Institute of Computer Science
Academy of Sciences of the Czech Republic

Trust-region interior-point method for large sparse l_1 optimization

L.Lukšan, C. Matonoha, J. Vlček

Technical report No. 942

November 2005

Pod Vodárenskou věží 2, 182 07 Prague 8 phone: +420 2 688 42 44, fax: +420 2 858 57 89,
e-mail:e-mail:ics@cs.cas.cz

Institute of Computer Science
Academy of Sciences of the Czech Republic

Trust-region interior-point method for large sparse l_1 optimization

L.Lukšan, C. Matonoha, J. Vlček [1]

Technical report No. 942

November 2005

Abstract:

In this paper, we propose an interior-point method for large sparse l_1 optimization. After a short introduction, the complete algorithm is introduced and some implementation details are given. We prove that this algorithm is globally convergent under standard mild assumptions. Thus nonconvex problems can be solved successfully. The results of computational experiments given in this paper confirm efficiency and robustness of the proposed method.

Keywords:
Unconstrained optimization, large-scale optimization, nonsmooth optimization, minimax optimization, interior-point methods, modified Newton methods, computational experiments.

[1]This work was supported by the Grant Agency of the Czech Academy of Sciences, project code IAA1030405, and the institutional research plan No. AV0Z10300504. L.Lukšan is also from the Technical University of Liberec, Hálkova 6, 461 17 Liberec.

1 Introduction

Consider the l_1 optimization problem: Minimize function

$$F(x) = \sum_{i=1}^{m} |f_i(x)|, \tag{1}$$

where $f_i : R^n \to R$, $0 \le i \le m$ (m is usually large), are smooth functions depending on a small number of variables (n_i, say) satisfying either Assumption 1 or Assumption 2.

Assumption 1. Functions $f_i(x)$, $1 \le i \le m$, are twice continuously differentiable on the convex hull of level set $\mathcal{L}(\overline{F}) = \{x \in R^n : F(x) \le \overline{F}\}$ for a sufficiently large upper bound $\overline{F}$ and they have bounded the first and second-order derivatives on conv$\mathcal{L}(\overline{F})$, i.e., constants $\overline{g}$ and $\overline{G}$ exist such that $\|\nabla f_i(x)\| \le \overline{g}$ and $\|\nabla^2 f_i(x)\| \le \overline{G}$ for all $1 \le i \le m$ and $x \in$ conv$\mathcal{L}(\overline{F})$.

Assumption 2 Functions $f_i(x)$, $1 \le i \le m$, are twice continuously differentiable on a sufficiently large convex compact set $\mathcal{D}$.

Since continuous functions attain their maxima on a compact set, Assumption 2 guarantees that constants $\overline{F}$, $\overline{g}$ and $\overline{G}$ exist such that $f_i(x) \le \overline{F}$, $\|g_i(x)\| \le \overline{g}$ and $\|G_i(x)\| \le \overline{G}$ for all $x \in \mathcal{D}$. The choice of $\overline{F}$ and $\mathcal{D}$ will be discussed later (see Assumption 3). Note that set conv$\mathcal{L}(\overline{F})$ used in Assumption 1 need not be compact.

Minimization of F is equivalent to the sparse nonlinear programming problem with $n+m$ variables $x \in R^n$, $z \in R^m$:

$$\text{minimize} \quad \sum_{i=1}^{m} z_i \quad \text{subject to} \quad -z_i \le f_i(x) \le z_i, \quad 1 \le i \le m. \tag{2}$$

Problem (2) can be solved by an arbitrary nonlinear programming method utilizing sparsity (sequential linear programming [7], sequential quadratic programming [10], interior-point [1], [11], [24] and nonsmooth equation [12]). In this paper, we introduce a trust-region interior-point method that utilizes a special structure of the l_1 problem (1). The constrained problem (2) is replaced by a sequence of unconstrained problems

$$\begin{aligned} \text{minimize} \quad B(x, z; \mu) \quad &= \quad \sum_{i=1}^{m} z_i - \mu \sum_{i=1}^{m} \log(z_i - f_i(x)) - \mu \sum_{i=1}^{m} \log(z_i + f_i(x)) \\ &= \quad \sum_{i=1}^{m} z_i - \mu \sum_{i=1}^{m} \log(z_i^2 - f_i^2(x)) \end{aligned} \tag{3}$$

with barrier parameter $\mu > 0$, where we assume that $z_i > |f_i(x)|$, $1 \le i \le m$, and $\mu \to 0$ monotonically. Here $B(x, z; \mu) : R^{n+m} \to R$ is a function of $n+m$ variables $x \in R^n$, $z \in R^m$.

Barrier function (3) remains unchanged if we replace problem (2) by equivalent problem

$$\text{minimize} \quad \sum_{i=1}^{m} z_i \quad \text{subject to} \quad f_i^2(x) \le z_i^2, \quad 1 \le i \le m. \tag{4}$$

The necessary first-order (KKT) conditions for the solution of (4) have the form

$$\sum_{i=1}^{m} 2w_i f_i(x) \nabla f_i(x) = 0, \quad 2w_i z_i = 1, \quad w_i \geq 0, \quad w_i(z_i^2 - f_i^2(x)) = 0, \quad 1 \leq i \leq m, \tag{5}$$

where w_i, $1 \leq i \leq m$, are Lagrange multipliers. Since $z_i = |f_i(x)|$, $1 \leq i \leq m$, at the solution of (4), we can write (5) in the simpler equivalent form

$$\sum_{i=1}^{m} u_i \nabla f_i(x) = 0, \quad \frac{u_i z_i}{f_i(x)} = 1, \quad z_i^2 - f_i^2(x) = 0, \quad 1 \leq i \leq m, \tag{6}$$

where $u_i = 2w_i f_i(x)$ for $1 \leq i \leq m$.

The interior-point method described in this paper is iterative, i.e., it generates a sequence of points $x_k \in R^n$, $k \in N$ (N is the set of integers). For proving the global convergence, we need the following assumption concerning function $F(x)$ and sequence $\{x_k\}_1^\infty$.

Assumption 3 Either Assumption 1 holds and $\{x_k\}_1^\infty \in \mathcal{L}(\overline{F})$ or Assumption 2 holds and $\{x_k\}_1^\infty \in \mathcal{D}$.

The interior-point method investigated in this paper is a trust-region modification of the Newton method. Approximation of the Hessian matrix is computed by the gradient differences which can be carried out efficiently if the Hessian matrix is sparse (see [2]). Since the Hessian matrix need not be positive definite in the non-convex case, the standard line-search realization cannot be used. There are two basic possibilities, either a trust-region approach or the line-search strategy with suitable restarts, which eliminate this insufficiency. We have implemented and tested both these possibilities and our tests have shown that the first possibility, used in Algorithm 1, is more efficient.

The paper is organized as follows. In Section 2, we introduce the interior-point method for large sparse l_1 optimization and describe the corresponding algorithm. Section 3 contains more details concerning this algorithm such as the trust-region strategy and the barrier parameter update. In Section 4 we study theoretical properties of the interior-point method and prove that this method is globally convergent if Assumption 3 holds. Finally, in Section 5 we present results of computational experiments confirming the efficiency of the proposed method.

2 Description of the method

Differentiating $B(x, z; \mu)$ given by (3), we obtain necessary conditions for minimum in the form

$$\sum_{i=1}^{m} \frac{2\mu f_i(x)}{z_i^2 - f_i^2(x)} \nabla f_i(x) \triangleq \sum_{i=1}^{m} u_i(x, z_i; \mu) \nabla f_i(x) = 0 \tag{7}$$

and

$$1 - \frac{2\mu z_i}{z_i^2 - f_i^2(x)} = 1 - u_i(x, z_i; \mu) \frac{z_i}{f_i(x)} = 0, \quad 1 \leq i \leq m. \tag{8}$$

Denoting $g_i(x) = \nabla f_i(x)$, $1 \le i \le m$, $A(x) = [g_1(x), \dots, g_m(x)]$,

$$f(x) = \begin{bmatrix} f_1(x) \\ \dots \\ f_m(x) \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ \dots \\ z_m \end{bmatrix}, \quad u(x, z; \mu) = \begin{bmatrix} u_1(x, z_1; \mu) \\ \dots \\ u_m(x, z_m; \mu) \end{bmatrix} \tag{9}$$

and $Z = \mathrm{diag}(z_1, \dots, z_m)$, we can write (7)–(8) in the form

$$A(x)u(x, z; \mu) = 0, \quad u(x, z; \mu) = Z^{-1} f(x). \tag{10}$$

The system of $n+m$ nonlinear equations (10) can be solved by the Newton method, which uses second-order derivatives. In every step of the Newton method, we solve a set of $n+m$ linear equations to obtain increments Δx and Δz of x and z, respectively. These increments can be used for obtaining new quantities

$$x^+ = x + \alpha \Delta x, \quad z^+ = z + \alpha \Delta z,$$

where $\alpha > 0$ is a suitable step-size. This is a standard way for solving general nonlinear programming problems. For special nonlinear programming problem (2), the structure of $B(x, z; \mu)$ allows us to obtain minimizer $z(x; \mu) \in R$ of function $B(x, z; \mu)$ for a given $x \in R^n$.

Lemma 1. *Function $B(x, z; \mu)$ (with x fixed) has the unique stationary point, which is its global minimizer. This stationary point is characterized by equations*

$$\frac{2\mu z_i(x; \mu)}{z_i^2(x; \mu) - f_i^2(x)} = 1 \quad or \quad z_i^2(x; \mu) - f_i^2(x) = 2\mu z_i(x; \mu), \quad 1 \le i \le m, \tag{11}$$

which have solutions

$$z_i(x; \mu) = \mu + \sqrt{\mu^2 + f_i^2(x)}, \quad 1 \le i \le m. \tag{12}$$

Proof. Function $B(x, z; \mu)$ (with x fixed) is convex for $z_i > |f_i(x)|$, $1 \le i \le m$, since it is a sum of convex functions. Thus if a stationary point of $B(x, z; \mu)$ exists, it is its unique global minimizer. Differentiating $B(x, z; \mu)$ by z (see (8)), we obtain quadratic equations (11), which define its unique stationary point. □

Assuming $z = z(x; \mu)$, we denote

$$B(x; \mu) = \sum_{i=1}^m z_i(x; \mu) - \mu \sum_{i=1}^m \log(z_i^2(x; \mu) - f_i^2(x)) \tag{13}$$

and $u(x; \mu) = u(x, z(x; \mu); \mu)$. In this case, barrier function $B(x; \mu)$ depends only on x. In order to obtain minimizer $(x, z) \in R^{n+m}$ of $B(x, z; \mu)$, it suffices to minimize $B(x; \mu)$ over R^n.

Lemma 2. *Consider barrier function (13). Then*

$$\nabla B(x;\mu) = A(x)u(x;\mu) \tag{14}$$

and

$$\nabla^2 B(x;\mu) = G(x;\mu) + A(x)V(x;\mu)A^T(x), \tag{15}$$

where

$$G(x;\mu) = \sum_{i=1}^m u_i(x;\mu)G_i(x) \tag{16}$$

with $G_i(x) = \nabla^2 f_i(x)$, $1 \le i \le m$, *and* $V(x;\mu) = \mathrm{diag}(v_1(x;\mu),\dots,v_m(x;\mu))$ *with*

$$v_i(x;\mu) = \frac{2\mu}{z_i^2(x;\mu) + f_i^2(x)}, \quad 1 \le i \le m. \tag{17}$$

Proof. Differentiating (13), we obtain

$$\begin{aligned}
\nabla B(x;\mu) &= \sum_{i=1}^m \nabla z_i(x;\mu) - 2\mu \sum_{i=1}^m \frac{z_i(x;\mu)\nabla z_i(x;\mu) - f_i(x)g_i(x)}{z_i^2(x;\mu) - f_i^2(x)} \\
&= \sum_{i=1}^m \left(1 - \frac{2\mu z_i(x;\mu)}{z_i^2(x;\mu) - f_i^2(x)}\right)\nabla z_i(x;\mu) + \sum_{i=1}^m \frac{2\mu f_i(x)g_i(x)}{z_i^2(x;\mu) - f_i^2(x)} \\
&= \sum_{i=1}^m u_i(x;\mu)g_i(x) = A(x)u(x;\mu)
\end{aligned}$$

by (11) and (7). Differentiating (11), one has

$$\frac{\nabla z_i(x;\mu)}{z_i^2(x;\mu) - f_i^2(x)} - \frac{2z_i(x;\mu)(z_i(x;\mu)\nabla z_i(x;\mu) - f_i(x)g_i(x))}{(z_i^2(x;\mu) - f_i^2(x))^2} = 0$$

for $1 \le i \le m$, which gives

$$\nabla z_i(x;\mu) = \frac{2z_i(x;\mu)f_i(x)g_i(x)}{z_i^2(x;\mu) + f_i^2(x)} \tag{18}$$

for $1 \le i \le m$ after arrangements. Thus

$$\begin{aligned}
\nabla u_i(x;\mu) &= \nabla\left(\frac{f_i(x)}{z_i(x;\mu)}\right) = \frac{z_i(x;\mu)g_i(x) - f_i(x)\nabla z_i(x;\mu)}{z_i^2(x;\mu)} \\
&= \left(1 - \frac{2f_i^2(x)}{z_i^2(x;\mu) + f_i^2(x)}\right)\frac{g_i(x)}{z_i(x;\mu)} \\
&= \frac{z_i^2(x;\mu) - f_i^2(x)}{z_i^2(x;\mu) + f_i^2(x)}\,\frac{g_i(x)}{z_i(x;\mu)} \\
&= \frac{2\mu}{z_i^2(x;\mu) + f_i^2(x)}g_i(x) = v_i(x;\mu)g_i(x)
\end{aligned}$$

by (8), (18), (11) and (17). Differentiating (14) and using the previous expression, we obtain

$$\begin{aligned}\nabla^2 B(x;\mu) &= \nabla \sum_{i=1}^m u_i(x;\mu) g_i(x) \\ &= \sum_{i=1}^m u_i(x;\mu) G_i(x) + \sum_{i=1}^m \nabla u_i(x;\mu) g_i^T(x) \\ &= \sum_{i=1}^m u_i(x;\mu) G_i(x) + \sum_{i=1}^m v_i(x;\mu) g_i(x) g_i^T(x),\end{aligned}$$

which is equation (15). □

Lemma 3. *Let vector $d \in R^n$ solve equation*

$$\nabla^2 B(x;\mu) d = -g(x;\mu), \tag{19}$$

where $g(x;\mu) = \nabla B(x;\mu) \neq 0$. If matrix $G(x;\mu)$ is positive definite, then $d^T g(x;\mu) < 0$ (direction vector d is descent for $B(x;\mu)$).

Proof. Equation (19) implies

$$d^T g(x;\mu) = -d^T \nabla^2 B(x;\mu) d = -d^T G(x;\mu) d - d^T A(x) V(x;\mu) A^T(x) d \leq -d^T G(x;\mu) d,$$

since $V(x;\mu)$ is positive definite by (17). Thus $d^T g(x;\mu) < 0$ if $G(x;\mu)$ is positive definite. □

Expression (17) implies that $v_i(x;\mu)$ is bounded if $f_i^2(x)$ is bounded from zero. If $f_i^2(x)$ tends to zero faster than μ then $v_i(x;\mu)$ can tend to infinity and $\nabla^2 B(x;\mu)$ can be ill-conditioned (see (15)). The following lemma gives the upper bound for $\|\nabla^2 B(x;\mu)\|$.

Lemma 4. *If Assumption 3 holds, then*

$$\|\nabla^2 B(x;\mu)\| \leq m(\overline{G} + \overline{g}^2 \|V(x;\mu)\|) \leq m(\overline{G} + \frac{\overline{g}^2}{2\mu}).$$

Proof. Using (15) and Assumption 3, we obtain

$$\begin{aligned}\|\nabla^2 B(x;\mu)\| &\leq \left\| G(x;\mu) + A(x) V(x;\mu) A^T(x) \right\| \\ &\leq \left\| \sum_{i=1}^m u_i(x;\mu) G_i(x) \right\| + \left\| \sum_{i=1}^m v_i(x;\mu) g_i(x) g_i^T(x) \right\| \\ &\leq m\overline{G} + m\overline{g}^2 \|V(x;\mu)\|,\end{aligned}$$

since inequalities $z_i(x;\mu) \geq |f_i(x)|$ and (8) imply that $|u_i| = |f_i(x)|/z_i(x;\mu) \leq 1$ for all $1 \leq i \leq m$. Since $V_\mu(x)$ is diagonal, one has

$$\|V(x;\mu)\| = \max_{1 \leq i \leq m} v_i = \max_{1 \leq i \leq m} \left(\frac{2\mu}{z_i^2(x;\mu) + f_i^2(x)} \right) \tag{20}$$

by (17). Using (12), we can write

$$\begin{aligned} z_i^2(x;\mu) + f_i^2(x) &= \left(\mu + \sqrt{\mu^2 + f_i^2(x)}\right)^2 + f_i^2(x) \\ &= 2\left(\mu^2 + \mu\sqrt{\mu^2 + f_i^2(x)} + f_i^2(x)\right) \geq 4\mu^2 \end{aligned}$$

for all $1 \leq i \leq m$, which together with (20) proves the lemma. □

Vector $d \in R^n$ obtained by solving (19) is descent for $B(x;\mu)$ if matrix $G(x;\mu)$ is positive definite. Unfortunately, positive definiteness of this matrix is not assured, which causes that standard line-search methods cannot be used. For this reason, trust-region methods were developed. These methods use the direction vector obtained as an approximate minimizer of the quadratic subproblem

$$\text{minimize} \quad Q(d) = \frac{1}{2} d^T \nabla^2 B(x;\mu) d + g^T(x;\mu) d \quad \text{subject to} \quad \|d\| \leq \Delta, \tag{21}$$

where Δ is the trust region radius (more details are given in Section 3). Direction vector d serves for obtaining new point $x^+ \in R^n$. Denoting

$$\rho(d) = \frac{B(x+d;\mu) - B(x;\mu)}{Q(d)}, \tag{22}$$

we set

$$x^+ = x \quad \text{if} \quad \rho(d) \leq 0, \quad \text{or} \quad x^+ = x + d \quad \text{if} \quad \rho(d) > 0. \tag{23}$$

Finally, we update the trust region radius in such a way that

$$\begin{aligned} \Delta^+ = \underline{\beta}\Delta \quad &\text{if} \quad \rho(d) < \underline{\rho}, \\ \Delta^+ = \Delta \quad &\text{if} \quad \underline{\rho} \leq \rho(d) \leq \overline{\rho}, \\ \Delta^+ = \overline{\beta}\Delta \quad &\text{if} \quad \overline{\rho} < \rho(d), \end{aligned} \tag{24}$$

where $0 < \underline{\rho} < \overline{\rho} < 1$ and $0 < \underline{\beta} < 1 < \overline{\beta}$.

Now we are in a position to describe the basic algorithm.

Algorithm 1.

Data: Termination parameter $\underline{\varepsilon} > 0$, minimum value of the barrier parameter $\underline{\mu} > 0$, rate of the barrier parameter decrease $0 < \tau < 1$, trust-region parameters $0 < \underline{\rho} < \overline{\rho} < 1$, trust-region coefficients $0 < \underline{\beta} < 1 < \overline{\beta}$, step bound $\overline{\Delta} > 0$.

Input: Sparsity pattern of matrix A. Initial estimation of vector x.

Step 1: *Initiation.* Choose initial barrier parameter $\mu > 0$ and initial trust-region radius $0 < \Delta \leq \overline{\Delta}$. Determine the sparsity pattern of matrix $\nabla^2 B$ from the sparsity pattern of matrix A. Carry out symbolic decomposition of $\nabla^2 B$. Compute values $f_i(x)$, $1 \leq i \leq m$, and $F(x) = \sum_{1 \leq i \leq m} |f_i(x)|$. Set $k := 0$ (iteration count).

Step 2: *Termination.* Determine vector $z(x;\mu)$ by (12) and vector $u(x;\mu)$ by (10). Compute matrix $A(x)$ and vector $g(x;\mu) = A(x)u(x;\mu)$. If $\mu \le \underline{\mu}$ and $\|g(x;\mu)\| \le \underline{\varepsilon}$, then terminate the computation. Otherwise set $k := k+1$.

Step 3: *Approximation of the Hessian matrix.* Compute approximation of matrix $G(x;\mu)$ by using differences $A(x+\delta v)u(x;\mu) - g(x;\mu)$ for a suitable set of vectors v (see [2]). Determine Hessian matrix $\nabla^2 B(x;\mu)$ by (15).

Step 4: *Direction determination.* Determine vector d as an approximate solution of trust-region subproblem (21).

Step 5: *Step-length selection.* Determine step-length α by (23) and set $x := x + \alpha d$. Compute values $f_i(x)$, $1 \le i \le m$, and $F(x) = \sum_{1 \le i \le m} f_i(x)$.

Step 6: *Trust-region update.* Determine new trust-region radius Δ by (24) and set $\Delta := \min(\Delta, \overline{\Delta})$.

Step 7: *Barrier parameter update.* If $\rho(d) \ge \underline{\rho}$ (where $\rho(d)$ is given by (22)), determine a new value of barrier parameter $\mu \ge \underline{\mu}$ (not greater than the current one) by the procedure described in Section 3. Go to Step 2.

The use of the maximum step-length $\overline{\Delta}$ has no theoretical significance but is very useful for practical computations. First, the problem functions can sometimes be evaluated only in a relatively small region (if they contain exponentials) so that the maximum step-length is necessary. Secondly, the problem can be very ill-conditioned far from the solution point, thus large steps are unsuitable. Finally, if the problem has more local solutions, a suitably chosen maximum step-length can cause a local solution with a lower value of F to be reached. Therefore, maximum step-length $\overline{\Delta}$ is a parameter, which is most frequently tuned.

The important part of Algorithm 1 is the update of barrier parameter μ. There are several influences that should be taken into account, which make the updating procedure rather complicated.

3 Implementation details

In Section 2, we have pointed out that direction vector $d \in R^n$ should be a solution of the quadratic subproblem (21). Usually, an inexact approximate solution suffices. There are several ways for computing a suitable approximate solutions (see, e.g., [19], [4], [22], [23], [18], [21], [13]). We have used two approaches based on direct decompositions of matrix $\nabla^2 B$ (we omit arguments x and μ in the subsequent considerations).

The first strategy, the dog-leg method described in [19], [4], seeks d as a linear combination of the Cauchy step $d_C = -(g^T g / g^T \nabla^2 B g) g$ and the Newton step $d_N = -(\nabla^2 B)^{-1} g$. The Newton step is computed by using either sparse Gill-Murray decomposition [8] or sparse Bunch-Parlett decomposition [5]. The sparse Gill-Murray decomposition has the form $\nabla^2 B + E = LDL^T = R^T R$, where E is a positive semidefinite diagonal matrix (which is equal to zero when $\nabla^2 B$ is positive definite), L is a lower triangular matrix, D is a positive definite diagonal matrix and R is an upper triangular matrix. The sparse Bunch-Parlett decomposition has the form $\nabla^2 B = PLML^T P^T$, where P is a permutation matrix, L is a lower triangular matrix and M is a block-diagonal matrix with 1×1 or 2×2 blocks (which is indefinite when $\nabla^2 B$ is indefinite). The following algorithm is a typical implementation of the dog-leg method.

Algorithm A: Data $\Delta > 0$.

Step 1: If $g^T \nabla^2 B g \le 0$, set $s := -(\Delta/\|g\|)g$ and terminate the computation.

Step 2: Compute the Cauchy step $d_C = -(g^T g / g^T \nabla^2 B g) g$. If $\|d_C\| \ge \Delta$, set $d := (\Delta/\|d_C\|) d_C$ and terminate the computation.

Step 3: Compute the Newton step $d_N = -(\nabla^2 B)^{-1} g$. If $(d_N - d_C)^T d_C \ge 0$ and $\|d_N\| \le \Delta$, set $d := d_N$ and terminate the computation.

Step 4: If $(d_N - d_C)^T d_C \ge 0$ and $\|d_N\| > \Delta$, determine number θ in such a way that $d_C^T d_C / d_C^T d_N \le \theta \le 1$, choose $\alpha > 0$ such that $\|d_C + \alpha(\theta d_N - d_C)\| = \Delta$, set $d := d_C + \alpha(\theta d_N - d_C)$ and terminate the computation.

Step 5: If $(d_N - d_C)^T d_C < 0$, choose $\alpha > 0$ such that $\|d_C + \alpha(d_C - d_N)\| = \Delta$, set $d := d_C + \alpha(d_C - d_N)$ and terminate the computation.

The second strategy, the optimum step method, computes a more accurate solution of (21) by using the Newton method applied to the nonlinear equation

$$\frac{1}{\|d(\lambda)\|} - \frac{1}{\Delta} = 0, \tag{25}$$

where $(\nabla^2 B + \lambda I) d(\lambda) = -g$. This way, described in [18] in more details, follows from the KKT conditions for (21). Since the Newton method applied to (25) can be unstable, safeguards (lower and upper bounds to λ) are usually used. The following algorithm is a typical implementation of the optimum step method.

Algorithm B: Data $0 < \underline{\delta} < 1 < \overline{\delta}$ (usually $\underline{\delta} = 0.9$ and $\overline{\delta} = 1.1$), $\Delta > 0$.

Step 1: Determine $\underline{\nu}$ as the maximum diagonal element of matrix $-\nabla^2 B$. Compute $\overline{\lambda} = \|g\|/\Delta + \|\nabla^2 B\|$, $\underline{\lambda} = \|g\|/\Delta - \|\nabla^2 B\|$ and set $\underline{\lambda} := \max(0, \underline{\nu}, \underline{\lambda})$, $\lambda := \underline{\lambda}$. Set $l = 0$ (inner iteration count).

Step 2: If $l > 0$ and $\lambda \le \underline{\nu}$, set $\lambda := \sqrt{\underline{\lambda}\overline{\lambda}}$.

Step 3: Determine Gill-Murray decomposition $\nabla^2 B + \lambda I + E = R^T R$. If $E = 0$ (i.e. if $\nabla^2 B + \lambda I$ is positive definite), go to Step 4. In the opposite case, determine vector $v \in R^n$ such that $\|v\| = 1$ and $v^T(\nabla^2 B + \lambda I)v \le 0$, set $\underline{\nu} := \lambda - v^T(\nabla^2 B + \lambda I)v$, $\underline{\lambda} := \max(\underline{\nu}, \underline{\lambda})$, $l := l + 1$ and go to Step 2.

Step 4: Determine vector $d \in R^n$ as a solution of equation $R^T R d + g = 0$. If $\|d\| > \overline{\delta}\Delta$, set $\underline{\lambda} := \lambda$ and go to Step 6. If $\underline{\delta}\Delta \le \|d\| \le \overline{\delta}\Delta$, terminate the computation. If $\|d\| < \underline{\delta}\Delta$ and $\lambda = 0$, terminate the computation. If $\|d\| < \underline{\delta}\Delta$ and $\lambda \ne 0$, set $\overline{\lambda} := \lambda$ and go to Step 5.

Step 5: Determine vector $v \in R^n$ as a good approximation of the eigenvector corresponding to the minimum eigenvalue of $\nabla^2 B$ in such a way that $\|v\| = 1$ and $v^T d \ge 0$ hold (this vector can be determined from decomposition $R^T R$ in the way used in subroutines of the LAPACK library). Determine number $\alpha \ge 0$ such that $\|d + \alpha v\| = \Delta$ holds. If $\alpha^2\|Rv\|^2 \le (1 - \underline{\delta}^2)(\|Rd\|^2 + \lambda\Delta^2)$, set $d := d + \alpha v$ and terminate the computation. In the opposite case, set $\underline{\nu} := \lambda - \|Rv\|^2$, $\underline{\lambda} := \max(\underline{\nu}, \underline{\lambda})$ and go to Step 6.

Step 6: Determine vector $v \in R^n$ as a solution of equation $R^T v = d$ and set

$$\lambda := \lambda + \frac{\|d\|^2}{\|v\|^2}\left(\frac{\|d\| - \Delta}{\Delta}\right).$$

If $\lambda < \underline{\lambda}$, set $\lambda := \underline{\lambda}$. If $\lambda > \overline{\lambda}$, set $\lambda := \overline{\lambda}$. Set $l := l + 1$ and go to Step 2.

The above algorithms generate direction vectors such that

$$\begin{aligned} \|d\| &\le \overline{\delta}\Delta, \\ \|d\| &< \underline{\delta}\Delta \;\Rightarrow\; \nabla^2 B d = -g, \\ -Q(d) &\ge \underline{\sigma}\|g\| \min\left(\|d\|, \frac{\|g\|}{\|\nabla^2 B\|}\right), \end{aligned}$$

where $0 < \underline{\sigma} < 1$ is a constant depending on the particular algorithm. These inequalities imply (see [20]), that a constant $0 < \underline{c} < 1$ exists such that

$$\|d\| \ge \underline{c}\gamma/\overline{B}, \tag{26}$$

where γ is the minimum norm of gradients that have been computed and $\overline{B}$ is an upper bound for $\|\nabla^2 B\|$ (assuming $\mu \ge \underline{\mu} > 0$, we can set $\overline{B} = m(\overline{G} + \overline{g}^2/(2\underline{\mu}))$ by Lemma 4). Thus

$$B(x + d; \mu) - B(x; \mu) \le \underline{\rho} Q(d) \le -\underline{\rho}\,\underline{\sigma}\,\underline{c}\frac{\gamma^2}{\overline{B}} \quad \text{if} \quad \rho \ge \underline{\rho} \tag{27}$$

by (23) and (26).

A very important part of Algorithm 1 is the update of the barrier parameter μ. There are two requirements, which play opposite roles. First $\mu \to 0$ should hold, since this is the main property of every interior point method. On the other hand, the convergence theory requires (27) to hold. Thus a lower bound $\underline{\mu}$ for the barrier parameter has to be used (we recommend value $\underline{\mu} = 10^{-6}$ in double precision arithmetic).

Algorithm 1 is also sensitive on the way in which the barrier parameter decreases. We have tested various possibilities for the barrier parameter update including simple geometric sequences, which were proved to be unsuitable. Better results were obtained by setting

$$\mu_{k+1} = \mu_k \quad \text{if} \quad \|g_k\|^2 > \tau\mu_k \quad \text{or} \quad \mu_{k+1} = \max(\underline{\mu}, \|g_k\|^2) \quad \text{if} \quad \|g_k\|^2 \le \tau\mu_k, \tag{28}$$

where $0 < \tau < 1$.

4 Global convergence

In the subsequent considerations, we will assume that $\underline{\varepsilon} = \underline{\mu} = 0$ and all computations are exact. We will investigate infinite sequence $\{x_k\}_1^\infty$ generated by Algorithm 1.

Lemma 5. *Let Assumption 3 be satisfied. Then values $\{\mu_k\}_1^\infty$, generated by Algorithm 1, form a non-increasing sequence such that $\mu_k \to 0$.*

Proof. (a) First we prove that $B(x;\mu)$ is bounded from below if μ is fixed. Since $z_i(x;\mu) \ge 0$ and

$$\begin{aligned} z_i^2(x;\mu) - f_i^2(x) &= 2\mu z_i(x;\mu) = 2\mu\left(\mu + \sqrt{\mu^2 + f_i^2(x)}\right) \\ &\le 2\mu(2\mu + |f_i(x)|) \le 2\mu(2\mu + \overline{F}) \end{aligned} \tag{29}$$

for all $1 \le i \le m$ by (11) and (12), we can write

$$B(x;\mu) = \sum_{i=1}^m z_i(x;\mu) - \mu \sum_{i=1}^m \log(z_i^2(x;\mu) - f_i^2(x)) \ge -\mu m \log\left(2\mu(2\mu + \overline{F})\right). \tag{30}$$

(b) Now we prove that the sequence of points in which μ_k is updated is infinite. If it was finite, an index $l \in N$ would exist such that $\mu_{k+1} = \mu_k = \mu_l \; \forall k \ge l$. Since function $B(x;\mu_l)$ is continuous, bounded from below by (a) and since (27) (with $\mu_k = \mu_l$) holds $\forall k \ge l$, it can be proved (see [20]) that $\liminf_{k\to\infty} \|g(x_k;\mu_l)\| = 0$. Thus an index $k \ge l$ exists such that $\|g(x_k;\mu_l)\|^2 \le \tau\mu_l$ and, therefore, $\mu_{k+1} = \|g(x_k;\mu_l)\|^2 \le \tau\mu_l < \mu_l$ by (28), which is a contradiction. Since the sequence of points where $\mu_{k+1} \le \tau\mu_k$ is infinite, we can conclude that $\mu_k \to 0$. $\square$

Now we will prove that

$$B(x_{k+1};\mu_{k+1}) \le B(x_{k+1};\mu_k) - L(\mu_{k+1} - \mu_k) \tag{31}$$

for some $L \in R$. For this purpose, we consider that $z(x;\mu)$ and $B(x;\mu)$ are functions of μ and we write $z(x,\mu) = z(x;\mu)$ and $B(x,\mu) = B(x;\mu)$.

Lemma 6. *Let $z_i(x,\mu)$, $1 \le i \le m$, be values given by Lemma 1 (for fixed x and variable μ). Then*

$$\frac{\partial z_i(x,\mu)}{\partial \mu} > 1, \quad \forall 1 \le i \le m,$$

and

$$\frac{\partial B(x,\mu)}{\partial \mu} = -\sum_{i=1}^{m} \log(z_i^2(x,\mu) - f_i^2(x)).$$

Proof. Differentiating expressions $z_i(x,\mu) = \mu + \sqrt{\mu^2 + f_i^2(x)}$, $1 \le i \le m$, following from Lemma 1, we obtain

$$\frac{\partial z_i(x,\mu)}{\partial \mu} = 1 + \frac{\mu}{\sqrt{\mu^2 + f_i^2(x)}} > 1, \quad 1 \le i \le m.$$

Differentiating function

$$B(x,\mu) = \sum_{i=1}^{m} z_i(x,\mu) - \mu \sum_{i=1}^{m} \log(z_i^2(x,\mu) - f_i^2(x)),$$

one has

$$\begin{aligned}
\frac{\partial B(x,\mu)}{\partial \mu} &= \sum_{i=1}^{m} \frac{\partial z_i(x,\mu)}{\partial \mu} - \sum_{i=1}^{m} \log(z_i^2(x,\mu) - f_i^2(x)) - \sum_{i=1}^{m} \frac{2\mu z_i(x,\mu)}{z_i^2(x,\mu) - f_i^2(x)} \frac{\partial z_i(x,\mu)}{\partial \mu} \\
&= \sum_{i=1}^{m} \frac{\partial z_i(x,\mu)}{\partial \mu} \left(1 - \frac{2\mu z_i(x,\mu)}{z_i^2(x,\mu) - f_i^2(x)}\right) - \sum_{i=1}^{m} \log(z_i^2(x,\mu) - f_i^2(x)) \\
&= -\sum_{i=1}^{m} \log(z_i^2(x,\mu) - f_i^2(x))
\end{aligned}$$

by (11). □

Lemma 7. *Let Assumption 3 be satisfied. Then (31) holds with some $L \in R$.*

Proof. Using Lemma 6, the mean value theorem and (29), we can write

$$\begin{aligned}
B(x_{k+1},\mu_{k+1}) - B(x_{k+1},\mu_k) &= -\sum_{i=1}^{m} \log(z^2(x_{k+1},\tilde{\mu}_k) - f_i^2(x_{k+1}))(\mu_{k+1} - \mu_k) \\
&\le -\sum_{i=1}^{m} \log\left(2\tilde{\mu}_k(2\tilde{\mu}_k + \overline{F})\right)(\mu_{k+1} - \mu_k) \\
&\le -\sum_{i=1}^{m} \log\left(2\mu_1(2\mu_1 + \overline{F})\right)(\mu_{k+1} - \mu_k) \\
&\triangleq -L(\mu_{k+1} - \mu_k),
\end{aligned}$$

where $0 < \mu_{k+1} \le \tilde{\mu}_k \le \mu_k \le \mu_1$. □

Theorem 1. *Let Assumption 3 be satisfied. Consider sequence* $\{x_k\}_1^\infty$*, generated by Algorithm 1. Then*

$$\liminf_{k\to\infty} \sum_{i=1}^m u_i(x_k;\mu_k) g_i(x_k) = 0$$

and

$$u_i(x_k;\mu_k) = \frac{f_i(x_k)}{z_i(x_k;\mu_k)}, \quad \lim_{k\to\infty} (z_i^2(x_k;\mu_k) - f_i^2(x_k)) = 0$$

for $1 \le i \le m$.

Proof. Equalities $u_i(x_k;\mu_k) = f_i(x_k)/z_i(x_k;\mu_k)$, $1 \le i \le m$, follow from (8).

(a) Since (31) holds, we can write

$$\begin{aligned} B(x_{k+1};\mu_{k+1}) - B(x_k;\mu_k) &= (B(x_{k+1};\mu_{k+1}) - B(x_{k+1};\mu_k)) + (B(x_{k+1};\mu_k) - B(x_k;\mu_k)) \\ &\le -L(\mu_{k+1}-\mu_k) + (B(x_{k+1};\mu_k) - B(x_k;\mu_k)), \end{aligned}$$

which together with (31), (27) and Lemma 5 implies

$$\begin{aligned} \underline{B} &\le \lim_{k\to\infty} B(x_k;\mu_k) \le B(x_1;\mu_1) - L\sum_{k=1}^\infty(\mu_{k+1}-\mu_k) + \sum_{k=1}^\infty (B(x_{k+1};\mu_k) - B(x_k;\mu_k)) \\ &\le B(x_1;\mu_1) + L\mu_1 + \sum_{\rho(d_k)\ge\underline{\rho}} (B(x_{k+1};\mu_k) - B(x_k;\mu_k)) \\ &\le B(x_1;\mu_1) + L\mu_1 - \frac{\underline{\rho}\,\underline{\sigma}\,\underline{c}}{\overline{B}} \sum_{\rho(d_k)\ge\underline{\rho}} \|\gamma_k\|^2, \end{aligned}$$

where

$$\underline{B} = \min_{\underline{\mu}\le\mu\le\mu_1} \left(-\mu m \log(2\mu(2\mu + \overline{F}))\right) \ge \min\left(0, -\mu_1 m \log(2\mu_1(2\mu_1+\overline{F}))\right)$$

(see (29)) and $\gamma_k = \min_{1\le j\le k} \|g(x_j;\mu_j)\|$. If $\liminf_{k\to\infty} \|g(x_k;\mu_k)\| = 0$ was not satisfied, a number $\varepsilon > 0$ would exist such that

$$\|g(x_k;\mu_k)\| \ge \varepsilon \quad \forall k \in N. \tag{32}$$

Then, using the previous inequality, we would obtain

$$\underline{B} - B(x_1;\mu_1) - L\mu_1 \le -\frac{\underline{\rho}\,\underline{\sigma}\,\underline{c}}{\overline{B}} \sum_{\rho(d_k)\ge\underline{\rho}} \varepsilon^2.$$

It remains to prove that the sum on the right hand side is infinite, which gives a contradiction. If this sum was finite, an index $l \in N$ would exist such that $\rho_k(d_k) < \underline{\rho}$ $\forall k \ge l$. Thus $\Delta_k \to 0$ by (24), which with inequality $\Delta_k \ge \|d_k\| \ge \underline{c}\gamma_k/\overline{B}$ (see (26)) gives $\gamma_k \to 0$. But this is in contradiction with (32).

(b) Using (29), one has $z_i^2(x_k;\mu_k) - f_i^2(x_k) \le 2\mu_k(2\mu_k + \overline{F})$. Thus $z_i^2(x_k;\mu_k) - f_i^2(x_k) \to 0$ as $\mu_k \to 0$. □

Remark 1. If we replace (23) by

$$x^+ = x \quad \text{if} \quad \rho(d) < \underline{\rho}, \quad \text{or} \quad x^+ = x + d \quad \text{if} \quad \rho(d) \ge \underline{\rho} \tag{33}$$

in Algorithm 1, then $\lim_{k\to\infty} \|g(x_k;\mu_k)\| = 0$. A proof of this assertion can be found, e.g., in [3].

Corollary 1. *Let assumptions of Theorem 1 and (33) hold. Then every cluster point* $x \in R^n$ *of sequence* $\{x_k\}_1^\infty$ *satisfies KKT conditions (6), where* $u \in R^m$ *is a cluster point of sequence* $\{u(x_k;\mu_k)\}_1^\infty$.

5 Computational experiments

The primal interior-point method was tested by using the collection of relatively difficult problems with optional dimension chosen from [15], which can be downloaded (together with the above report) from `www.cs.cas.cz/~luksan/test.html` as Test 14 and Test 15. Functions $f_i(x)$, $1 \le i \le m$, given in [15] serve for defining objective function

$$F(x) = \sum_{1\le i\le m} |f_i(x)|. \tag{34}$$

We have used parameters $\underline{\varepsilon} = 10^{-6}$, $\underline{\mu} = 10^{-6}$, $\underline{\delta} = 0.9$, $\overline{\delta} = 1.1$, $\overline{\Delta} = 1000$, $\underline{\rho} = 0.1$, $\overline{\rho} = 0.9$, $\underline{\beta} = 0.5$, $\overline{\beta} = 2.0$, $\tau = 0.01$ in Algorithm 1 as defaults (step bound $\overline{\Delta}$ was sometimes tuned).

The first set of the tests concerns comparison of interior-point methods with various trust-region and line-search strategies and the bundle variable metric method proposed in [17]. Medium-size test problems with 200 variables were used. The results of computational experiments are reported in two tables, where only summary results (over all 22 test problems) are given. Here M is the method used: T1 – Algorithm A with the Bunch-Parlett decomposition, T2 – Algorithm A with the Gill-Murray decomposition, T3 – Algorithm B with the Gill-Murray decomposition, L – line-search with restarts described in [14], B – bundle variable metric method described in [17]; NIT is the total number of iterations, NFV is the total number of function evaluations, NFG is the total number of gradient evaluations, NR is the total number of restarts, NF is the number of problems, for which the global minimizer was not found (either a worse local minimum was obtained or the method failed even if parameter $\overline{\Delta}$ was tuned), NT is the number of problems for which parameter $\overline{\Delta}$ was tuned and TIME is the total computational time in seconds.

M	NIT	NFV	NFG	NR	NF	NT	TIME
T1	2263	2520	21133	-	1	10	4.14
T2	2411	2768	18417	-	-	8	2.67
T3	2441	2784	20143	2	-	6	4.02
L	11955	23625	62658	-	1	8	11.30
B	32608	32634	32634	22	2	11	24.72

Table 1: Test 14: Function (34) with 200 variables

M	NIT	NFV	NFG	NR	NF	NT	TIME
T1	3440	4050	19623	3	1	13	3.69
T2	2541	2986	13745	5	-	11	2.63
T3	4655	5477	27090	2	-	12	5.11
L	6886	17162	43339	28	1	16	12.17
B	34608	34854	34854	22	1	20	13.61

Table 2: Test 15: Function (34) with 200 variables

The results introduced in these tables indicate that trust-region strategies are more efficient than restarted line-search strategies in connection with the interior-point method for l_1 optimization. These observations differs from conclusions concerning the interior-point method for minimax optimization proposed in [14], where matrix $\nabla^2 B$ has a different structure. The trust-region interior-point method is less sensitive to the choice of parameters and requires a lower number of iterations and shorter computational time in comparison with the bundle variable metric method proposed in [17]. This method also finds the global minimum (if the l_1 problems has several local solutions) more frequently (see column NF in the above tables).

The second set of tests concerns a comparison of the interior-point method, realized as a dog-leg method with the Gill-Murray decomposition, with the bundle variable metric method described in [17]. Large-scale test problems with 1000 variables are used. The results of computational experiments are given in two tables, where P is the problem number, NIT is the number of iterations, NFV is the number of function evaluations, NFG is the number of gradient evaluations and F is the function value reached. The last row of every table contains summary results including the total computational time. The bundle variable metric method was chosen for comparison, since it is based on a quite different principle and can also be used for large sparse l_1 optimization.

	Trust-region interior-point method				Bundle variable metric method			
P	NIT	NFV	NFG	F	NIT	NFV	NFG	F
1	1591	1595	6368	0.4459497398E-08	7819	7842	7842	0.1740238903E-20
2	411	503	2884	0.1735612989E-07	127	130	130	0.7355230443E-17
3	29	30	210	0.9017167132E-06	89	89	89	0.3593641257E-14
4	26	27	189	269.4995435	81	81	81	269.4995435
5	26	27	162	0.5541094393E-05	39	39	39	0.1224564314E-14
6	27	28	392	0.5068252242E-05	100	100	100	0.1103581175E-12
7	18	20	171	336.9371813	211	211	211	336.9371813
8	18	19	342	761774.9537	36	39	39	761774.9537
9	120	135	2178	327.6803233	5725	5725	5725	327.6832858
10	855	1021	15408	0.4793646012E-01	14463	14463	14463	0.8311440390E-01
11	33	50	204	86.86730605	319	319	319	10.77658789
12	131	132	528	982.2736173	115	117	117	982.2736173
13	24	25	100	0.4834264843E-05	16	17	17	0.1391782738E-18
14	1	12	6	0.1293828504E-08	3	3	3	0.1293829244E-08
15	138	172	556	1.961691268	3970	3973	3973	1.971202794
16	163	200	984	0.2339556263E-12	4345	4382	4382	0.4752696299E-03
17	174	200	1050	0.1067680449E-07	456	458	458	0.8530087038E-06
18	2043	2466	12264	0.1973176916E-04	1206	1216	1216	0.1296949257E-03
19	16	17	102	59.59862420	182	182	182	59.59862413
20	1441	1810	8652	0.3369077042E-11	6197	6197	6197	0.7724217956E-06
21	20	21	126	2.138663960	29	30	30	2.138663772
22	1531	1908	9192	1.000000000	337	341	341	1.000000000
Σ	8836	10418	62068	TIME=42.24	45865	45954	45954	TIME=132.83

Table 3: Test 14: Function (34) with 1000 variables

	Trust-region interior-point method				Bundle variable metric method			
P	NIT	NFV	NFG	F	NIT	NFV	NFG	F
1	1464	1477	5860	0.1233457780E-12	359	540	540	0.8157570619E-08
2	83	108	420	4.000005849	453	473	473	0.1533432491E-07
3	26	30	162	0.7993891363E-08	114	114	114	0.3749132061E-08
4	66	77	268	648.2320709	53	54	54	648.2320706
5	6	7	42	0.6605826997E-14	285	285	285	0.4227249141E-05
6	8	9	126	0.7194939089E-13	560	560	560	0.6495303739E-08
7	86	119	348	12029.94352	542	650	650	12029.94285
8	83	102	252	0.8235062515E-05	1029	1032	1032	0.6800618388E-04
9	280	336	1967	2777.654183	4428	4429	4429	2780.112235
10	67	85	408	658.0491855	854	854	854	658.0486548
11	3435	3644	13744	0.1110223025E-15	411	454	454	0.8383737038E-09
12	463	553	2784	3125.762684	1879	1882	1882	3125.852391
13	85	112	516	14808.85074	727	728	728	14808.85239
14	103	123	728	566.1130406	740	740	740	566.1127477
15	59	83	420	181.9262645	647	647	647	181.9261639
16	96	127	679	66.53358282	984	984	984	66.53333334
17	52	64	477	0.1775328747E-11	9092	9092	9092	0.3379782731E-08
18	1109	1126	6660	0.1021405183E-13	3160	3160	3160	0.7549008346
19	29	31	330	0.3510525204E-12	15933	15944	15944	0.2392440468E-08
20	59	73	360	0.1522693083E-10	1509	1699	1699	0.7569758154E-08
21	94	125	570	1326.921870	425	426	426	1327.950158
22	133	167	804	2993.367736	9875	9875	9875	2993.375706
Σ	7886	8578	37925	TIME=29.30	54059	54622	54622	TIME=135.00

Table 4: Test 15: Function (34) with 1000 variables

The results introduced in these tables confirm conclusions following from the previous tables. The trust-region interior-point method seems to be more efficient than the bundle variable metric method in all indicators. Especially, the computational time is much shorter and also the number of global minima attained is greater in the case of the trust-region interior-point method. We believe that the efficiency of the interior-point method could be improved by using a better procedure for the barrier parameter update.

References

[1] R.H.Byrd, J.Nocedal, R.A.Waltz: Feasible interior methods using slacks for nonlinear optimization. Report No. OTC 2000/11, Optimization Technology Center, December 2000.

[2] T.F.Coleman, J.J.Moré: Estimation of sparse Hessian matrices and graph coloring problems. Mathematical Programming 28 (1984) 243-270.

[3] A.R.Conn, N.I.M.Gould, P.L.Toint: Trust-region methods. SIAM 2000.

[4] J.E.Dennis, H.H.W.Mei: An unconstrained optimization algorithm which uses function and gradient values. Report No. TR 75-246, 1975.

[5] I.S.Duff, M.Munksgaard, H.B.Nielsen, J.K.Reid: Direct solution of sets of linear equations whose matrix is sparse and indefinite. J. Inst. Maths. Applics. 23 (1979) 235-250.

[6] R.Fletcher: Practical Methods of Optimization (second edition). Wiley, New York, 1987.

[7] R.Fletcher, E.Sainz de la Maza: Nonlinear programming and nonsmooth optimization by successive linear programming. Mathematical Programming (43) (1989) 235-256.

[8] Gill, P.E., Murray, W: Newton type methods for unconstrained and linearly constrained optimization. Mathematical Programming, 7, 311-350 (1974).

[9] N.I.M.Gould, S.Lucidi, M.Roma, P.L.Toint: Solving the trust-region subproblem using the Lanczos method. Report No. RAL-TR-97-028, 1997.

[10] P.E.Gill, W.Murray, M.A.Saunders: SNOPT: An SQP algorithm for large-scale constrained optimization. SIAM Review 47 (2005) 99-131.

[11] L.Lukšan, C.Matonoha, J.Vlček: Interior point method for nonlinear nonconvex optimization. Numerical Linear Algebra with Applications 11 (2004) 431-453.

[12] L.Lukšan, C.Matonoha, J.Vlček: Nonsmooth equation method for nonlinear nonconvex optimization. In: Conjugate Gradient Algorithms and Finite Element Methods (M.Křížek, P.Neittaanmäki, R.Glovinski, S.Korotov eds.). Springer Verlag, Berlin 2004.

[13] L.Lukšan, C.Matonoha, J.Vlček: A shifted Steihaug-toint method for computing a trust-region step. Technical Report V-914. Prague, ICS AS CR, 2004. Submitted to BIT Numerical Mathematics.

[14] L.Lukšan, C.Matonoha, J.Vlček: Primal interior-point method for large sparse minimax optimization. Technical Report V-941. Prague, ICS AS CR, 2005.

[15] L.Lukšan, J.Vlček: Sparse and partially separable test problems for unconstrained and equality constrained optimization, Report V-767, Prague, ICS AS CR, 1998.

[16] L.Lukšan, J.Vlček: Indefinitely Preconditioned Inexact Newton Method for Large Sparse Equality Constrained Nonlinear Programming Problems. Numerical Linear Algebra with Applications 5 (1998) 219-247.

[17] Lukšan L., Vlček J.: Variable metric method for minimization of partially separable nonsmooth functions. To appear in Pacific Journal on Optimization.

[18] J.J.Moré, D.C.Sorensen: Computing a trust region step. SIAM Journal on Scientific and Statistical Computations 4 (1983) 553-572.

[19] M.J.D.Powell: A new algorithm for unconstrained optimization. In: "Nonlinear Programming" (J.B.Rosen O.L.Mangasarian, K.Ritter, eds.) Academic Press, London 1970.

[20] M.J.D.Powell: On the global convergence of trust region algorithms for unconstrained optimization. Mathematical Programming 29 (1984) 297-303.

[21] M.Rojas, S.A.Santos, D.C.Sorensen: A new matrix-free algorithm for the large-scale trust-region subproblems. SIAM J. Optimization 11 (2000) 611-646.

[22] T.Steihaug: The conjugate gradient method and trust regions in large-scale optimization. SIAM Journal on Numerical Analysis 20 (1983) 626-637.

[23] P.L.Toint: Towards an efficient sparsity exploiting Newton method for minimization. In: Sparse Matrices and Their Uses (I.S.Duff, ed.), Academic Press, London 1981, 57-88.

[24] J.Vanderbei, D.F.Shanno: An interior point algorithm for nonconvex nonlinear programming. Computational Optimization and Applications, 13, 231-252, 1999.