Ladislav Lukšan

Quasi-Newton methods without projections for linearly constrained minimization

Kybernetika, Vol. 18 (1982), No. 4, 307--319

Persistent URL: http://dml.cz/dmlcz/125371

KYBERNETIKA — VOLUME 18 (1982), NUMBER 4

QUASI-NEWTON METHODS WITHOUT PROJECTIONS FOR LINEARLY CONSTRAINED MINIMIZATION

LADISLAV LUKŠAN

This contribution contains a description of a class of quasi-Newton methods without projections for linearly constrained minimization. These methods are generalizations of quasi-Newton methods without projections proposed by the author and use the active set strategy with reduced gradients in the form which is a combination of the results given previously in [4] and [7]. Moreover an algorithm is presented which is an implementation of a class of quasi-Newton methods without projections for linearly constrained minimization and the efficiency of this algorithm is demonstrated by means of test functions.

1. INTRODUCTION

In [6] the present author has proposed several forms of quasi-Newton methods without projections for unconstrained minimization which are modification of quasi-Newton methods described in [2]. In this paper we shall use a product form of quasi-Newton methods without projections to derive a class of reduced gradient methods for linearly constrained minimization.

At the beginning of this section we summarize briefly the results about the product form of quasi-Newton methods without projections given in Section 4 of [6]. An iteration of the product form of quasi-Newton methods without projections has the form

$$\left\{\begin{aligned} x^+ &= x - \varrho S\tilde{g} \\ \tilde{a} &= \tilde{v} + \sqrt{(\varphi)}\,(\beta\tilde{u} - \alpha\tilde{v}) \\ \tilde{b} &= \sqrt{(q)}\,\tilde{v} - \sqrt{(\varphi)}\,(\delta\tilde{u} - \gamma\tilde{v}) \\ \tilde{u}^+ &= \frac{1}{q}\left(I + \frac{1}{\lambda}\,\tilde{b}\tilde{a}^{\mathrm{T}}\right)(\delta\tilde{u} - \gamma\tilde{v}) \\ S^+ &= S\left(I + \frac{1}{\lambda}\,\tilde{a}\tilde{b}^{\mathrm{T}}\right) \\ &\text{where} \\ \lambda &= \delta + \beta\sqrt{(q)} + (\beta\sigma - \alpha\tau)\sqrt{\varphi} \\ &\text{and} \\ q &= \frac{\delta}{\beta} - \frac{\varphi}{\beta\tau}(\mathrm{B} + \mathrm{D}) \end{aligned}\right. \tag{1.1}$$

(we use the notation $x^+ = x - \varrho S\tilde{g}$ instead of the standard notation $x_{k+1} = x_k - \varrho_k S_k \tilde{g}_k$, $k = 1, 2, \ldots$). At the same time $\tilde{g} = S^T g$, $\tilde{d} = -\varrho\tilde{g}$, $\tilde{y} = S^T(g^+ - g)$, $\tilde{v} = \tilde{d} - \tilde{y}$ and $\alpha = \tilde{u}^T\tilde{y}$, $\beta = \tilde{v}^T\tilde{y}$, $\gamma = \tilde{u}^T\tilde{d}$, $\delta = \tilde{v}^T\tilde{d}$, $\varepsilon = \tilde{u}^T\tilde{u}$, $\sigma = \tilde{u}^T\tilde{v}$, $\tau = \tilde{v}^T\tilde{v}$ and

$$\begin{aligned}
\mathrm{A} &= \beta^2(\varepsilon\tau - \sigma^2)\\
\mathrm{B} &= \beta\delta(\varepsilon\tau - \sigma^2)\\
\mathrm{C} &= \delta^2(\varepsilon\tau - \sigma^2)\\
\mathrm{D} &= (\beta\sigma - \alpha\tau)^2 = (\delta\sigma - \gamma\tau)^2
\end{aligned}$$

and ϱ is a steplength which is taken so that $F(x^+) < F(x)$ may hold. The choice of the free parameter φ was investigated in Section 5 of [6]. S is a matrix with linearly independent columns and $\tilde{u} = \tilde{g}$ in the first iteration.

The product form of quasi-Newton methods without projections can be generalized for problems with linear constraints by suitable choice of the matrix S. This matrix must be chosen in such a manner that its columns may define a basis in the orthogonal complement of the subspace spanned by normals of active constraints. We shall show, in the following sections, the rules for changing the matrix S whenever the set of active constraints is changed. The second order information must be kept in the matrix S over these changes.

At the end of this paper we shall describe an algorithm which implements a class of quasi-Newton methods without projections and we shall show its efficiency by means of several testing functions.

2. MINIMIZATION WITH LINEAR CONSTRAINTS

Consider the problem (P) of minimizing an objective function $F(x)$ in a convex polytope

$$C = \{x \in R_n : a_i^T x \geqq b_i,\ 1 \leqq i \leqq m\}$$

A point $x \in C$ is called feasible. The set of indices

$$I(x) = \{i : a_i^T x = b_i,\ 1 \leqq i \leqq m\}$$

can be defined for each feasible point $x \in C$. The constraints are called active if their indices belong to $I(x)$ and we suppose them to be linearly independent. Let g be the gradient of the objective function $F(x)$ at the feasible point $x \in C$.

Definition 2.1. A direction s is called descent at the feasible point $x \in C$ if $g^T s < 0$. A direction s is called feasible at the feasible point $x \in C$ if $a_i^T s \geqq 0$ for all $i \in I(x)$.

The method studied in this paper uses only feasible directions. Its iteration begins at a feasible point $x \in C$ and use a descent feasible direction s so that the point

$$x^+ = x + \varrho s$$

may be feasible and $F(x^+) < F(x)$ for some steplength ϱ, $0 < \varrho \leqq \tilde{\varrho}$. The maximum steplength $\tilde{\varrho}$ is defined as

$$(2.1)\qquad \begin{cases} \tilde{\varrho} = \min\limits_{i \in \tilde{I}(x)} \left(\dfrac{b_i - a_i^T x}{a_i^T s} \right) \\ \text{where} \\ \tilde{I}(x) = \{ i \notin I(x) : a_i^T s < 0 \} \end{cases}$$

Now suppose the normals a_i, $i \in I(x)$ to be linearly independent at each feasible point $x \in C$.

Definition 2.2. Let A be a matrix whose columns are the normals a_i, $i \in I(x)$ and let S be a matrix such that $[A, S]$ is a nonsingular square matrix of order n and $A^T S = 0$. Then we say that A, S is an orthogonal pair of matrices generated by the set $I(x)$.

Lemma 2.1. Let A, S be an orthogonal pair of matrices generated by the set $I(x)$. Let the gradient g be not a linear combination of the normals a_i, $i \in I(x)$. Then

$$s = -SS^T g$$

is a descent feasible direction.

Proof. The gradient g is not a linear combination of the normals a_i, $i \in I(x)$, hence $S^T g \neq 0$ by Definition 2.2 and therefore $g^T s = -g^T S S^T g < 0$. Moreover $a_i^T s = 0$, $i \in I(x)$ holds by Definition 2.2. □

If $g = Au$, then the direction $s = -SS^T g$ is zero since the minimum of the objective function has been found on the linear manifold defined by active constraints. If in addition, $u \geqq 0$, then Kuhn-Tucker conditions are satisfied and the problem (P) can be as usual solved. If $g = Au$ and $u_j < 0$ for some $j \in I(x)$, the j-th active constraint can be deleted from the basis of active constraints. Then matrices A^-, S^- can be found so as to form an orthogonal pair of matrices generated by the set $I^-(x) = I(x) \setminus \{j\}$.

Theorem 2.1. Let A, S be an orthogonal pair of matrices generated by the set $I(x)$. Let $u_j < 0$ for some $j \in I(x)$, where

$$u = (A^T A)^{-1} A^T g$$

and where g is the gradient of the objective function $F(x)$ at a feasible point $x \in C$. Let A^-, S^- be an orthogonal pair of matrices generated by the set $I^-(x) = I(x) \setminus \{j\}$ such that $S^- = [S, s_0]$ and $S^T s_0 = 0$. Then

$$s^- = -S^-(S^-)^T g$$

is a descent feasible direction. Moreover $a_j^T s^- > 0$.

Proof. Since $[A, S]$ is a nonsingular square matrix of order n, the gradient g can be uniquely expressed in the form $g = Au + Sv$ so that $A^{\mathrm{T}}g = A^{\mathrm{T}}Au + A^{\mathrm{T}}Sv = A^{\mathrm{T}}Au$. The matrix $A^{\mathrm{T}}A$ is nonsingular so that $u = (A^{\mathrm{T}}A)^{-1} A^{\mathrm{T}}g$. Now $s_0 \neq 0$, $(A^-)^{\mathrm{T}}s_0 = 0$ and $S^{\mathrm{T}}s_0 = 0$ by assumption, which implies $s_0^{\mathrm{T}}a_j \neq 0$ and therefore $(S^-)^{\mathrm{T}} a_j \neq 0$. Furthermore we have

$$S^{\mathrm{T}}S^-(S^-)^{\mathrm{T}} a_j = S^{\mathrm{T}}SS^{\mathrm{T}}a_j + S^{\mathrm{T}}s_0s_0^{\mathrm{T}}a_j = 0$$

since $S^{\mathrm{T}}A = 0$ and therefore $S^{\mathrm{T}}a_j = 0$. Now let us set $s^- = -S^-(S^-)^{\mathrm{T}} g$. The matrix A^- results from the matrix A aftet deleting the column a_j so that

$$a_j^{\mathrm{T}}s^- = -a_j^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} g = -a_j^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} (Au + Sv) =$$
$$= -a_j^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} a_ju_j - a_j^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} Sv = -a_j^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} a_ju_j$$

since we have proved $S^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} a_j = 0$ and since $(S^-)^{\mathrm{T}} A^- = 0$ implies $(S^-)^{\mathrm{T}} Au = (S^-)^{\mathrm{T}} a_ju_j$. Now $u_j < 0$ by assumption and $(S^-)^{\mathrm{T}} a_j \neq 0$ as we have proved above so that $a_j^{\mathrm{T}}s^- > 0$. Moreover $(A^-)^{\mathrm{T}} s^- = -(A^-)^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} g = 0$ so that s^- is a feasible direction. But s^- is even a descent feasible direction since $g^{\mathrm{T}}s^- = -g^{\mathrm{T}} S^-(S^-)^{\mathrm{T}} g < 0$. □

Theorem 2.1 shows that the constraint with the normal a_j can be deleted from the basis whenever $u_j < 0$, even if g is not a linear combination of the normals a_i, $i \in I(x)$.

If $\varrho = \tilde{\varrho}$ holds in the iteration (1.1) where $\tilde{\varrho}$ is defined by (2.1) then a new constraint must be added to the basis. It is just the constraint with the normal a_j, say, which has limited the steplength ϱ. Then matrices A^+, S^+ can be found so as to form an orthogonal pair of matrices generated by the set $I(x^+) = I(x) \cup \{j\}$.

We have proved that either the Kuhn-Tucker conditions are satisfied or a descent feasible direction can be found at each feasible point $x \in C$. In the first case the problem (P) is usually solved (necessary conditions are satisfied), otherwise the product form iteration (1.1) can be applied.

The new class of quasi-Newton methods without projections uses three matrices A, S and R which represent a current basis of active constraints. Here A, S is an orthogonal pair of matrices generated by the set $I(x)$ and R is an upper triangular matrix of order n such that $R^{\mathrm{T}}R = A^{\mathrm{T}}A$. The matrix R serves for computation of the Lagrange multipliers from the equation

$$R^{\mathrm{T}}Ru = A^{\mathrm{T}}g \,. \tag{2.2}$$

The matrices A, S and R must be updated after each change of the current basis of active constraints.

3. ADDING A CONSTRAINT TO THE BASIS

We shall describe a construction of the matrices A^+, S^+ and R^+ for the case when the constraint with the normal a_j is added to the basis.

Lemma 3.1. Let A, S be an orthogonal pair of matrices generated by the set $I(x)$. Then

$$SS^T = H - H\,A(A^THA)^{-1}\,A^TH \tag{3.1}$$

where H is some symmetric positive definite matrix of order n.

Proof. Let $H = SS^T + AA^T$. Then H is symmetric and positive definite since A, S is an orthogonal pair of matrices generated by the set $I(x)$. Moreover (3.1) holds since $A^TS = 0$. □

The matrix H contains a second order information obtained during preceding iterations (it is an approximation of Hessian matrix of the Lagrange function). Therefore we require the matrix S^+ to satisfy the condition

$$S^+(S^+)^T = H - HA^+((A^+)^T HA^+)^{-1}(A^+)^T H \tag{3.2}$$

Lemma 3.2. Let A, S be an orthogonal pair of matrices generated by the set $I(x)$ and let (3.1) holds. Let A^+, S^+ be an orthogonal pair of matrices generated by the set $I(x^+) = I(x) \cup \{j\}$ where $j \notin I(x)$. Then condition (3.2) is satisfied if and only if

$$S^+(S^+)^T = SS^T - \frac{SS^Ta_ja_j^TSS^T}{a_j^TSS^Ta_j} \tag{3.3}$$

Proof. See [5] for example. □

Theorem 3.1. Let A, S be an orthogonal pair of matrices generated by the set $I(x)$ and let $A^+ = [A, a_j]$ where the normal a_j is not a linear combination of the normals a_i, $i \in I(x)$. Let

$$S^+ = \tilde{S} - \left(\frac{1 - ts_k^Ta_j}{s^Ta_j}\,s + ts_k\right) a_j^T\tilde{S} \tag{3.4}$$

where $\tilde{S}$ is a matrix resulting from the matrix S after deleting an arbitrary column s_k and where $s = SS^Ta_j$. If t is a root of the quadratic equation

$$\omega^2t^2 + 2ts_k^Ta_j - 1 = 0$$

where $\omega = a_j^T\tilde{S}\tilde{S}^Ta_j = s^Ta_j - (s_k^Ta_j)^2$, then (3.3) holds. Moreover A^+, S^+ is an orthogonal pair of matrices generated by the set $I(x^+) = I(x) \cup \{j\}$.

Proof. The normal a_j is not a linear combination of the normals a_i, $i \in I(x)$ so that $S^T a_j \neq 0$. Moreover $s = SS^T a_j \neq 0$ since the matrix S has linearly independent columns. Let

$$S^+ = \tilde{S} - (\lambda s + t s_k)\, a_j^T \tilde{S}$$

where

$$\lambda = \frac{1 - t s_k^T a_j}{s^T a_j}$$

Then we obtain

$$\begin{aligned} S^+(S^+)^T &= (\tilde{S} - (\lambda s + t s_k)\, a_j^T \tilde{S})(\tilde{S} - (\lambda s + t s_k)\, a_j^T \tilde{S})^T = \\ &= \tilde{S}\tilde{S}^T + (\omega\lambda^2 - 2\lambda)\, \tilde{S}\tilde{S}^T a_j a_j^T \tilde{S}\tilde{S}^T + \\ &+ (\omega\lambda^2 s_k^T a_j - \lambda s_k^T a_j - t + \omega\lambda t)\, s_k a_j^T \tilde{S}\tilde{S}^T + \\ &+ (\omega\lambda^2 s_k^T a_j - \lambda s_k^T a_j - t + \omega\lambda t)\, \tilde{S}\tilde{S}^T a_j s_k^T + \\ &+ (\omega\lambda^2 (s_k^T a_j)^2 + 2\omega\lambda t s_k^T a_j + \omega t^2)\, s_k s_k^T = \\ &= \tilde{S}\tilde{S}^T - \frac{1}{s^T a_j}\, \tilde{S}\tilde{S}^T a_j a_j^T \tilde{S}\tilde{S}^T - \\ &- \frac{s_k^T a_j}{s^T a_j}(s_k a_j^T \tilde{S}\tilde{S}^T + \tilde{S}\tilde{S}^T a_j s_k^T) + \frac{s^T a_j - (s_k^T a_j)^2}{s^T a_j}\, s_k s_k^T = \\ &= \tilde{S}\tilde{S}^T + s_k s_k^T - \frac{(\tilde{S}\tilde{S}^T + s_k s_k^T)\, a_j a_j^T (\tilde{S}\tilde{S}^T + s_k s_k^T)}{s^T a_j} = \\ &= SS^T - \frac{SS^T a_j a_j^T SS^T}{a_j^T SS^T a_j} \end{aligned}$$

if we use the definition of s, t, ω and λ successively.

Now we shall prove the last part of the theorem. Since $s = SS^T a_j$ and $A^T S = 0$ by assumption, we obtain $A^T S^+ = 0$ from (3.4). Moreover $a_j^T S^+ = 0$ follows directly from (3.4) so that $(A^+)^T S^+ = 0$. We shall show that the matrix $[A^+, S^+]$ is nonsingular. The matrix S^+ has the same rank as the matrix $S^+(S^+)^T$. Suppose that $S^+(S^+)^T w = 0$ for some nonzero vector w. Then (3.3) implies that

$$S^+(S^+)^T w = SS^T(w - \mu a_j) = 0$$

where

$$\mu = \frac{a_j^T SS^T w}{a_j^T SS^T a_j}$$

and, by assumption, $w - \mu a_j = Au$ holds for some vector u. Now $w = Au + \mu a_j = A^+ u^+$ so that $\operatorname{rank}(A^+) + \operatorname{rank}(S^+) = n$. Moreover $(A^+)^T S^+ = 0$ has been proved so that the matrix $[A^+, S^+]$ is nonsingular. □

Theorem 3.1 is a generalization of the result proposed by Ritter in [7]. The matrix R^+ can be determined by the following theorem.

Theorem 3.2. Let A be a matrix whose columns are normals a_i, $i \in I(x)$ and R be an upper triangular matrix such that $R^T R = A^T A$. Let $A^+ = [A, a_j]$ where the normal a_j is not a linear combination of the normals a_i, $i \in I(x)$ and

$$R^+ = \begin{bmatrix} R, & r_1 \\ 0, & r_2 \end{bmatrix}$$

where

$$R^T r_1 = A^T a_j$$

and

$$r_2^2 = a_j^T a_j - r_1^T r_1$$

Then R^+ is an upper triangular matrix and $(R^+)^T R^+ = (A^+)^T A^+$.

Proof. See [3] for example. □

4. DELETING A CONSTRAINT FROM THE BASIS

We shall describe a construction of the matrices A^-, S^- and R^- for the case when the constraint with the normal a_j is deleted from the basis.

Lemma 4.1. Let A, S be an orthogonal pair of matrices generated by the set $I(x)$. Let A^-, S^- be an orthogonal pair of matrices generated, by the set $I^-(x) = I(x) \setminus \{j\}$ where $j \in I(x)$ such that $S^- = [S, s_0]$. Then

$$S^-(S^-)^T = SS^T + \frac{S^-(S^-)^T a_j a_j^T S^-(S^-)^T}{a_j^T S^-(S^-)^T a_j} \tag{4.1}$$

Proof. By the assumption $S^-(S^-)^T = SS^T + s_0 s_0^T$. Since $S^T a_j = 0$ and $s_0^T a_j \neq 0$ (see the proof of Theorem 2.1) we obtain $a_j^T S^-(S^-)^T a_j = a_j^T s_0 s_0^T a_j$ and $S^-(S^-)^T a_j = s_0 s_0^T a_j$ so that

$$s_0 = \frac{S^-(S^-)^T a_j}{s_0^T a_j} = \pm \frac{S^-(S^-)^T a_j}{\sqrt{(a_j^T S^-(S^-)^T a_j)}}$$

After substituting the last expression into the formula $S^-(S^-)^T = SS^T + s_0 s_0^T$ we obtain (4.1). □

Suppose that (3.1) and (4.1) hold. Then Lemma 3.2 implies (after changing the notation) that

$$S^-(S^-)^T = H - HA^-((A^-)^T HA^-)^{-1} (A^-)^T H$$

so that the second order information obtained in preceding iterations is kept.

Theorem 4.1. Let A, S be an orthogonal pair of matrices generated by the set $I(x)$ and R be an upper triangular matrix such that $R^T R = A^T A$. Let M be a permutation matrix which transfer the column a_j of the matrix A on the last position so that RM is an upper Hessenberg matrix. Let Q be an orthogonal matrix such that $QRM = \tilde{R}$ where $\tilde{R}$ is an upper triangular matrix. Denote

$$\tilde{R} = \begin{bmatrix} R^-, & \tilde{r}_1 \\ 0, & \tilde{r}_2 \end{bmatrix}, \quad e = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Let A^- be a matrix resulting from the matrix A after deleting the column a_j and $S^- = [S, s_0]$ where $s_0 = AM\tilde{R}^{-1}e$. Then $(R^-)^T R^- = (A^-)^T A^-$ and A^-, S^- is an orthogonal pair of matrices generated by the set $I^-(x) = I(x) \setminus \{j\}$ where $j \in I(x)$ such that $S^T s_0 = 0$.

Proof. From the assumed form of the matrix $\tilde{R}$ we have

$$\begin{bmatrix} (R^-)^T R^-, & (R^-)^T \tilde{r}_1 \\ \tilde{r}_1^T R^-, & \tilde{r}_1^T \tilde{r}_1 + \tilde{r}_2^2 \end{bmatrix} = \tilde{R}^T \tilde{R}$$

On the other hand we have

$$\begin{bmatrix} (A^-)^T A^-, & (A^-)^T a_j \\ a_j^T A^-, & a_j^T a_j \end{bmatrix} = (AM)^T AM = (QRM)^T QRM = \tilde{R}^T \tilde{R}$$

since $R^T R = A^T A$ and $QRM = \tilde{R}$. By comaring both above matrices we obtain $(R^-)^T R^- = (A^-)^T A^-$. Using the expression $s_0 = AM\tilde{R}^{-1}e$ we obtain $S^T s_0 = S^T AM\tilde{R}^{-1}e = 0$ since $S^T A = 0$. Furthermore

$$\begin{bmatrix} (A^-)^T \\ a_j^T \end{bmatrix} s_0 = (AM)^T AM\tilde{R}^{-1}e = M^T R^T RM(QRM)^{-1}e =$$
$$= (QRM)^T e = \tilde{R}^T e = \begin{bmatrix} 0 \\ \tilde{r}_2 \end{bmatrix}$$

since $R^T R = A^T A$ and $QRM = \tilde{R}$ so that $(A^-)^T s_0 = 0$. Set $S^- = [S, s_0]$. Then $(A^-)^T S^- = 0$ and $[A^-, S^-]$ is a nonsingular square matrix of order n since $(A^-)^T s_0 = 0$ and $S^T s_0 = 0$. □

Theorem 4.1 is a generalization of the result proposed by Gill and Murray in [4]. Note that the role of the orthogonal matrix Q used in Theorem 4.1 is to turn the subdiagonal elements of the upper Hessenberg matrix RM into zero. It can be a product of elementary Givens matrices. The orthogonal matrix Q is not used explicitly and need not be stored. The equality $S^T s_0 = 0$ is a necessary assumption of Theorem 2.1.

5. THE IMPLEMENTATION OF QUASI-NEWTON METHODS WITHOUT PROJECTIONS

In this section we shall describe an algorithm which is an implementation of quasi-Newton methods without projections for linearly constrained minimization. This new algorithm is a composition of results given in [7], [4] and [6]. It uses numerically stable QR factorization of the matrix A instead of its pseudoinverse used in [7]. Moreover it works with a nonorthogonal matrix S instead of an orthogonal one used in [4]. Before describing this algorithm we must state several notes:

1. The definition of the active constraints must be slightly modified so as to ensure the numerical stability of the algorithm. The constraint with the normal a_i will be assumed active if $|a_i^T x - b_i| \leqq \varepsilon_1$, where $\varepsilon_1 > 0$ is a small number.
2. The steplength ϱ must be chosen so as to satify the conditions $F^+ - F \leqq \varepsilon_4 \varrho s^T g$ and either $\varrho = \tilde{\varrho}$ or $F^+ - F \geqq (1 - \varepsilon_4) \varrho s^T g$, where $0 < 2\varepsilon_4 < 1$. The safeguarded cubic interpolation with the initial estimate $\varrho = \min(1, 4 (\tilde{F} - F)/s^T g)$ can be used, where $\tilde{F}$ is a lower bound of the minimum value of the objective function.
3. If $B + D \leqq 0$, the product form iteration (1.1) cannot be used without sacrificing the desired positive semidefiniteness of the matrix $S^+(S^+)^T$. In this case we use the product form of BFGS method so that

$$S^+ = S + \frac{1}{\tilde{y}^T \tilde{d}} S\tilde{d} \left(\sqrt{\left(\frac{\tilde{y}^T \tilde{d}}{\tilde{d}^T \tilde{d}} \right)} \tilde{d} - \tilde{y} \right)^T \tag{5.1}$$

4. The selection of the quasi-Newton method without projections is controlled by the value of the integer M in the same manner as in [6] (see note 4 in [6]).
5. The values α, β, γ, δ, ε, σ and τ must be scaled as in [6] (see note 5 in [6]).
6. The deletion of constraints from the basis will be controlled by an integer REM. For $REM = 1$ we remove a constraint only in the neighbourhood of the minimum on the linear manifold defined by active constraints. For $REM = 2$ we remove a constraint whenever a negative Lagrange multiplier occurs.
7. The algorithm requires an initial feasible point. It can be obtained by solving a linear programming problem.

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

Algorithm 5.1.

Step 1: Determine the initial feasible point x and compute values $F := F(x)$ and $g := g(x)$. Set $NEW := 0$ and $K := 0$.

Step 2: In the first iteration (when $K = 0$) go to step 3 else go to step 7.

Step 3: Restart. Suppose that $|a_i^T x - b_i| \leqq \varepsilon_1$, $1 \leqq i \leqq l$ and $|a_i^T x - b_i| > \varepsilon_1$, $l < i \leqq m$ (after the permutation of indices). Let A and R be empty matrices and $S = I$ (I is the unit matrix of order n). Set $j := 1$.

Step 4: If $j \leqq l$ go to step 5 else set $\tilde{g} := S^{\mathrm{T}}g$ and go to step 6.

Step 5: Let A^+, S^+ and R^+ be matrices determined from the matrices A, S and R by Theorems 3.1 and 3.2. Set $A := A^+$, $S := S^+$ and $R := R^+$. Set $j := j + 1$ and go to step 4.

Step 6: Set $L := 0$. Set $\tilde{u} := \tilde{g}$ and go to step 21.

Step 7: Determine $\tilde{y} := S^{\mathrm{T}}(g - g_1)$ and $\tilde{v} := -\varrho\tilde{g} - \tilde{y}$ and compute $\tau := \tilde{v}^{\mathrm{T}}\tilde{v}$. If $\tau \leqq 0$ go to step 3 else go to step 8.

Step 8: Compute $\varepsilon := \tilde{u}^{\mathrm{T}}\tilde{u}$. If $\varepsilon \leqq 0$ go to step 9 else go to step 10.

Step 9: If $L = 0$ go to step 3 else go to step 14.

Step 10: Set $\lambda := \sqrt{(\tau/\varepsilon)}$, determine $\tilde{u} := \lambda\tilde{u}$ and compute $\alpha := \tilde{u}^{\mathrm{T}}\tilde{y}$, $\beta := \tilde{v}^{\mathrm{T}}\tilde{y}$ and $\sigma := \tilde{u}^{\mathrm{T}}\tilde{v}$. If $\beta = 0$ go to step 19 else go to step 11.

Step 11: Set $\alpha := \alpha/\tau$, $\beta := \beta/\tau$, $\sigma := \sigma/\tau$, $\gamma := \alpha + \sigma$, $\delta := \beta + 1$ and $\omega := 1 - \sigma^2$ If $\omega \leqq 0$ go to step 9 else go to step 12.

Step 12: Set $\mathrm{A} := \beta^2\omega$, $\mathrm{B} := \beta\delta\omega$ and $\mathrm{D} := (\beta\sigma - \alpha)^2$. If $\mathrm{B} + \mathrm{D} \leqq 0$ go to step 13 else go to step 15.

Step 13: If $L = 0$ go to step 19 else go to step 14.

Step 14: Set $L = 0$. Set $\tilde{u} := \tilde{g}$ and go to step 10.

Step 15: Choose the value of the parameter φ according to the integer value M (see note 4 above). If either $\varphi < 0$ or $\varphi > 10^4$ go to step 16 else go to step 17.

Step 16: If $\beta\delta \leqq 0$ go to step 13 else set $\varphi := 0$ and go to step 17.

Step 17: Set $q := (\delta - \varphi(\mathrm{B} + \mathrm{D}))/\beta$. If $q \leqq 0$ go to step 3 else go to step 18.

Step 18: Let $\tilde{u}^+$ be a vector and S^+ be a matrix determined from (1.1). Set $\tilde{u} := \tilde{u}^+$ and $S := S^+$. Set $L := 1$ and go to step 21.

Step 19: Determine $\tilde{d} := -\varrho\tilde{g}$ and compute $\sigma := \tilde{d}^{\mathrm{T}}\tilde{y}$ and $\varepsilon := \tilde{d}^{\mathrm{T}}\tilde{d}$. If either $\sigma \leqq 0$ or $\varepsilon \leqq 0$ go to step 3 else go to step 20.

Step 20: Let S^+ be a matrix determined from (5.1). Set $S := S^+$ and go to step 6.

Step 21: If $NEW = 0$ go to step 23 else go to step 22.

Step 22: Set $j := NEW$. Let A^+, S^+ and R^+ be matrices determined from the matrices A, S and R by Theorems 3.1 and 3.2. Set $A := A^+$, $S := S^+$ and $R := R^+$.

Step 23: Compute Lagrange multiplier vector u from (2.2) and determine the index j of minimum Lagrange multiplier u_j. Set $OLD := j$ and $c := \max(0, -u_j)$. If $REM = 1$ go to step 26. If $REM = 2$ go to step 24.

Step 24: If $c \leqq \varepsilon_2$ go to step 26 else go to step 25.

Step 25: Set $j := OLD$. Let A^-, S^- and R^- be matrices determined from the matrices A, S and R by Theorem 4.1. Set $A := A^-$, $S := S^-$ and $R := R^-$ and go to step 23.

Step 26: If $\|g - Au\| \leqq \varepsilon_3\|g\|$ go to step 27 else go to step 28.

Step 27: If $c \leqq \varepsilon_2$ then stop else go to step 25.

Step 28: Determine $\tilde{g} := S^{\mathrm{T}}g$ and $s := -S\tilde{g}$. If $-s^{\mathrm{T}}g \leqq \varepsilon_0\|s\|\,\|g - Au\|$ go to step 3 else go to step 29.

Step 29: Determine the maximum steplength $\tilde{\varrho}$ from (2.1) and the index j of the constraint which becomes active for $\varrho = \tilde{\varrho}$. Set $NEW := j$.

Step 30: Set $x_1 := x$, $F_1 := F$ and $g_1 := g$. Use a standard procedure to determine the steplength ϱ (see note 2 above). Compute $x := x_1 + \varrho s$, $F := F(x)$ and $g := g(x)$.

Step 31: If $(\tilde{\varrho} - \varrho)\ \|s\| > \varepsilon_1$ then set $NEW := 0$ and go to step 32 else go to step 32.

Step 32: Set $K := K + 1$ and go to step 2.

Algorithm 5.1 uses integers K, L, REM, NEW and OLD. Here K is an iteration count, L is a working integer which indicates that the product form iteration (1.1) was successful, M is a parameter controlling the selection of a definite quasi-Newton method without projections specified by user, REM is a parameter controlling the strategy of removing constraints specified by user, NEW is the index of the constraint added to the basis and OLD is the index of the constraint deleted from the basis. Besides, Algorithm 5.1 uses some tolerances. The values $\varepsilon_0 = 10^{-3}$, $\varepsilon_1 = 10^{-7}$, $\varepsilon_2 = 10^{-10}$, $\varepsilon_3 = 10^{-5}$ and $\varepsilon_4 = 10^{-2}$ were used in the implementation of this algorithm on computer IBM 370/135 in double precision arithmetic.

6. NUMERICAL EXPERIMENTS

Efficiency of Algorithm 5.1 was tested by means of 14 examples proposed in [1] and [8]. Table 1 contains the original notation of these examples and minimum values reached by Algorithm 5.1.

Table 1.

	No. in [1]	No. in [8]	Reached minimum value
1	11·8	62	26272·510369
2	11·1	53	4·0930232558
3	—	112	−47·761090858
4	12·6	—	10^{-19}
5	12·12	—	0·050426187894
6	12·13	—	0·050426187894
7	12·2	21	99·96
8	12·21	24	1·0
9	12·22	37	3456·0
10	12·14	41	1·9259259259
11	12·1	45	1·0
12	12·17	86	−32·348678966
13	12·26	—	−5280335·1332
14	12·20	119	244·89969752

Results of the tests are shown in Tables 2 and 3. Each column in the Tables 2 and 3 corresponds to a value of the integer M (choice of quasi-Newton method without projections). Each row in Tables 3 and 4 corresponds to one example (numbers 1−14 agree with numbers in the Table 1). A pair of values in the Tables 3 and 4 which are separated by a stoke are the number of iterations and the number of function evaluations. The second value shows the efficiency of the algorithm and it ought to be as small as possible. An asterisk in the row 6 shows that an alternative solution with value $F = 4.941\ 229\ 3180$ was found.

Table 2.

	Quasi-Newton methods without projections: $REM = 1$					
	$M = 1$	$M = 2$	$M = 3$	$M = 4$	$M = 5$	$M = 6$
1	7—11	7—11	6—10	8—12	7—11	7—11
2	3—6	3—6	3—6	3—6	5—9	5—9
3	42—52	44—52	41—51	57—69	42—51	41—51
4	10—21	10—21	10—21	10—21	10—21	10—21
5	2—8	2—8	2—8	2—8	2—8	2—8
6	7—19*	8—28	7—20*	8—28	8—28	8—28
7	1—4	1—4	1—4	1—4	1—4	1—4
8	2—10	2—10	2—10	2—10	2—10	2—10
9	2—5	2—5	2—5	2—5	2—5	2—5
10	10—10	10—10	10—10	9—9	9—9	9—9
11	3—12	3—12	3—12	3—12	3—12	3—12
12	8—11	8—12	8—12	8—12	8—11	8—11
13	5—7	5—7	5—7	5—7	5—7	5—7
14	20—19	19—17	19—17	20—18	20—18	21—19

Table 3.

	Quasi-Newton methods without projections: $REM = 2$					
	$M = 1$	$M = 2$	$M = 2$	$M = 4$	$M = 5$	$M = 6$
1	7—11	7—11	6—10	7—11	7—11	8—12
2	3—6	3—6	3—6	5—9	5—9	3—6
3	35—50	31—42	31—46	30—40	31—39	50—68
4	10—21	10—21	10—21	10—21	10—21	10—21
5	2—8	2—8	2—8	2—8	2—8	2—8
6	7—16*	7—16*	7—17*	7—16*	7—16*	6—16*
7	1—4	1—4	1—4	1—4	1—4	1—4
8	2—10	2—10	2—10	2—10	2—10	2—10
9	2—5	2—5	2—5	2—5	2—5	2—5
10	8—8	8—8	8—8	8—8	8—8	8—8
11	3—12	3—12	3—12	3—12	3—12	3—12
12	8—10	8—10	8—10	8—10	8—10	13—18
13	5—7	5—7	5—7	5—7	5—7	5—7
14	17—12	18—14	17—13	17—12	17—12	21—20

Tables 2 and 3 differ only in the value of the integer *REM* used in Algorithm 5.1.

(Received October 20, 1981.)

REFERENCES

[1] J. T. Betts: An accelerated multiplier method for nonlinear programming. J. Optim. Theory Appl. *21* (1977), 2, 137—174.

[2] W. C. Davidon: Optimally conditioned optimization algorithms without line searches. Math. Programming *9* (1975), 1,1—30.

[3] P. E. Gill, W. Murray: A numerically stable form of the simplex algorithm. Linear Algebra Appl. *7* (1973), 2, 99—138.

[4] P. E. Gill, W. Murray: Newton-type methods for unconstrained and linearly constrained optimization. Math. Programming *7* (1974), 3, 311—350.

[5] D. Goldfarb: Extension of Davidon's variable metric method to maximization under inequality and equality linear constraints. SIAM J. Appl. Math. *17* (1969), 4, 739—764.

[6] L. Lukšan: Quasi-Newton methods without projections for unconstrained minimization. Kybernetika *18* (1892), 4, 290—306.

[7] K. Ritter: A variable metric method for linearly constrained minimization problems. In: Nonlinear Programming 3 (O. L. Mangasarian, R. R. Meyer, S. M. Robinson eds.). Academic Press, London 1978.

[8] W. Hock, K. Schittkowski: Test Examples for Nonlinear Programming Codes. (Lecture Notes in Economics and Mathematical Systems 187.) Springer - Verlag, Berlin—Heidelberg—New York 1981.

Ing. Ladislav Lukšan, CSc., Středisko výpočetní techniky ČSAV (General Computing Centre — Czechoslovak Academy of Sciences), Pod vodárenskou věží 4, 182 07 Praha 8. Czechoslovakia.