1}{2} F(x)^T F(x) \\ \text{s.t.} \quad & c_i(x) = 0, \quad i \in M_1 \\ & c_j(x) \geq 0, \quad j \in M_2, \end{aligned} \tag{1}$$

where  $F(x) = [f_1(x), \dots, f_l(x)]^T$ ,  $x$  is an  $n$ -vector, and for each  $f_\delta$ ,  $c_i$  and  $c_j$  are functions from  $\mathbf{R}^n$  to  $\mathbf{R}^1$ , all assumed to be twice continuously differentiable. We refer to each  $f_\delta$  as a *residual*. There are good reasons for numerical analysts to study least squares problems, mainly because of their applicability to many practical problems. Almost anyone who formulates a parameterized model for a chemical, physical, financial, social or

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

economic application uses a problem of the form (1) to select values for the parameters that best match the model to the data. Although the problem (1) can be solved by a general constrained nonlinear optimization method, in most circumstances the inherent structure of the objective function in (1) makes it worthwhile to use specialized techniques. Notice that if  $G(x)$  is the matrix with its columns being the gradients  $\nabla f_\delta(x)$ , then the gradient of  $\phi(x)$  is:

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

and the Hessian of  $\phi(x)$  is:

$$\nabla^2\phi(x) = G(x)G(x)^T + \sum_{\delta=1}^l f_\delta(x)\nabla^2 f_\delta(x) := G(x)G(x)^T + S(x). \quad (2)$$

In many applications, it is possible to calculate the first partial derivatives that make up the matrix  $G(x)$  explicitly. These could be used to calculate the gradient  $\nabla\phi(x)$ . However, the distinctive feature of a least squares problem is that by knowing  $G(x)$ , we can compute the first part of the Hessian  $\nabla^2\phi(x)$  for free. Moreover, the term  $G(x)G(x)^T$  is often more important than the second summation term in (2), either because of near-linearity of the model near the solution (that is, the  $\nabla^2 f_\delta(x)$  being small) or because of small residuals (that is, the  $f_\delta$  being small). Most algorithms for unconstrained nonlinear least squares exploit these structural properties of the Hessian. Newton's method uses the search direction  $p$  found by solving,

$$\left(G(x)G(x)^T + S(x)\right)p = -G(x)F(x). \quad (3)$$

For problems where,  $\|S(x)\|$  is small compared to  $\|G(x)G(x)^T\|$  as  $x$  approaches a minimizer  $x^*$ , the term  $S(x)$  is dropped in (3) to yield the Gauss-Newton method. This method works well when  $G(x)$  has full row rank and  $\|S(x)\|$  is "sufficiently small", for discussions see (Dennis, 1977; Gill and Murray, 1978).

Approximation for  $S(x)$  in cases where it cannot be neglected is a recurring theme in the literature. In the unconstrained case, quasi-Newton approximations to only the second term,  $S(x)$ , of the Hessian matrix (2) have been developed (Dennis *et al.*, 1989; Dennis and Walker, 1981; Engels and Martinez, 1991). Pertinent to our proposal is the work by Dennis *et al.* (1981), who made investigations of quasi-Newton updates to a matrix  $B \approx S(x)$ . This additive structure was analyzed by Dennis *et al.* (1989), was used by Al-Baali and Fletcher (1985) and was extended to an infinite dimensional setting by Griewank (1987). These strategies are called "*structured quasi-Newton*" methods. Tapia (1988) extended the class of secant updates to equality constrained optimization utilizing the structure present in the Hessian of the augmented Lagrangian. Local Q-superlinear convergence of these structured secant methods were established by Dennis and Walker (1981). Based on Tapia's approach, Huschens (1993) also presented an approach to exploit the nonlinear least squares structure. A hybrid method, linking the reduced successive quadratic programming (SQP) method with the ideas of Fletcher and Xu (1987), was proposed by Tjoa and Biegler (1991). Mahdavi-Amiri and Bartels (1989) adjoined the reduced SQP method with the ideas developed by Dennis *et al.* (1981) for the unconstrained structured quasi-Newton BFGS update to develop a projected structured approach for solving the problem (1).

## 2.1 An exact penalty function

Here  $\psi(x, \mu)$  denotes the  $\ell_1$ -exact penalty function,

$$\psi(x, \mu) = \mu\phi(x) + \sum_{i \in M_1} |c_i(x)| - \sum_{j \in M_2} \min(0, c_j(x)),$$

where the penalty parameter  $\mu$  is a positive number. It is well known that for appropriate values of the penalty parameter  $\mu$ , stationary points of  $\psi(x, \mu)$  are either KKT points of the nonlinear program (1) or infeasible stationary points (Byrd *et al.*, 2005). Pietrzykowski (1969) showed this penalty function to be *exact* in the sense that if  $x^*$  is an isolated minimizer of (1) and the gradients of the *active (binding) constraints* at  $x^*$  are linearly independent, then there exists a real number  $\mu^* > 0$  such that  $x^*$  is also an isolated local minimizer of  $\psi(x, \mu)$ , for each  $\mu, 0 < \mu \leq \mu^*$ . Luenberger (1970) determined, in the convex case, the threshold value of  $\mu$ .

The parameter  $\varepsilon \geq 0$  is used to determine the closeness of any constraint to zero, and hence to judge its activity. Using the approach of Mahdavi-Amiri and Bartels (1989), we define the index sets  $AC(x, \varepsilon) = \{i \in M_1 \cup M_2 \mid |c_i(x)| \leq \varepsilon\}$  of *active constraints*,  $AE(x, \varepsilon) = AC(x, \varepsilon) \cap M_1$  of *active equality constraints*,  $AI(x, \varepsilon) = AC(x, \varepsilon) \cap M_2$  of *active inequality constraints*,  $VE(x, \varepsilon) = M_1 - AE(x, \varepsilon)$  of *violated equality constraints*, and  $VI(x, \varepsilon) = M_2 - AI(x, \varepsilon)$  of *violated inequality constraints*. Thus, the minimization of  $\psi$  is carried out with the aid of an  $\varepsilon$ -active merit function:

$$\psi_\varepsilon(x, \mu) = \mu\phi(x) + \sum_{i \in VE(x, \varepsilon)} \text{sgn}(c_i(x))c_i(x) - \sum_{j \in VI(x, \varepsilon)} c_j(x).$$

This provides a differentiable approximation to the true merit function  $\psi$  in a neighborhood of  $x$ . It is trivial that  $\psi_\varepsilon(x, \mu)$  is equal in value to  $\psi(x, \mu)$  when  $\varepsilon = 0$ . Loosely speaking, step directions are determined using  $\psi_\varepsilon$ , but line searches and optimality tests use  $\psi$ .

A necessary condition for  $x^*$  being an isolated local minimizer of  $\psi$  under the assumptions made above on  $\phi$ , the  $c_i$ , and the  $c_j$  is that there exist *multipliers*  $\lambda_r^*$ , for  $r \in AC(x^*, 0)$ , such that

$$\nabla \psi_0(x^*, \mu) = \sum_{r \in AC(x^*, 0)} \lambda_r^* \nabla c_r(x^*) = A(x^*) \lambda^*, \quad (4)$$

and

$$-1 < \lambda_r^* < 1, \quad r \in AE(x^*, 0), \quad (5)$$

$$0 < \lambda_r^* < 1, \quad r \in AI(x^*, 0), \quad (6)$$

where  $A(x)$  is the matrix with its columns being the gradients of the active constraints at  $x$ , for more details see (Coleman and Conn, 1980). A point  $x$  for which only (4) above is satisfied is said to be a *stationary point* of  $\psi$ . If  $x^*$  is an isolated minimizer, then it is necessary for  $x^*$  to be a stationary point and satisfy (5) and (6). Note that stationarity and optimality are determined using  $\psi_0$ , which is  $\psi_\varepsilon$  with  $\varepsilon = 0$ .

We estimate  $\lambda_r$  of the numbers  $\lambda_r^*$  only in the neighborhoods of stationary points. In such neighborhoods, the numbers  $\lambda_r$  are taken to be the least squares solution to:

$$\min_{\lambda} \|A(x)\lambda - \nabla \psi_\varepsilon(x, \mu)\|, \quad (7)$$

and in practice, the QR decomposition of  $A(x)$  is used to solve the least squares problem:

$$A(x) = Q \begin{bmatrix} R \\ 0 \end{bmatrix} = \begin{bmatrix} Y & Z \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix}. \quad (8)$$

If the columns of  $A(x)$  are linearly independent, then the columns of  $Y$  and  $Z$ , respectively, serve as bases for the range space of  $A(x)$  and null space of  $A(x)^T$ ; that is,  $A(x) = YR$ , with  $R$  nonsingular,  $A(x)^T Z = 0$ ,  $Y^T Y$  and  $Z^T Z$  are the identities. Nearness to a stationary point is governed by the *stationary tolerance*  $\tau > 0$ . The  $\lambda_r$  are computed only if the projected gradient,  $Z Z^T \nabla \psi_\varepsilon(x, \mu)$ , is deemed "small enough" according to this tolerance.

## 2.2 The quadratic subproblem

Fundamental to the Mahdavi-Amiri and Bartels (1989) approach is a particular unconstrained quadratic problem:

$$\min_w (Z^T \nabla \psi_\varepsilon(x, \mu))^T w + \frac{1}{2} w^T H_z w. \quad (9)$$

The matrix  $H_z$  is a positive definite approximation of one of two projected Hessian matrices,

$$Z^T (\nabla^2 \psi_\varepsilon(x, \mu) - \sum_{r \in AC(x, \varepsilon)} \lambda_r \nabla^2 c_r(x)) Z,$$

as explained next. When the algorithm is in its "local" state, that is in the proximity of a stationary point, the dual variables  $\lambda$  are computed and in its "global" state, that is far from a stationary point, the value of the  $\lambda_r$  are taken to be zero. For general nonlinear programming problems, a variety of projected quasi-Newton updates is available. For the general constrained nonlinear least squares problems, we make use of the ideas in (Dennis *et al.*, 1989), in the unconstrained case, for structuring and in (Nocedal and Overton, 1985) for projecting.

## 2.3 The step directions

The minimization of  $\psi$  is carried out using several alternative step directions derived from  $\psi_\varepsilon$ :

- (1) the global horizontal step direction;
- (2) the dropping step direction;
- (3) the Newton step direction:
  - (3a) the asymptotic horizontal step direction;
  - (3b) the vertical step direction.

The global horizontal step direction is  $h_G := Z w$  where  $w$  is the solution of the quadratic problem (9). The matrix  $H_z$ , of course, approximates the global projected Hessian.

The asymptotic horizontal step direction  $h_A$  is a component of the Newton step direction,  $h_A + v$ . The Newton steps are only attempted in close neighborhoods of stationary points that are expected to be minimizers. The step direction  $h_A$  is computed as in the global case above. In this case, the matrix  $H_z$  is to be an approximation of the local projected Hessian.

At  $x + h_A$ , the constraints of  $AC(x, \varepsilon)$  may no longer be within  $\varepsilon$  of zero. The vertical step is derived by taking a single Newton step toward the value of  $v$  that solves the system of equations,

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

where  $C_{AC(x, \varepsilon)}$  is the vector of the active constraint functions, ordered in accordance with the columns of  $A(x)$ .

The dropping step direction tries to *drop* a function  $c_i$  from equalities or  $c_j$  from inequalities of the collection of active constraints and provides a direction that gives a local first-order decrease in the penalty function value. It is the step direction  $d$  that satisfies the system of equations,

$$A(x)^T d = -\text{sgn}(\lambda_r) e_r ,$$

where  $r \in AC(x, \varepsilon)$  is chosen for whichever one of (5) or (6) is violated, and  $e_r$  is the  $r$ th unit vector.

The steps described above are used, broadly speaking, as follows:

- (1) When  $\|ZZ^T \nabla \psi_\varepsilon(x, \mu)\| > \tau$ , then we set  $\bar{x} \leftarrow x + \alpha h_G$ , where a line search is used to determine  $\alpha > 0$ .
- (2) When  $\|ZZ^T \nabla \psi_\varepsilon(x, \mu)\| \leq \tau$ , then the multipliers  $\lambda_r, r \in AC(x, \varepsilon)$ , are approximated by the least squares solution of (7).
  - (a) If (5) or (6) is not satisfied, then an index  $r \in AC(x, \varepsilon)$  is chosen for whichever one of (5) or (6) is violated, and we set  $\bar{x} \leftarrow x + \alpha d$ , where a line search is used to determine  $\alpha > 0$ .
  - (b) If (5) and (6) are satisfied, then we set  $\bar{x} \leftarrow x + h_A + v$  (no line search is used for  $h_A$  or  $v$ ).

Under the standing assumptions on the  $f_\delta, c_i$ , and  $c_j$ , the steps incurred by the  $\varepsilon$ -approximation to the penalty function produce descent for the penalty function if  $\varepsilon$  and  $\tau$  are "correctly set". Algorithm 1 below gives an outline of the classical  $\ell_1$ -penalty method.

**Algorithm 1.** *Classical  $\ell_1$ -penalty method.*

Give  $\mu_0 > 0$  and starting point  $x_s^0$ ;

**for**  $k = 0, 1, 2, \dots$

find minimizer  $x^k$  of  $\psi(x, \mu)$ , starting at  $x_s^k$ ;

**if**  $x^k$  is feasible **then STOP**

**else**

choose new  $\mu_{k+1}$  so that  $\mu_{k+1} < \mu_k$  (say,  $\mu_{k+1} = \mu_k / 8$ );

set a new starting point  $x_s^{k+1}$  (say,  $x^k$ );

**end if**;

**endfor.**

Note that care must be taken to detect infeasibility or unboundedness. Also, when the value of  $\mu$  is too large, in which case the optimal point for  $\psi$  may be infeasible, Algorithm 1 reduces  $\mu$  and the minimization of  $\psi$  is repeated, for more details see the first two algorithms presented by Mahdavi-Amiri and Bartels (1989). The details of how  $B_z$  is to be managed will be given in Section 2.4.

## 2.4 The projected least squares structure

For computing the steps using (9), we intend to provide a quasi-Newton approximation  $H_z$  for

$$Z^T (\nabla^2 \psi_\varepsilon(x, \mu) - \sum_{r \in AC(x, \varepsilon)} \lambda_r \nabla^2 c_r(x)) Z = Z^T [\mu G(x) G(x)^T + S(x, \lambda)] Z ,$$

where

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

$$S(x, \lambda) = \mu \sum_{\delta=1}^l f_{\delta}(x) \nabla^2 f_{\delta}(x) + \sum_{i \in VE(x, \varepsilon)} \text{sgn}(c_i(x)) \nabla^2 c_i(x) \\ - \sum_{j \in VI(x, \varepsilon)} \nabla^2 c_j(x) - \sum_{r \in AC(x, \varepsilon)} \lambda_r \nabla^2 c_r(x).$$

In the rest of our work, we set  $\bar{Q}_z(x) \equiv Z(x)^T G(x) G(x)^T Z(x)$  and in order to simplify the 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  $k$ . Thus, we have

$$H_z = \mu \bar{Q}_z(x) + B_z,$$

and intend to provide a quasi-Newton approximation,

$$B_z \approx Z^T S(x, \lambda) Z.$$

We interpret the multipliers  $\lambda_r$  to be zero or estimated according to (7), whichever is appropriate at the  $k$ th iteration of the algorithm. Following (Mahdavi-Amiri and Bartels, 1989), the matrix  $B_z$  is updated such that the new matrix  $\bar{B}_z$  satisfies the secant condition

$$\bar{B}_z s = y,$$

where

$$s = \bar{Z}^T (\bar{x} - x), \quad (10)$$

and  $y$  is an appropriate approximation to  $Z^T S(x, \lambda) Z s$ . Based on the structure principle given by Dennis *et al.* (1989), Engels and Martinez (1991) derived the structured Broyden family. According to this, we presented a scheme (Mahdavi-Amiri and Ansari, 2011a) for updating the matrix  $B_z$  and derived a BFGS  $B_z$ -update scheme as

$$\bar{B}_z = B_z - \frac{H'_z s (H'_z s)^T}{s^T H'_z s} + \frac{u u^T}{u^T s}, \quad (11)$$

and a DFP update formula as

$$\bar{B}_z = B_z + \frac{(z - H'_z s) z^T + z (z - H'_z s)^T}{s^T z} - \frac{s^T (z - H'_z s)}{(s^T z)^2} z z^T, \quad (12)$$

where

$$u = \mu \bar{Q}_z s + y, \text{ and } H'_z = \mu \bar{Q}_z + B_z.$$

Later, we use the BFGS scheme for testing the problems, because of its generally satisfactory numerical behavior in other contexts. We also impose positive definiteness of  $\bar{H}_z$ , if necessary, to guarantee descent, by use of the modified Cholesky factorization.

### 3. The line search strategy

Here, we propose a nonsmooth line search strategy that makes use of the structure of least squares and is specially designed for exact  $\ell_1$ -penalty function  $\psi(x, \mu)$ . In a close neighborhood of  $x^*$  (a point satisfying second-order sufficiency conditions), a step size of unity is always taken. Therefore, the procedure to be explained here is to be used when 'far' from a solution.

Let  $x$  be the current iterate and  $h$  be the direction of search along which the step  $\alpha$  is to be taken. We are interested in finding  $\alpha > 0$  such that a sufficient reduction in  $\psi$  is imposed, that is,

$$\psi(x + \alpha h, \mu) < (\psi(x, \mu) - \delta),$$

where  $\delta$  is a given small positive constant to be specified by the following assumption.

**Assumption 3.1** If  $h$  is a descent direction for  $\psi$  at  $x$ , then  $\alpha$  is determined so that

$$\psi(x, \mu) - \psi(x + \alpha h, \mu) \geq \gamma_1 (h^T g)^2 := \delta, \quad \gamma_1 > 0, \quad (13)$$

where

$$g = \nabla \psi_\varepsilon(x, \mu) + \sum_{i \in \tilde{V}E(x, \varepsilon)} \text{sgn}(c_i(x)) \nabla c_i - \sum_{i \in \tilde{V}I(x, \varepsilon)} \nabla c_i,$$

and

$$\begin{aligned} \tilde{V}E(x, \varepsilon) &= \left\{ i \in AE(x, \varepsilon) \mid \left| \nabla c_i^T h \right| > 0 \right\}, \\ \tilde{V}I(x, \varepsilon) &= \left\{ i \in AI(x, \varepsilon) \mid \nabla c_i^T h < 0 \right\}. \end{aligned}$$

We note that Assumption 3.1 has also been used by Coleman and Conn (1980). It is a direct extension of a result of Proposition 1 of Conn and Pietrzykowski (1977), to show that the above condition can be satisfied.

For a function  $p(x)$ , let  $p(\alpha) = p(x + \alpha h)$ . It should always be clear from the context whether we are thinking of  $p$ , as a function of a vector or a step length value  $\alpha$ . Also, we denote the derivative of  $p(\alpha)$  along  $h$ , by

$$p'(\alpha) = \nabla p(x + \alpha h)^T h$$

and the left and right derivatives of  $p(\alpha)$ , by  $p'_-(\alpha)$  and  $p'_+(\alpha)$ , respectively, by

$$p'_-(\alpha) = \lim_{t \rightarrow 0^-} \frac{p(\alpha + t) - p(\alpha)}{t}, \quad \text{and} \quad p'_+(\alpha) = \lim_{t \rightarrow 0^+} \frac{p(\alpha + t) - p(\alpha)}{t}.$$

Then

$$\psi(\alpha) = \mu \phi(\alpha) + \sum_{i \in M_1} |c_i(\alpha)| - \sum_{j \in M_2} \min(0, c_j(\alpha)).$$

A point where the derivative of  $\psi(\alpha)$  does not exist will be referred to as a breakpoint. Note that the breakpoints occur at the zeros of the  $c_i(\alpha)$ . Since  $\psi(\alpha)$  is a continuous function and bounded below, a minimizer  $\alpha^*$  exists and a minimum occurs either at a breakpoint, where  $\psi'_-(\alpha^*) \leq 0$  and  $\psi'_+(\alpha^*) > 0$  or at a point where  $\psi'(\alpha^*) = 0$ .

We gain our motivation from the case when the  $f_\delta$  and the  $c_i$  are taken to be linear. A similar idea was suggested by Coleman and Conn (1982b) for the general nonlinear programming problems, but our approach is different in our consideration of the objective function. They make a linear approximation to the objective function and constraints but, in fact, we make a quadratic (sum of squares of linear functions) approximation of the objective function and a linear approximation of the constraints. Clearly, this should provide a better approximation. In the new scheme, we try out a sequence of candidate values  $\alpha$ , stopping to accept one of these values when certain conditions are satisfied. The numerical results show the new strategy to do better than the general line search in (Murray and Overton, 1978) considered by Mahdavi-Amiri and Bartels (1989).

Suppose that  $h$  is a descent direction for  $\psi$  so that  $\psi'_+(x)^T h < 0$ . As an approximation of  $\psi(\alpha)$ , we define

$$\tilde{\psi}(\alpha) = \mu \tilde{\phi}(\alpha) + \sum_{i \in M_1} \left| c_i(x) + \alpha \nabla c_i(x)^T h \right| + \sum_{j \in M_2} \min(0, c_j(x) + \alpha \nabla c_j(x)^T h),$$

where

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

$$\tilde{\phi}(\alpha) = \sum_{\delta=1}^l (f_{\delta}(x) + \alpha \nabla f_{\delta}(x)^T h)^2.$$

Note that the breakpoints of  $\tilde{\psi}$  occur at zeros of the  $c_i(x) + \alpha \nabla c_i(x)^T h$ ,  $i \in M_1 \cup M_2$ . We can compute  $\tilde{\psi}'_-(\alpha)$ ,  $\tilde{\psi}'_+(\alpha)$  or  $\tilde{\psi}'(\alpha)$ , for all  $\alpha$  and without any new objective or constraint function evaluation, if needed. Let  $\{x^j = x + \gamma_j h \mid \gamma_j > 0, j = 1, \dots, r\}$  be a sorted list (in nonincreasing order with respect to  $\gamma_j$ ) of these breakpoints along  $h$ . Now, an algorithm to determine the minimizer of  $\tilde{\psi}(\alpha)$  is straightforward: by examining breakpoints sequentially, we find either a breakpoint as a minimizer (that is a breakpoint, at which  $\tilde{\psi}'_- \leq 0$  and  $\tilde{\psi}'_+ > 0$ ) or determine an interval (say,  $(\gamma_{j-1}, \gamma_j)$  or  $(\gamma_r, \infty)$ ) on which  $\tilde{\psi}'$  is continuous, containing the minimizer that is found by setting  $\tilde{\psi}'(\alpha) = 0$ . Next, we explain this strategy in more detail.

We consider moving along the positive direction of  $h$  as long as the corresponding directional derivative of  $\tilde{\psi}$  along  $h$  is non-positive (in doing this, we may encounter breakpoints). Later, we discuss how to compute  $\tilde{\psi}'_-$ ,  $\tilde{\psi}'_+$  and  $\tilde{\psi}'$ . Now, suppose we are along the interval  $[\gamma_{i-1}, \gamma_i]$ ,  $\tilde{\psi}'_+(\gamma_{i-1}) \leq 0$  and a minimizer of  $\tilde{\psi}$  is yet to be determined. We compute  $\tilde{\psi}'_-(\gamma_i)$ . If  $\tilde{\psi}'_-(\gamma_i) > 0$ , then the minimizer is in  $(\gamma_{i-1}, \gamma_i)$  and we determine the step length  $\alpha$  so that  $\tilde{\psi}'(\alpha) = 0$ , for  $\alpha \in (\gamma_{i-1}, \gamma_i)$ . But, if  $\tilde{\psi}'_-(\gamma_i) \leq 0$ , then we compute  $\tilde{\psi}'_+(\gamma_i)$ . Now, if  $\tilde{\psi}'_+(\gamma_i) > 0$ , then clearly  $\gamma_i$  is the desired step length and the breakpoint  $x^i = x + \gamma_i h$  is a minimizer of  $\tilde{\psi}$  along  $h$ . In cases where  $\tilde{\psi}'_+(\gamma_i) \leq 0$ , we expect more reduction along  $h$ , and thus the next interval  $[\gamma_i, \gamma_{i+1}]$  is determined and the process is repeated. If we progress through the entire sorted list of breakpoints and  $\tilde{\psi}$  continues to reduce past the last breakpoint  $x^r = x + \gamma_r h$ , then the step length  $\alpha$  is determined by setting  $\tilde{\psi}'(\alpha) = 0$ , for  $\alpha \in (\gamma_r, \infty)$ .

Now, we discuss the computational details. We compute breakpoints of  $\tilde{\psi}$  by

$$\gamma_i = -c_{j_i}(x) / c'_{j_i}(x) > 0.$$

For computing the derivative and directional derivative of  $\tilde{\psi}$ , we note that for any  $\gamma > 0$ , we have

$$\tilde{\phi}'(x + \gamma h) - \tilde{\phi}'(x) = \gamma h^T G G^T h.$$

Inductively, consider we are in the interval  $[\gamma_{i-1}, \gamma_i)$  and  $\tilde{\psi}'_+(\gamma_{i-1})$  is computed. Along this interval, derivatives of the linearized constraints would remain unchanged. Therefore, the change in  $\tilde{\psi}'$  is determined by the change in  $\tilde{\phi}'$  only. Considering the above and  $x + \gamma h = x^{i-1} + (\gamma - \gamma_{i-1})h$ , for  $\gamma \in (\gamma_{i-1}, \gamma_i)$ , we have

$$\tilde{\psi}'(\gamma) = \psi'_+(\gamma_{i-1}) + \mu(\gamma - \gamma_{i-1})h^T G G^T h, \quad \forall \gamma \in (\gamma_{i-1}, \gamma_i), \quad (14)$$

and

$$\tilde{\psi}'_-(\gamma_i) = \lim_{\gamma \rightarrow \gamma_i^-} \tilde{\psi}'(\gamma) = \tilde{\psi}'_+(\gamma_{i-1}) + \mu(\gamma_i - \gamma_{i-1})h^T G G^T h.$$

Now, suppose  $\gamma_i$  corresponds to  $c_{j_i}$  and

$$\tilde{c}(\alpha) = \begin{cases} \left| c_{j_i}(x) + \alpha \nabla c_{j_i}(x)^T h \right|, & \text{if } j_i \in M_1, \\ -\min(0, c_{j_i}(x) + \alpha \nabla c_{j_i}(x)^T h), & \text{if } j_i \in M_2. \end{cases}$$

We have (by the fact that  $c_{j_i}(x) + \gamma_i \nabla c_{j_i}(x)^T h = 0$ ),

$$\tilde{c}'_+(\gamma_i) = \tilde{c}'_-(\gamma_i) - \rho \operatorname{sgn}(c_{j_i}(x)) c'_{j_i}(x),$$

where

$$\rho = \begin{cases} 2, & \text{if } j_i \in M_1, \\ 1, & \text{if } j_i \in M_2. \end{cases} \quad (15)$$

Since at  $\gamma_i$  the change in directional derivative of  $\tilde{\psi}$  is determined by the change in directional derivative of  $\tilde{c}$  only, we have

$$\begin{aligned} \tilde{\psi}'_+(\gamma_i) &= \tilde{\psi}'_-(\gamma_i) + \tilde{c}'_+(\gamma_i) - \tilde{c}'_-(\gamma_i) \\ &= \tilde{\psi}'_-(\gamma_i) - \rho \operatorname{sgn}(c_{j_i}(x)) c'_{j_i}(x), \end{aligned}$$

where  $\rho$  is as defined by (15). Finally, by (14), the solution of  $\tilde{\psi}'(\alpha) = 0$ , for  $\alpha \in (\gamma_{i-1}, \gamma_i)$ , is:

$$\tilde{\psi}'(\alpha) = \tilde{\psi}'_+(\gamma_{i-1}) + \mu(\alpha - \gamma_{i-1})h^T GG^T h = 0,$$

and therefore,

$$\alpha = \gamma_{i-1} - \frac{\tilde{\psi}'_+(\gamma_{i-1})}{\mu h^T GG^T h}.$$

It is clear that if  $G^T h = 0$ , then  $\tilde{\psi}'$  is constant along each interval and therefore we are certain that the minimizer is a breakpoint.

**Algorithm 2.** *Line search procedure.*

**Input:**  $x$ , the current point,  $h$ , a descent direction for  $\psi$  at  $x$ ,  $\varepsilon_1 > 0$  and  $\delta > 0$ .

**Output:** The step length  $\alpha$  such that  $\psi(x + \alpha h, \mu) < (\psi(x, \mu) - \delta)$ , where,  $\delta$  is a given small positive constant specified by (13).

$\alpha_l := 0$ ;

$x^1 := x$ ;

**while** ( $\psi' < 0$ );

{Determine  $\alpha$ , the minimizer of  $\tilde{\psi}(x^1 + \alpha h, \mu)$ };

**if** ( $\alpha_l = 0$ ) **then**

$\alpha := \min(1.0, \alpha)$ ;

**else**

$\alpha := \alpha_l + \max(\varepsilon_1, \alpha)$

**endif**

$x^1 := x + \alpha h$ ;

**if**  $\psi(x^1, \mu) < (\psi(x, \mu) - \delta)$  **then**

return  $\alpha$ ;

**endif**

$\alpha_l := \alpha$ ;

$\psi' := \nabla \psi(x^1, \mu)^T h$ ;

**endwhile**;

{Perform the (Murray and Overton, 1978) line search with the initial step length  $\alpha$ };

return  $\alpha$ ;

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

Note that for  $\psi(x, \mu)$ , the resulting point  $x + \alpha h$  may not necessarily satisfy the sufficient decrease. In this situation, if  $\psi'(\alpha) < 0$ , then we repeat the above procedure with  $x$  replaced by  $x + \alpha h$ . On the other hand, if  $\psi'(\alpha) \geq 0$ , then we have a good interval containing the minimizer and thus the approach of Murray and Overton (1978) can be performed conveniently using their successive polynomial interpolation with safeguard strategy (using the available information at the last point  $x$  and  $x + \alpha h$ , their algorithm works out more favorably; for details, see (Murray and Overton, 1978)). Our numerical testing showed that an initial large value of  $\alpha$  may harm the performance of the line search. Thus, for the first trial, we adjust the choice of  $\alpha$  by setting  $\alpha = \min(1, \alpha)$ , to ensure that the Newton-like step is tried (if it gives enough reduction). For subsequent values of  $\alpha$ , when the first trial does not provide a sufficient decrease specified by (13), we adjust  $\alpha$  by at least an  $\varepsilon_1 > 0$  to ensure enough change in the value of  $\alpha$  and admit a finite termination of the algorithm.

We are now ready to give our line search algorithm. In Algorithm 2, the parameter  $\varepsilon_1$  (we used  $\varepsilon_1 = 10^{-3}$  in our experiments in Section 6) is used to avoid acceptance of a small correction to  $\alpha$ . Our experiments showed that the usage of this parameter would accelerate the procedure. The parameter  $\delta$  is used to determine a sufficient decrease in  $\psi$  and is characterized by (line search) Assumption 3.1 as specified by (3.1).

It should be pointed out that our numerical experiments in Section 6 showed that Algorithm 2 often achieved adequate reductions inside the loop without needing to perform the Murray and Overton line search.

### 4. Global and local convergence results

Here, we briefly state the global and the local two-step Q-superlinear convergence results. The analysis of global convergence is based on the results of Byrd and Nocedal (1991), who studied convergence of the Coleman and Conn (1984) approach, with our necessary adjustments for the nonlinear least squares. Let  $D$  denote a compact set so that  $x^k \in D$  in  $D$ , for all  $k$ . Let  $C_{AC}(x)$  denote the vector of active constraints at  $x$  and  $\hat{S}$  denote the set of first-order points of  $\psi$  in  $D$ , see (Coleman and Conn, 1980) for justification of this terminology.

**Assumptions 4.1** Let  $D$  be a compact set.

- (A)  $\phi$  and  $c_i, i \in M_1 \cup M_2$ , are twice continuously differentiable and their first and second derivatives are uniformly bounded in norm on  $D$ ;
- (B)  $\{x^k\}$  is generated by Algorithm 1 starting from an arbitrary initial point, and  $x^k \in D$ , for all  $k$ ;
- (C) the gradients of the active constraints at any point in  $D$  are linearly independent.

**Theorem 4.2** Suppose that Assumptions 4.1 hold and assume that

- (i) the number of stationary points of  $\psi$  in  $D$  is finite;
- (ii) all first-order points of  $\psi$  in  $D$  are strict second-order points of  $\psi$ ;
- (iii) if  $\{x^{k_i}\}$  is a subsequence and  $\bar{x}$  is a first-order point such that

$$x^{k_i} \rightarrow \bar{x} \text{ and } AC(x^{k_i}, \varepsilon_{k_i}) = AC(\bar{x}, 0),$$

then there exists a sufficiently small positive constant  $\Delta$  such that

$$\left\| (B_{z, k_i} - Z_{k_i}^T S(x^{k_i}, \lambda_{k_i}) Z_{k_i}) s_{k_i} \right\| \leq \Delta \|s_{k_i}\|,$$

where  $s_k$  is defined by (10);

(iv) the proposed line search strategy produces a step length  $\alpha$  satisfying Assumption 3.1.

Then,

- (1)  $\varepsilon_k \rightarrow 0$ .
- (2)  $x^k \rightarrow \bar{x} \in \hat{S}$ .
- (3) For  $k$  sufficiently large,  $AC(x^k, \varepsilon_k) = AC(\bar{x}, 0)$ .
- (4) For  $k$  sufficiently large, the "Newton step" is executed.

**Proof.** See Theorem 3.4 in (Mahdavi-Amiri and Ansari, 2011b).

**Remark 4.3** It should be noted that a similar result for the general nonlinear case (see Coleman and Conn, 1982b, Theorem 1) makes use of a stronger condition,  $H_{z,ki} \rightarrow \bar{Z}^T G_L(\bar{x}) \bar{Z}$  as  $x^{ki} \rightarrow \bar{x}$ , where  $G_L$  is the Hessian of Lagrangian function, while Theorem 4.2 is proved under a weaker condition as specified by (iii). Considering Theorem 4.13 of Mahdavi-Amiri and Ansari (2011a), our condition in (iii) is reasonable.

Now, we state a local two-step superlinear convergence of our method. The asymptotic analysis is developed adapting the approach of Nocedal and Overton (1985) to the nonlinear least squares. We will denote  $x^*$  to be a local optimal solution of problem (1) and suppose the active set at  $x^*$  to be  $\{1, \dots, t\}$ . Furthermore, we assume that we are sufficiently close to  $x^*$  so that the proper active set has been identified. We need the following assumptions.

**Assumptions 4.4** Assume that Assumptions 4.1 hold, and that  $Y(x)$  and  $Z(x)$ , given by a particular implementation of (8), are Lipschitz continuous in  $D$ . We assume that  $\nabla^2 \phi$ ,  $\nabla^2 c_i$ ,  $i \in AC$ , and  $Q_z(x)$  are Lipschitz continuous matrix functions.

**Theorem 4.5** Suppose that Assumptions 4.4 hold. Let the matrix  $B_{z,k}$  be updated by (12). Then, the sequence  $\{x^k\}$  generated by Algorithm 1 converges to  $x^*$  with a local two-step Q-superlinear rate.

**Proof.** See Theorem 4.17 in (Mahdavi-Amiri and Ansari, 2011a).

## 5. Computational considerations

Here, we indicate the measures taken to implement Algorithm 1. The reader is referred to (Mahdavi-Amiri and Bartels, 1989) for more details.

The test for nearness to stationarity has the form,

$$\left\| \bar{Z}^T \nabla \psi_\varepsilon(x, \mu) \right\| \leq \text{reference}_1(\tau, \|\nabla \psi_\varepsilon(x, \mu)\|), \quad (16)$$

where, for given scalars  $\zeta$  and  $val$ , we define  $\text{reference}_1$  to be a composite absolute-relative test,

$$\text{reference}_1(\zeta, val) = \zeta^2 + \zeta val.$$

This avoids underflow, or a stringent test, in either case that  $val$  becomes far from or close to zero. A much smaller tolerance,  $\theta$ , of course, replaces  $\tau$  in (16) when the tests for convergence are made.

By the same token, activity is determined by,

$$|c_r(x)| \leq \text{reference}_2(\varepsilon, c_r(x)),$$

and the violation of an equality constraint is determined by the reverse of this inequality. If  $c_r(x)$  defines an inequality constraint, it is regarded as violated if we have

$$c_r(x) < -reference_2(\varepsilon, c_r(x)).$$

For any scalar  $\zeta$ , we define

$$reference_2(\zeta, c_r(x)) = reference_1\left(\zeta, \frac{\|F(x)\| + \sum_{i \in M_1 \cup M_2} |c_i(x)|}{|M_1 \cup M_2| + 1}\right),$$

in which the second term serves us as a general-purpose average function value.

Tests for feasibility are carried out exactly as for activity, except that a much smaller user-defined tolerance,  $\gamma$ , must be used in place of  $\varepsilon$ .

The main difficulty in the context of exact penalty method is the strategy for choosing and updating the penalty parameter  $\mu$ . The initial value of  $\mu = 1$  was used for most test problems, though we sometimes found that other values were more advantageous to use. These are noted in the reported results. Only naive changes are made to  $\mu$  during the running of the program. The value of  $\mu$  is reduced through division by 8 each time a minimizer  $x$  for  $\psi(x, \mu)$  is found that is not feasible for (1). Recently, Byrd *et al.* (2008) proposed an effective penalty parameter update strategy appropriated for trust region approaches. Of course, a possible adaptation to line search strategies is welcome. We have work in progress to consider a new penalty parameter updating strategy in the context of combined line search and trust region exact penalty methods.

The value of  $\lambda_r$  must be tested to determine whether it satisfies (5) or (6). We carried out this relative to a boundary tolerance  $\theta$  in analogy with the corresponding ones in (Mahdavi-Amiri and Bartels, 1989).

The value of  $\mu$  is regarded as being too small when

$$\mu \|F(x)\| \leq reference_2(mascheps, c_r(x)),$$

where *mascheps* is the unit round-off error. We regard  $\varepsilon$  to be too small when  $\varepsilon \leq \gamma$ , and  $\tau$  is regarded as being too small when  $\tau \leq \theta$ .

The difference in penalty function values,  $\psi(\bar{x}, \mu) - \psi(x, \mu)$ , has to be tested to determine whether a sufficient decrease has been obtained. We choose the condition for a sufficient decrease on the Newton step to be:

$$\psi(x + h_A + v, \mu) - \psi(x, \mu) \leq -reference_1(\beta, \|\bar{Z}^T \nabla \psi_\varepsilon(x, \mu)\|^2 + \|C_{AC}(x, \varepsilon)\|_1).$$

A sufficient decrease for the other two steps (global and dropping) is demanded by setting the parameter  $\gamma_1$  in the line search condition (13). The numerical positive definiteness of the matrix  $H_z$  is enforced, if necessary, by using the modified Cholesky factorization described in (Fang and O'Leary, 2008) during the process of solving (9) to guarantee descent.

The requirements for optimality are that, for reasonable choices of reference values,

- (1)  $\|Z^T \nabla \psi_\varepsilon(x, \mu)\| \leq reference_1(\theta, \|\nabla \psi_\varepsilon(x, \mu)\|),$
- (2a)  $-1 + \theta \leq \lambda_i \leq +1 - \theta, \quad \forall i \in AE(x, \varepsilon),$
- (2b)  $\theta \leq \lambda_j \leq +1 - \theta, \quad \forall j \in AI(x, \varepsilon),$
- (3)  $|\psi_\varepsilon(\bar{x}, \mu) - \psi_\varepsilon(x, \mu)| \leq reference_1(\gamma, |\psi_\varepsilon(\bar{x}, \mu)|),$
- (4)  $\|\bar{x} - x\| \leq reference_1(\theta, \|\bar{x}\|).$

One might also be prepared to accept as a case of optimal termination the event that one or more of the above fail to be satisfied, but the user should be notified whenever this happens, for details see (Mahdavi-Amiri and Bartels, 1989).

## 6. Computational results

Here, we report the comparative results obtained on three sets of problems. First, thirty CNLLS problems were taken from Hock and Schittkowski (1981): 1, 2, 6, 13, 14, 16, 17, 18, 20, 22, 23, 26, 27, 28, 30, 31, 32, 42, 46, 48, 49, 50, 51, 52, 53, 60, 65, 77 and 79. These were chosen because they were least squares problems or could be conveniently cast in that form. The number of variables in these problems varies from 1 to 5, and the number of constraints varies from 1 to 13. Problems 28, 48, 50, 51, 52, and 53 have both linear constraints and linear residual functions and the other problems have at least a constraint or a residual function to be nonlinear. Second, we considered seven problems generated from (Bartels and Mahdavi-Amiri, 1986) who presented a program for random generation of general nonlinearly constrained nonlinear programming problems and also generated an example of a least squares problem where the dimensions and the curvature of the problem could be varied. These random test problems have both nonlinear constraints and residual functions. Third, we used twelve larger test problems in the constrained nonlinear least squares form from (Lukšan and Vlček, 1999) as listed below:

1. Chained Rosenbrock function with trigonometric-exponential constraints.
2. Chained Wood function with Broyden banded constraints.
3. Chained Powell singular function with simplified trigonometric-exponential constraints.
4. Chained Cragg-Levy function with tridiagonal constraints.
5. Chained HS46 (Hock and Schittkowski, 1981), problem number 46) problem.
6. Chained HS47 problem.
7. Chained modified HS48 problem.
8. Chained modified HS49 problem.
9. Chained modified HS50 problem.
10. Chained modified HS51 problem.
11. Chained modified HS52 problem.
12. Chained modified HS53 problem.

These problems were modified in such a way that a variable number of  $n$  may be used. In our tests we take  $n \approx 300$  (we were not able to consider larger  $n$ , since the free KNITRO package we are using is limited to problems with  $n = 300$ ). In all of these test problems the constraints and the residual functions are both nonlinear. The algorithm has successfully solved all of the considered problems quite efficiently. The local convergence on all problems clearly showed a two-step superlinear rate. Moreover, the significant reduction in the number of global steps confirms the effectiveness of our proposed line search procedure (Algorithm 2), as compared to the results obtained by Mahdavi-Amiri and Bartels (1989) as well as methods reported by Hock and Schittkowski (1981) and results obtained by our testing of three algorithms (Interior/Direct, Interior/CG and Active-Set) recently developed in the KNITRO 6.0 package. Our calls to programs in KNITRO are managed by a C++ program to have the same testing environment as our program.

Our algorithm was coded in a portable subset of C++ and ran in double precision arithmetic on the Microsoft visual C++ 6.0 compiler on an x86-based PC with an AMD~1667Mhz processor. The parameters of Sections 2 and 5 were set as follows:

$$\eta = 1, \tau = 10^{-1}, \varepsilon = 10^{-2}, \gamma = 10^{-14}, \gamma_1 = 10^{-4}, \theta = 10^{-12}, \text{ and } \beta = 10^{-6}.$$

These values were determined by experience as ones that have worked well on most of the problems we had tried.

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

Tables 1 and 2 present the results on problems from (Hock and Schittkowski, 1981). The same initial points as in this reference were used. The function value obtained by our algorithm is found in the column headed by "FV" for comparison with those listed. It should be noted that our objective functions correspond to those reported in (Hock and Schittkowski, 1981) multiplied by 1/2. This was simply a matter of convenience for us in arranging the test problems in a least squares format. The number given in the "NI" column counts the cumulative number of steps through the for loop of the minimization for  $\psi$ . The "NAI" column counts the number of steps needed to be taken for termination after the identification of the correct active set at the solution and execution of the Newton steps. The "NFE" column counts the number of times the value of the objective function was computed. The spread between the NI and the NFE provides an impression of how much work was required by the line search. Indeed, the spread was not found to be significant and this points to the efficiency of our proposed line search scheme. The numbers listed in the column headed by "H&SH" give the number of function evaluations required by the best method tested in (Hock and Schittkowski, 1981) for attaining a function value at least as good as the one we attained. We list the number of function evaluations needed by the method of Mahdavi-Amiri and Bartels (1989) in the column headed by "MA&B". For the three algorithms provided by KNITRO 6.0 (Byrd *et al.*, 2006) integrated package for general constrained nonlinear problems, we used the default parameter values without defining any special case for the objective function or constraints, and used the dense quasi-Newton BFGS Hessian approximation of the Hessians.

Table 1. Initializing with the zero matrix

PN	FV	NI	NAI	NFE	H&SH	MA&B	D	CG	AS
1	7.45045E-24	9	3	10	24	21	44	87	70
2	2.15624E-02	8	2	8	40	6	15	19	11
6 <sup>(1)</sup>	7.54965E-30	5	2	7	10	6	13	14	17
13 <sup>(2)</sup>	5.07423E-01	20	5	24	45	29	47	15	65
14	6.96732E-01	4	1	5	6	8	47	32	44
15 <sup>(3)</sup>	1.53250E+02	4	1	5	5	3	13	15	7
16 <sup>(3)</sup>	1.15723E+00	6	2	8	89	19	10	18	5
17	5.00000E-01	9	1	12	12	20	9	15	17
18	2.50000E+00	13	4	19	8	36	13	12	19
20 <sup>(3)</sup>	2.00994E+01	7	1	10	17	46	8	5	4
22	5.00000E-01	4	1	5	9	9	7	7	5
23	1.00000E+00	5	3	6	7	7	9	8	8
26 <sup>(1)</sup>	2.01655E-23	5	2	6	19	37	29	27	104
27	2.00000E-02	13	3	18	25	23	3588	22	26
28	5.23853E-32	2	1	2	5	2	11	9	10
30	5.00000E-01	2	1	3	14	2	9	7	21
31	3.00000E+00	9	5	11	10	15	13	10	22
32	5.00000E-01	4	1	3	3	9	8	9	4
42	6.92893E+00	7	3	8	10	14	13	10	18
46	5.84019E-09	16	9	16	14	40	44	32	94
48	5.54668E-32	2	1	2	7	2	11	12	28
49	1.35973E-07	11	6	11	9	11	23	26	26
50	1.70714E-30	2	1	2	18	8	9	8	10
51	3.69779E-32	2	1	2	5	2	13	7	13
52	2.66332E+00	3	1	3	8	7	7	9	10
53	2.04651E+00	3	1	3	8	8	9	9	8
60 <sup>(1)</sup>	1.62841E-02	12	3	14	9	29	11	16	100
65 <sup>(4)</sup>	4.76764E-01	11	3	13	-	20	11	19	21
77 <sup>(4)</sup>	1.20753E-01	15	4	17	16	39	39	19	25
79	3.93884E-02	12	4	15	10	29	13	12	17

Table 2. Initializing with the identity matrix

PN	FV	NI	NAI	NFE	H&SH	MA&B	D	CG	AS
1	9.59790E-18	9	4	10	24	29	44	87	70
2	2.15624E-02	7	2	7	40	8	15	19	11
6 <sup>(1)</sup>	4.48518E-23	8	2	11	10	13	13	14	17
13 <sup>(2)</sup>	5.03425E-01	22	6	28	45	30	47	15	65
14	6.96732E-01	4	1	5	6	8	47	32	44
15 <sup>(3)</sup>	1.53250E+02	3	1	4	5	3	13	15	7
16 <sup>(3)</sup>	1.15723E+00	6	2	8	89	9	10	8	5
17	5.00000E-01	9	1	12	12	20	9	5	17
18	2.50000E+00	13	4	19	8	32	13	12	19
20 <sup>(3)</sup>	2.00994E+01	7	2	10	17	32	8	5	4
22	5.00000E-01	4	1	5	9	11	7	7	5
23	1.00000E+00	5	3	6	7	34	9	8	8
26 <sup>(1)</sup>	2.41009E-24	7	4	8	19	15	29	27	104
27	2.00000E-02	14	8	16	25	18	3588	22	26
28	7.70372E-33	3	2	3	5	4	11	9	10
30	5.00000E-01	3	2	3	14	3	9	7	21
31	3.00000E+00	9	5	11	10	15	13	10	22
32	5.00000E-01	5	1	6	3	11	8	9	4
42	6.92893E+00	12	4	22	10	25	13	10	18
46	5.41352E-08	20	10	22	14	27	44	32	94
48	0.00000E-00	5	3	4	7	5	11	12	28
49	2.25804E-07	13	7	13	9	14	23	26	26
50	1.63935E-30	5	1	5	18	8	9	8	10
51	1.23260E-32	3	1	3	5	4	13	7	13
52	2.66332E+00	5	3	5	8	10	7	9	10
53	2.04651E+00	4	3	4	8	10	9	9	8
60 <sup>(1)</sup>	1.62841E-02	12	4	14	9	18	11	16	100
65 <sup>(4)</sup>	4.76764E-01	9	2	11	-	19	11	19	21
77 <sup>(4)</sup>	1.20753E-01	20	8	22	16	33	39	19	25
79	3.93884E-02	15	7	19	10	29	13	12	17

In Tables 1 and 2, the columns headed by "D", "CG" and "AS" respectively give the number of function evaluations required by Interior/Direct (barrier), Interior/CG (barrier) and Active-Set algorithms in this package. Our method of counting function evaluations was consistent with (Hock and Schittkowski, 1981; Mahdavi-Amiri and Bartels, 1989; and KNITRO), so that these columns serve as a basis for comparison. Obviously, the function evaluations are compared on cases where the algorithms have converged to the same solution point. The results for initialization of the projected matrix with the zero matrix are given in Table 1 and those for initialization with the identity matrix are given in Table 2. There is no entry in the "H&SH" column for problem 65, which serves to indicate that our program reached a minimum value significantly lower than that of any of the methods reported on in (Hock and Schittkowski, 1981). In the first column of Tables 1 and 2, the superscript (1) or (4) indicates that the initial value of  $\mu$  was set to 100 or 10, respectively, rather than 1. The problems in question, i.e. 6, 26, 60, 65, and 77, had constraint function values at some points whose magnitudes dominated the magnitude of the penalty function, forcing the minimization process to take an unusually long time in attempting to maintain the constraints within feasibility. The larger value of  $\mu$  compensated for the imbalance in scale between the objective function and the constraints. The superscript (2) or (3) indicates that the initial value of

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

$\mu$  was set to 0.01 or 0.001, respectively, instead of 1. In the problems for which this was done, i.e. 13, 15, 16, and 20, there was a reverse imbalance of scale. The magnitude of the objective function at some point was far larger than that of the constraints. This caused the minimization to waste time in finding a sequence of infeasible stationary points for a sequence of decreasing values of  $\mu$ . The smaller value of  $\mu$  compensated for the imbalance in scale between the objective function and the constraints.

It should be pointed out that we have work in progress to come up with a new automatic penalty parameter updating strategy in the context of combined line search and trust region exact penalty methods. Despite this, the method we have described shows itself competitive in comparison with those tested by Hock and Schittkowski (1981) and those solved by KNITRO. It did, respectively, as well as, better than and poorer than the best of the methods in Hock and Schittkowski on approximately 40%, 50% and 10% of the test problems and in comparison with the method of Mahdavi-Amiri and Bartels (1989), the corresponding results are 30%, 70% and 0%, respectively. In comparison with Interior/Direct algorithm, our algorithm did as well as, better and poorer on 13%, 70% and 17%, respectively. In comparison with Interior/CG algorithm, the corresponding results are 13%, 74% and 13%, respectively. Finally, in comparison with Active-Set algorithm, results are 37%, 50% and 13%, respectively. Thus, our new code here with the modifications proposed along with the new line search strategy significantly outperforms the ones reported in (Bartels and Mahdavi-Amiri, 1986; Mahdavi-Amiri and Bartels, 1989), as well as the three general nonlinear algorithms of KNITRO (Byrd *et al.*, 2006), on the test problems. This should justify the appropriateness of our structural approach to nonlinear least squares.

Table 3. Results on randomly generated test problems initializing with the zero matrix

PN	SP	$\sigma$	FV	NI	NAI	NFE	NCE	RFE
1	1	1	21.6541	19	5	19	24	E-12
1	10	1	21.6541	37	6	40	46	E-12
1	100	1	21.6541	39	7	40	47	E-13
2	1	-1	21.6541	24	6	25	31	E-12
2	10	-1	21.6541	39	7	45	52	E-11
2	100	-1	21.6541	41	6	42	48	E-11
3	1	-10	21.6541	33	7	37	44	E-12
3	10	-10	21.6541	39	7	45	52	E-11
3	100	-10	21.6541	36	7	37	44	E-12
4	1	1	99.3233	35	4	36	40	E-15
4	10	1	98.8173	51	4	56	60	E-14
5	1	-1	98.8173	51	4	55	59	E-16
5	10	-1	98.8173	115	8	117	125	E-15
6	1	1	862.628	105	3	113	116	E-15
6	10	1	862.628	116	4	117	121	E-16
7	1	-1	861.698	125	3	130	133	E-14
7	10	-1	861.698	140	5	152	155	E-14

In Tables 3 and 4, we report the results obtained by our program on three sets of random problems (problems 1, 2 and 3 as set 1, problems 4 and 5 as set 2, and finally problems 6 and 7 as set 3) generated and tested by Bartels and Mahdavi-Amiri (1986). Each set of random problems are generated with all quantities ( $x^*$ , Lagrange multipliers,  $U_1$ , etc.) being exactly the same, and only having different  $\nabla^2 L$  obtained by setting different values for  $\sigma$  ( $\nabla^2 L$  is definite if  $\sigma > 0$  and is indefinite if  $\sigma \leq 0$ , for details see (Bartels and Mahdavi-Amiri, 1986). The numbers of the equality constraints, the inequality constraints and the active constraints at solution for the three sets respectively are (2, 3, 4), (5, 5, 7) and (8, 12, 10). The results obtained by

each algorithm provided by KNITRO 6.0 on these random problems are separately reported in Table 5. In Tables 3 and 4, the "SP" column shows the value to which all components of the starting point are set. The " $\sigma$ " column shows the value considered for  $\sigma$ . The "NCE" column gives the count for the number of times the constraint function is computed. We compute the composite absolute-relative feasibility error at the final point  $x$  as:

$$rfe(x) = \frac{\sum_{i \in M_1} |c_i(x)| - \sum_{j \in M_2} \min(0, c_j(x))}{1 + \|C(x)\|_1},$$

where  $C(x)$  denotes the vector of constraints evaluated at  $x$ . Rounded value of  $rfe(x)$  is shown in the column headed by "RFE". The other column headlines have the same meanings as those given for Tables 1 and 2.

Table 4. Results on randomly generated test problems initializing with the identity matrix

PN	SP	$\sigma$	FV	NI	NAI	NFE	NCE	RFE
1	1	1	21.6541	25	5	25	30	E-10
1	10	1	21.6541	45	6	47	53	E-12
1	100	1	21.6541	37	6	38	44	E-12
2	1	-1	21.6541	27	6	30	36	E-11
2	10	-1	21.6541	56	7	56	63	E-11
2	100	-1	21.6541	45	6	45	51	E-12
3	1	-10	21.6541	28	7	31	38	E-11
3	10	-10	21.6541	45	7	52	59	E-12
3	100	-10	21.6541	44	7	45	52	E-11
4	1	1	99.3233	38	6	40	46	E-15
4	10	1	98.8173	52	4	58	62	E-15
5	1	-1	98.8961	30	5	31	36	E-15
5	10	-1	98.9681	89	10	95	105	E-15
6	1	1	862.628	132	4	138	142	E-15
6	10	1	862.628	163	3	173	176	E-15
7	1	-1	861.698	152	7	157	164	E-15
7	10	-1	861.698	138	3	146	149	E-14

In Table 5 for each algorithm of KNITRO we have three columns, and all column headlines have the same meanings as those given for other tables.

Note that on problems 4 (with starting point 10), 5 (with starting point 1), 6 and 7 our program reached a locally optimal solution having significantly smaller objective values than the values obtained by the other three algorithms. At these solutions, our program terminated with the active constraints set  $\{1, \dots, 7, 9\}$  on problems 4 and 5, and with the active constraints set  $\{1, \dots, 15, 17, 18, 19\}$  on problems 6 and 7.

In Table 6, we report the results obtained by our program and three programs of KNITRO 6.0 on problems taken from (Lukšan and Vlček, 1999). The same initial points as in this reference were used. The columns headed by " $n$ " and " $m$ " respectively give the number of variables and constraints. The superscript (in the first column) and the other column headlines have the same meanings as those given for Table 1. The Interior/Direct program, on problems 2, 3, 4, 7 and 9 reached the iteration limit (of 10000) or failed to solve the problem and on problem 8 it found a local optimal solution having objective value 1.67453E+02, which is significantly smaller than the values obtained by the other three programs. On problem 3, the Interior/CG program and the Active-Set program terminated with the objective values 3.40390E+02 and 7.08426E+00, respectively. On problem 1, the

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

Active-Set program found a local optimal solution having the objective value 1.92827E+00, which is significantly larger than the values obtained by the other three programs.

The results in Tables 3, 4, 5 and 6 show the effectiveness of the proposed algorithm as compared to the three algorithms recently developed in KNITRO (Byrd *et al.*, 2006) on constrained nonlinear least squares problems. Observing the low values of the number of local iterations under the column "NAI" in all the tables, the local two-step superlinear convergence of our proposed algorithm is well supported. Moreover, the overall performance of the proposed algorithm on the test problems attests to its robustness.

Table 5. Results of KNITRO on randomly generated test problems

			D			CG			AS		
PN	SP	$\sigma$	FV	NFE	RFE	FV	NFE	RFE	FV	NFE	RFE
1	1	1	21.6541	36	E-9	21.6543	45	E-8	21.6541	45	E-9
1	10	1	21.6544	58	E-9	21.6657	62	E-7	21.6542	327	E-10
1	100	1	21.6543	83	E-6	21.6562	77	E-5	22.8154	97	E-18
2	1	-1	21.6541	42	E-8	21.6543	45	E-8	21.6541	45	E-9
2	10	-1	21.6546	64	E-9	21.6565	76	E-8	21.6564	256	E-9
2	100	-1	21.6543	91	E-8	21.7348	72	E-5	21.8106	132	E-15
3	1	-10	21.6541	52	E-9	21.6541	69	E-8	21.6541	54	E-9
3	10	-10	21.6541	67	E-7	21.6665	77	E-7	21.6542	312	E-10
3	100	-10	21.6543	93	E-8	21.6546	87	E-8	21.8070	74	E-18
4	1	1	99.3281	266	E-8	99.3902	163	E-7	99.3281	151	E-11
4	10	1	99.3522	169	E-7	99.3531	130	E-7	99.3307	417	E-13
5	1	-1	99.3305	386	E-7	99.3304	139	E-9	99.3281	193	E-12
5	10	-1	98.8990	182	E-9	99.7139	99	E-8	99.3301	766	E-12
6	1	1	959.052	533	E-7	959.567	3096	E-7	959.327	406	E-12
6	10	1	959.061	8140	E-7	971.849	17787	E-14	959.327	406	E-12
7	1	-1	960.240	13174	E-6	959.517	423	E-10	960.210	127	E-15
7	10	-1	959.406	7835	E-5	959.514	255	E-9	959.040	3813	E-15

Table 6. Results on problems taken from (Lukšan and Vlček, 1999)

PN	$n$	$m$	FV	NI	NAI	NFE	D	CG	AS
1 <sup>(1)</sup>	300	298	1.12048E-14	30	5	40	36	34	31
2 <sup>(3)</sup>	300	290	2.62320E+03	250	9	340	-	186	699
3	300	2	2.00542E+02	27	6	34	-	544	189
4	300	296	7.15723E+02	50	8	65	-	35	180
5 <sup>(1)</sup>	299	198	2.34529E-13	30	4	38	25	23	37
6	297	222	2.22740E+02	35	5	48	25	27	174
7	299	198	1.18060E+03	32	6	39	-	41	80
8	299	198	7.79817E+02	75	8	125	159	183	184
9 <sup>(1)</sup>	297	222	1.67930E-06	115	8	135	-	194	588
10	297	222	5.06731E-14	29	7	38	91	24	10
11 <sup>(3)</sup>	297	222	2.11459E+02	50	9	75	55	56	121
12 <sup>(3)</sup>	297	222	1.77561E+02	40	6	53	19	27	71

**Remark 6.1** To investigate the effectiveness of projected structured updating schemes proposed by Mahdavi-Amiri and Bartels (1989) and (11) proposed here, we used alternative update rules instead of (11) in the local steps and ran the algorithm on all the test problems again. The asymptotic results showed a marginal

outperformance of (11). Thus, the overall clear outperformance of our new code to the one in (Mahdavi-Amiri and Bartels, 1989) (as we mentioned before) should be attributed to our global steps and the use of our new nonsmooth line search strategy. With this observation, it is remarkable that the two-step superlinear convergence of the method in (Mahdavi-Amiri and Bartels, 1989) is yet to be proved.

## 7. Conclusions

We presented a structurally projected exact penalty method adapted for solving constrained nonlinear least squares problems. The projected structure made appropriate use of the ideas of Nocedal and Overton (1985) for general nonlinear programs. Numerical results showed our proposed algorithm to be efficient and reliable. For robustness, we devised a nonsmooth line search procedure accounting for the structure in the nonlinear least squares objective. We implemented the proposed algorithm and tested the program and three nonlinear programming codes from KNITRO on both small and large residual problems from (Hock and Schittkowski, 1981; Lukšan and Vlček, 1999) and some randomly generated ones due to Bartels and Mahdavi-Amiri (1986). The results showed the practical relevance of our special line search and projected considerations for the sum of squares objective.

## 8. Acknowledgment

The first author thanks the Research Council of Sharif University of Technology for supporting this work.

## 9. References

- AL-BAALI, M. and FLETCHER, R., 1985. Variational methods for non-linear least squares. *Journal of the Operational Research Society*, **36**: 405-421.
- ANITESCU, M., 2005. Global convergence of an elastic mode approach for a class of mathematical programs with complementarity constraints. *SIAM J. Optim.*, **16**(1): 120-145.
- BARTELS, R.H. and MAHDAVI-AMIRI, N., 1986. On generating test problems for nonlinear programming algorithms. *SIAM Journal on Scientific and Statistical Computing*, **7**(3): 769-798.
- BENSON, H.Y., SEN, A., SHANNO, D.F. and VANDERBEI, R.J., 2006. Interior-point algorithms, penalty methods and equilibrium problems. *Computational Optimization and Applications*, **34**(2): 155-182.
- BYRD, R.H., GOULD, N.I.M., NOCEDAL, J. and WALTZ, R.A., 2004. An algorithm for nonlinear optimization using linear programming and equality constrained subproblems. *Math. Program. Series B*, **100**(1): 27-48.
- BYRD, R.H. and NOCEDAL, J., 1991. An analysis of reduced Hessian method for constrained optimization. *Mathematical Programming*, **49**: 285-323.
- BYRD, R.H., NOCEDAL, J. and WALTZ, R.A., 2006. KNITRO: An integrated package for nonlinear optimization. In *Large-Scale Nonlinear Optimization*, eds G. Di Pillo and M. Roma, Springer, pp. 35-59.
- BYRD, R.H., NOCEDAL, J. and WALTZ, R.A., 2008. Steering exact penalty methods for nonlinear programming. *Optimization Methods & Software*, **23**(2): 197-213.
- COLEMAN, T.F. and CONN, A.R., 1980. Second-order conditions for an exact penalty function. *Math. Program.*, **19**: 155-177.
- COLEMAN, T.F. and CONN, A.R., 1982a. Nonlinear programming via an exact penalty function: Asymptotic analysis. *Mathematical Programming*, **24**: 123-136.
- COLEMAN, T.F. and CONN, A.R., 1982b. Nonlinear programming via an exact penalty function: Global analysis. *Mathematical Programming*, **24**: 137-161.
- COLEMAN, T.F. and CONN, A.R., 1984. On the local convergence of a quasi-Newton method for the nonlinear programming problem. *SIAM Journal on Numerical Analysis*, **21**: 755-769.

## A SUPERLINEARLY CONVERGENT PENALTY METHOD

- CONN, A.R. and PIETRZYKOWSKI, T., 1977. A penalty function method converging directly to a constrained minimum. *SIAM Journal on Numerical Analysis*, **14**: 348-375.
- DENNIS, Jr., J.E. 1977. Nonlinear least squares and equations. In *The State of the Art of Numerical Analysis*, eds D. Jacobs, Academic Press, Orlando, Florida, USA.
- DENNIS, Jr., J.E., GAY, D.M. and WELSCH, R.E., 1981. An adaptive nonlinear least squares algorithm. *ACM Trans. Math. Soft.*, **7(3)**: 348-368.
- DENNIS, Jr., J.E., MARTINEZ, H.J. and TAPIA, R.A., 1989. A convergence theory for the structured BFGS secant method with an application to nonlinear least squares. *Journal of Optimization Theory and Applications*, **61**: 161-178.
- DENNIS, Jr., J.E. and WALKER, H.F., 1981. Convergence theorems for least-changes secant update methods. *SIAM. J. Numer. Anal.*, **18**: 949-987.
- ENGELS, J.R. and MARTINEZ, H.J., 1991. Local and superlinear convergence for partially known quasi-Newton methods. *SIAM Journal on Optimization*, **1**: 42-56.
- FANG, H.R. and O'LEARY, D.P., 2008. Modified Cholesky algorithm: A catalog with new approaches. *Math. Program.*, **115(2)**: 319-349.
- FLETCHER, R., 1987. *Practical Methods of Optimization*. (2nd Edition) J. Wiley and Sons, Chichester, England.
- FLETCHER, R. and XU, C., 1987. Hybrid methods for nonlinear least squares. *IMA Journal of Numerical Analysis*, **7**: 371-389.
- GILL, P.E. and MURRAY, W., 1978. Algorithms for the solution of the nonlinear least squares problem, *SIAM J. Numer. Anal.*, **15(5)**: 977-992.
- GILL, P.E., MURRAY, W. and SAUNDERS, M.A., 2002. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM J. Optim.*, **12**: 979-1006.
- GRIEWANK, A., 1987. On the iterative solution of differential and integral equation using secant updating techniques. In *The State of the Art in Numerical Analysis*, eds A. Iserles and M.J.D. Powell, Clarendon Press: 299-324.
- HOCK, H. and SCHITTKOWSKI, K., 1981. *Test Examples for Nonlinear Programming Codes*. Lecture Notes in Economic and Mathematical System No.187, Springer-Verlag, New York.
- HUSCHENS, J., 1993. Exploiting additional structure in equality constrained optimization by structured SQP secant algorithms. *Journal of Optimization Theory and Applications*, **77**: 359-382.
- LEYFFER, S., L'OPEZ-CALVA, G. and NOCEDAL, J., 2006. Interior methods for mathematical programs with complementarity constraints. *SIAM Journal on Optimization*, **17(1)**: 52-77.
- LUENBERGER, D.G., 1970. Control problem with kinks. *IEEE Transactions on Automatic Control*, **15**: 570-574.
- LUKŠAN, L. and VLČEK, J., 1999. *Sparse and Partially Separable Test Problems for Unconstrained and Equality Constrained Optimization*. Institute of Computer Science, Academy of Sciences of the Czech Republic, Technical Report 767.
- MAHDAVI-AMIRI, N. and ANSARI, M.R., 2011a. Superlinearly convergent exact penalty projected structured Hessian updating schemes for constrained nonlinear least squares: Asymptotic analysis. *Bulletin of the Iranian Mathematical Society* (accepted).
- MAHDAVI-AMIRI, N. and ANSARI, M.R., 2011b. Superlinearly convergent exact penalty projected structured Hessian updating schemes for constrained nonlinear least squares: Global analysis. Submitted.
- MAHDAVI-AMIRI, N. and 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)**: 220-242.
- MURRAY, W. and OVERTON, M.L., 1978. *Steeplength Algorithms for Minimizing a Class of Nondifferentiable Functions*. STAN-CS-78-679, Stanford University, Stanford, California, USA.
- NOCEDAL, J. and OVERTON, M.L., 1985. Projected Hessian updating algorithms for nonlinearly constrained optimization. *SIAM Journal on Numerical Analysis*, **22(5)**: 821-850.

- PIETRZYKOWSKI, T., 1969. An exact potential method for constrained maxima. *SIAM Journal on Numerical Analysis*, **6**: 299-304.
- TAPIA, R.A., 1988. On secant updates for use in general constrained optimization. *Mathematical Computing*, **51**: 181-203.
- TJOA, I.B. and BIEGLER, L.T., 1991. Simultaneous solution and optimization strategies for parameter estimation of differential-algebraic equation systems. *Industrial Engineering Chemical Research*, **30**: 376-385.
- 

Received 11 May 2011

Accepted 16 October 2011