KYBERNETIKA — VOLUME 22 (1986), NUMBER 1

CONJUGATE DIRECTION ALGORITHMS FOR EXTENDED CONIC FUNCTIONS

LADISLAV LUKŠAN

The original conjugate direction methods were developed in such a way that they find a minimum of a quadratic function after a finite number of steps making use of perfect line searches. These methods have been frequently modified and improved by many authors. Some previously proposed modifications of the conjugate direction methods minimize extended quadratic functions after a finite number of steps. This paper describes the conjugate direction methods which minimize extended conic functions after a finite number of steps.

1. INTRODUCTION

Consider the class of objective functions of the form

$$F(x) = \varphi(\tilde{F}(x), l(x)) \tag{1.1}$$

where $\tilde{F}$: $R_n \to R$ is a quadratic function with the constant positive definite Hessian matrix $\tilde{G}$, l: $R_n \to R$ is a linear function with the constant gradient c and φ: $R_2 \to R$ is a twice continuously differentiable function. Define

$$\begin{aligned} \sigma(x) &= \frac{\partial\varphi(\tilde{F}(x), l(x))}{\partial\tilde{F}}, \\ \tau(x) &= \frac{\partial\varphi(\tilde{F}(x), l(x))}{\partial l}, \end{aligned} \tag{1.2}$$

and suppose that $\sigma(x) > 0$ for all $x \in X \subset R_n$ where either $X = R_n$ or X is a sufficiently large subset of R_n such that all points $x \in R_n$ considered in the next sections lie in X. These functions generalize a class of so called conic functions that were introduced by Bjørstad and Nocedal [2] for line search and by Davidon [5] and Sorensen [22], who had used them for the construction of a new class of the variable metric methods for unconstrained minimization. Therefore they will be called the extended conic functions. Note that the functions (1.1) also generalize a class of the extended quadratic functions that were introduced by Davison and Wong [6], and by Spedicato [23] and that can be obtained from (1.1) by setting $c = 0$.

In this paper conjugate directions methods for extended conic functions are described. The original conjugate direction methods were developed in such a way that they find a minimum of the quadratic function after a finite number of steps making use of perfect line searches (see [13], [11] for conjugate gradient method and [4], [10] for variable metric methods). These methods have been frequently modified and improved by many authors. Dixon [7] has proposed conjugate direction methods which find a minimum of the quadratic function after a finite number of steps without perfect line searches (see also [8], [20]). Sloboda [21], Shirey [19] and Abaffy and Sloboda [1] have generalized the conjugate gradient method in such a way that it finds a minimum of an extended quadratic function after a finite number of steps. Similar generalization of the variable metric methods has been described by Spedicato [23] and Flachs [9]. In the previous paper [16], the present author has proposed two modifications of the conjugate gradient method which minimize a conic function after a finite number of steps. Similar generalization of the variable metric methods has been described in [5], [22] and [17].

This paper generalizes previous results. It describes the conjugate direction methods which minimize both the extended quadratic function and the extended conic function after a finite number of steps. These methods use the values and the gradients of the objective function only and they need no information of the actual form of the function $\varphi: R_2 \to R$. Section 2 contains some results concerning the extended conic functions. It also contains a description of a new algorithm. Section 3 is devoted to the derivation and analysis of the basic conjugate direction methods for extended conic functions. Section 4 is devoted to the investigation of imperfect versions of the conjugate direction methods for extended conic functions.

2. SOME PROPERTIES OF EXTENDED CONIC FUNCTIONS

Consider the extended conic function (1.1). In order to simplify the notation, we omit the parameter x. We denote by F, g, G and $\tilde{F}$, $\tilde{g}$, $\tilde{G}$ the value, the gradient and the Hessian matrix of the function $F(x)$ and $\tilde{F}(x)$ respectively at the point $x \in R_n$. Furthermore, we denote by l and c the value and the gradient of the function $l(x)$.

Using (1.1) we get the following formulae

$$F = \varphi(\tilde{F}, l), \qquad g = \sigma \tilde{g} + \tau c \tag{2.1}$$

where $\sigma = \partial\varphi/\partial\tilde{F}$ and $\tau = \partial\varphi/\partial l$ with $\sigma > 0$. The vector c that appears in (2.1) can be determined by the following lemma.

Lemma 2.1. Let $F: R_n \to R$ be an extended conic function. Let $x \in R_n$ and let s_1

and s_2 be two linearly independent directions. Let the gradients $g = g(x)$, $g_{i1} = g(x + \alpha_{i1}s_i)$, and $g_{i2} = g(x + \alpha_{i2}s_i)$ be linearly independent for $1 \leqq i \leqq 2$. Furthermore let the gradients g_{12}, g_{22} and $g_{32} = g(x_{32})$ be linearly independent, where $x_{32} = \lambda_{12}x_{12} + \lambda_{22}x_{22}$ with $\lambda_{12} > 0$, $\lambda_{22} > 0$, and $\lambda_{12} + \lambda_{22} = 1$. Then $c \in C$ where

(2.2) $$C = \mathscr{L}(g, g_{11}, g_{12}) \cap \mathscr{L}(g, g_{21}, g_{22}) \cap \mathscr{L}(g_{12}, g_{22}, g_{32})$$

(Here $\mathscr{L}(\cdot, \cdot, \cdot)$ is the subspace spanned by its arguments).

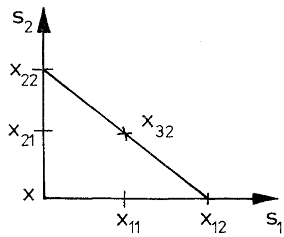

Fig. 1.

Proof. The situation is indicated in Figure 1. Since the points x, $x_{i1} = x + \alpha_{i1}s_i$. and $x_{i2} = x + \alpha_{i2}s_i$ lie on the straight line given by the direction s_i, we can write

$$g = \sigma\tilde{g} + \tau c ,$$
$$g_{i1} = \sigma_{i1}\tilde{g} + \alpha_{i1}\sigma_{i1}\tilde{G}s_i + \tau_{i1}c ,$$
$$g_{i2} = \sigma_{i2}\tilde{g} + \alpha_{i2}\sigma_{i2}\tilde{G}s_i + \tau_{i2}c ,$$

by (2.1) so that

$$\frac{g_{i1}}{\sigma_{i1}} - \frac{g}{\sigma} = \alpha_{i1}\tilde{G}s_i + \left(\frac{\tau_{i1}}{\sigma_{i1}} - \frac{\tau}{\sigma}\right) c ,$$

$$\frac{g_{i2}}{\sigma_{i2}} - \frac{g}{\sigma} = \alpha_{i2}\tilde{G}s_i + \left(\frac{\tau_{i2}}{\sigma_{i2}} - \frac{\tau}{\sigma}\right) c ,$$

and, consequently

$$\frac{1}{\alpha_{i2}}\left(\frac{g_{i2}}{\sigma_{i2}} - \frac{g}{\sigma}\right) - \frac{1}{\alpha_{i1}}\left(\frac{g_{i1}}{\sigma_{i1}} - \frac{g}{\sigma}\right) = \lambda_i c ,$$

where

$$\lambda_i = \frac{1}{\alpha_{i2}}\left(\frac{\tau_{i2}}{\sigma_{i2}} - \frac{\tau}{\sigma}\right) - \frac{1}{\alpha_{i1}}\left(\frac{\tau_{i1}}{\sigma_{i1}} - \frac{\tau}{\sigma}\right).$$

Therefore $c \in \mathscr{L}(g, g_{i1}, g_{i2})$ since $\lambda_i \neq 0$ follows from linear independence of the gradients g, g_{i1} and g_{i2}. Since the points x_{12}, x_{22}, and x_{32} lie also on a straight line, we get $c \in \mathscr{L}(g_{12}, g_{22}, g_{32})$ by the same considerations. □

Lemma 2.1 can be used for determination of the vector c in case $n \geqq 4$. We demonstrate it by the following example.

Example. Consider the objective function of the form

$$F(x) = \frac{x_1^2 + x_2^2 + x_3^2 + (x_4 + 1)^2}{x_3 + 1}.$$

Then

$$g(x) = \begin{bmatrix} 2x_1 \\ 2x_2 \\ 2x_3 \\ 2x_4 + 2 \end{bmatrix} \frac{1}{x_3 + 1} - \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \frac{x_1^2 + x_2^2 + x_3^2 + (x_4 + 1)^2}{(x_3 + 1)^2}$$

Let $x = [0, 0, 0, 0]^T$, $s_1 = [1, 0, 0, 0]^T$ and $s_2 = [0, 1, 0, 0]^T$. Let $\alpha_{i1} = 1$ and $\alpha_{i2} = 2$ for $1 \leqq i \leqq 2$ and let $x_{32} = x + s_1 + s_2$. Then

$$g = [0, 0, -1, 2]^T, \quad g_{11} = [2, 0, -2, 2]^T, \quad g_{12} = [4, 0, -5, 2]^T,$$
$$g_{21} = [0, 2, -2, 2]^T, \quad g_{22} = [0, 4, -5, 2]^T, \quad g_{32} = [2, 2, -3, 2]^T,$$

so that

$$\mathscr{L}(g, g_{11}, g_{12})^\perp = \mathscr{L}([0, 1, 0, 0]^T) \equiv \mathscr{V}_1,$$
$$\mathscr{L}(g, g_{21}, g_{22})^\perp = \mathscr{L}([1, 0, 0, 0]^T) \equiv \mathscr{V}_2,$$
$$\mathscr{L}(g_{12}, g_{22}, g_{32})^\perp = \mathscr{L}([1, 1, 0, -2]^T) \equiv \mathscr{V}_3.$$

Therefore $C = (\mathscr{V}_1 \cup \mathscr{V}_2 \cup \mathscr{V}_3)^\perp = \mathscr{L}([0, 0, 2, 0]^T)$ and we can see that $c = [0, 0, 1, 0]^T \in C$.

The above example shows the process for determining a basis in C. This process consists in the determination of bases in $\mathscr{V}_1 = \mathscr{L}(g, g_{11}, g_{12})^\perp$, $\mathscr{V}_2 = \mathscr{L}(g, g_{21}, g_{22})^\perp$, and $\mathscr{V}_3 = \mathscr{L}(g_{12}, g_{22}, g_{32})^\perp$ and then in construction of a basis in $C = (\mathscr{V}_1 \cup \mathscr{V}_2 \cup \mathscr{V}_3)^\perp$. Note that this process cannot be used when $n \leqq 3$. If this is the case then usually $\mathscr{V}_1 = \mathscr{V}_2 = \mathscr{V}_3 = \emptyset$ so that $\dim C = n$. Therefore we always consider the extended quadratic model whenever $n \leqq 3$.

If the above procedure is used for the general objective function then usually $\mathscr{V}_1 \cup \mathscr{V}_2 \cup \mathscr{V}_3 = R_n$ and, consequently, $C = \emptyset$ for $n > 4$. Therefore we again use the more simple extended quadratic model if this situation arises.

Having the vector c, we can determine the ratios σ/σ_1 and σ/σ_2 by means of three point formula.

Lemma 2.2. Let $F: R_n \to R$ be an extended conic function. Let x, $x_1 = x + \alpha_1 s$, and $x_2 = x + \alpha_2 s$ be three different points such that the gradients g, g_1, and g_2 are linearly independent. Then

$$\frac{\sigma}{\sigma_1} = \frac{g_1^T P g g_2^T P g_2 - g_1^T P g_2 g_2^T P g}{g_1^T P g_1 g_2^T P g_2 - (g_1^T P g_2)^2} \frac{\alpha_2 - \alpha_1}{\alpha_2},$$

(2.3)

$$\frac{\sigma}{\sigma_2} = \frac{g_1^T P g_1 g_2^T P g - g_1^T P g g_2^T P g_1}{g_1^T P g_1 g_2^T P g_2 - (g_1^T P g_2)^2} \frac{\alpha_1 - \alpha_2}{\alpha_1},$$

where

$$P = I - \frac{cc^{\mathrm{T}}}{c^{\mathrm{T}}c}. \tag{2.4}$$

Proof. Using (2.1) we can write

$$g = \sigma\tilde{g} + \tau c,$$
$$g_1 = \sigma_1\tilde{g} + \alpha_1\sigma_1\tilde{G}s + \tau_1 c,$$
$$g_2 = \sigma_2\tilde{g} + \alpha_2\sigma_2\tilde{G}s + \tau_2 c.$$

Since $Pc = 0$, we get

$$\frac{Pg_1}{\sigma_1} - \frac{Pg}{\sigma} = \alpha_1 P\tilde{G}s,$$
$$\frac{Pg_2}{\sigma_2} - \frac{Pg}{\sigma} = \alpha_2 P\tilde{G}s,$$

so that

$$g_1^{\mathrm{T}}Pg_1\frac{\alpha_2}{\sigma_1} - g_1^{\mathrm{T}}Pg_2\frac{\alpha_1}{\sigma_2} = \frac{\alpha_2 - \alpha_1}{\sigma}g_1^{\mathrm{T}}Pg,$$
$$-g_2^{\mathrm{T}}Pg_1\frac{\alpha_2}{\sigma_1} + g_2^{\mathrm{T}}Pg_2\frac{\alpha_1}{\sigma_2} = \frac{\alpha_1 - \alpha_2}{\sigma}g_2^{\mathrm{T}}Pg.$$

The last two equations have the solution (2.3) provided

$$g_1^{\mathrm{T}}Pg_1g_2^{\mathrm{T}}Pg_2 - (g_1^{\mathrm{T}}Pg_2)^2 \neq 0. \tag{2.5}$$

But (2.5) is valid if and only if Pg_1 and Pg_2 are linearly independent (the Schwartz inequality). It certainty holds since in the opposite case the vectors $P\tilde{g}$ and $P\tilde{G}s$ have to be linearly dependent and, consequently, the vectors $\tilde{g}$, $\tilde{G}s$, and c have also to be linearly dependent, which contradicts the assumed linear independence of the vectors g, g_1, and g_2. □

Note that (2.3) turns into the well known formula for determination of parameters of the extended quadratic function when $P = I$ (see [19] and [9]).

Both Lemma 2.1 and Lemma 2.2 assume linear independence of three gradients $g(x)$, $g(x + \alpha_1 s)$, and $g(x + \alpha_2 s)$, say, lying on a straight line. This situation arises only if $\tilde{G}s$ is not parallel to the vectors c and $Pg(x + \alpha s) \neq 0$ for all $\alpha \in R$. The following lemma allows us to remove this requirement and gives the possibility of determining the parameter σ_2 by means of two point formula in the case when the conjugate directions are generated without perfect line searches.

Lemma 2.3. Let F: $R_n \to R$ be an extended conic function. Let x and $x_2 = x + \alpha_2 s$ be two different points and let $v \neq 0$ be a vector, which is conjugated to the vector s (i.e. $v^{\mathrm{T}}\tilde{G}s = 0$), such that $v^{\mathrm{T}}c = 0$ and $v^{\mathrm{T}}g \neq 0$. Then

$$\frac{\sigma_2}{\sigma} = \frac{v^{\mathrm{T}}g_2}{v^{\mathrm{T}}g}. \tag{2.6}$$

Proof. Using (2.1) we can write

$$g = \sigma \tilde{g} + \tau c ,$$
$$g_2 = \sigma_2 \tilde{g} + \alpha_2 \sigma_2 \tilde{G} s + \tau_2 c .$$

Since $v^T c = 0$ and $v^T \tilde{G} s = 0$, we get $v^T g = \sigma v^T \tilde{g}$ and $v^T g_2 = \sigma_2 v^T \tilde{g}$ so that (2.6) is valid provided $v^T g \neq 0$. □

Note that (2.6) can be used also for an extended quadratic function. The condition $v^T c = 0$ is satisfied automatically in this case.

The following lemma is essential for the minimization of an extended conic function. Note that it is unnecessary for extended quadratic function.

Lemma 2.4. Let $F: R_n \to R$ be an extended conic function. Let $x_1 \in R_n$ and $x_2 \in R_n$ be two different points such that $Pg_1 = 0$ and $Pg_2 = 0$. Let $x^* \in R_n$ be a critical point of the function $F(x)$. Then there exists a steplength α^* such that

$$x^* = x_2 + \alpha^*(x_2 - x_1) . \tag{2.7}$$

Proof. Since $Pg_1 = 0$ and $Pg_2 = 0$, we can write

$$g_1 = \sigma_1 \tilde{g}_1 + \tau_1 c = \lambda_1 c ,$$
$$g_2 = \sigma_2 \tilde{g}_2 + \tau_2 c = \lambda_2 c .$$

Therefore

$$\tilde{g}_1 = \tilde{\lambda}_1 c ,$$
$$\tilde{g}_2 = \tilde{\lambda}_2 c$$

for $\sigma_1 > 0$ and $\sigma_2 > 0$ so that

$$x_2 - x_1 = \tilde{G}^{-1}(\tilde{g}_2 - \tilde{g}_1) = (\tilde{\lambda}_2 - \tilde{\lambda}_1)\, \tilde{G}^{-1} c .$$

Note that $\tilde{\lambda}_2 \neq \tilde{\lambda}_1$ since we assume $x_2 \neq x_1$ ($\sigma_1 > 0$ and $\sigma_2 > 0$ follows from the definition of the extended conic function). But x^* is a critical point of the extended conic function $F(x)$ so that $g^* = 0$ and also $Pg^* = 0$. Therefore, we get

$$x^* - x_2 = \tilde{G}^{-1}(\tilde{g}^* - \tilde{g}_2) = (\tilde{\lambda}^* - \tilde{\lambda}_2)\, \tilde{G}^{-1} c$$

using the above considerations. Combining these equations, we obtain (2.7) where $\alpha^* = (\tilde{\lambda}^* - \tilde{\lambda}_2)/(\tilde{\lambda}_2 - \tilde{\lambda}_1)$. □

Let $x_1^0 \in R_n$ be an arbitrary point. If we minimize the extended conic function (1.1) subject to the single linear constraint $l(x) = l(x_1^0)$, we get the point $x_1 \in R_n$ such that $Pg_1 = 0$. Let $x_2^0 = x_1 + \alpha_1 c$ with $\alpha_1 \neq 0$. Then $l(x_2^0) \neq l(x_1^0)$ and if we minimize the extended conic function (1.1) subject to the constraint $l(x) = l(x_2^0)$, we get the point $x_2 \in R_n$ such that $x_2 \neq x_1$ and $Pg_2 = 0$. Using Lemma 2.4, we find the minimizer $x^* \in R_n$ of the extended conic function (1.1) (if it exists) by the formula (2.7) where α^* is a steplength chosen by the perfect line search. Since the functions $\varphi(\tilde{F}(x), l(x_1^0))$ and $\varphi(\tilde{F}(x), l(x_2^0))$ are extended quadratic functions and since we suppose that $\tilde{G} > 0$ and $\sigma(x) > 0$, the points $x_1 \in R_n$ and $x_2 \in R_n$ such that $Pg_1 = 0$

and $Pg_2 = 0$ always exist and they can be obtained by means of usual methods developed for linearly constrained minimization of the extended quadratic functions. Note that these methods have to be designed in such a way that they use no values of the function $\tau(x)$, since these values cannot be determined by the above theory.

Now we will describe an outer algorithm which summarizes our results. We denote by $\mathscr{M}(x, c)$ a linear manifold which contains the point $x \in R_n$ and which is orthogonal to the vector c so that $\mathscr{M}(x, c) = R_n$ in case $c = 0$.

Algorithm 2.1.

Step 1: Determine an initial point $x \in R_n$ and compute the value $F := F(x)$ and the gradient $g := g(x)$. Set $k := 0$.

Step 2: If the termination criteria are satisfied (for example if $\|g\|$ is sufficiently small) then stop.

Step 3: If $k = 0$ then compute the vector c by means of Lemma 2.1 and go to Step 4. If $k = 1$ then set $x_1 := x$, $s := -g$ and go to Step 5. If $k = 2$ then go to Step 4. If $k = 3$ then set $x_2 := x$, $s := -\text{sgn}\,(g^{\mathrm{T}}(x_2 - x_1))\,(x_2 - x_1)$ and go to Step 6.

Step 4: If $c \neq 0$ then set $k := k + 1$. Using some conjugate direction subalgorithm, determine a minimizer $x^* \in R_n$ of the extended conic function $F\colon R_n \to R$ on the linear manifold $\mathscr{M}(x, c)$ and compute the value $F^* := F(x^*)$ and the gradient $g^* := g(x^*)$. Go to Step 7.

Step 5: Set $k := k + 1$. Using an imperfect line search procedure, determine the the point $x^* = x + \alpha^* s$ such that $F(x^*) < F(x)$ and compute the value $F^* := F(x^*)$ and the gradient $g^* := g(x^*)$. Go to Step 7.

Step 6: Set $k := 0$. Using a perfect line search procedure, determine the point $x^* = x + \alpha^* s$ such that $x^* = \arg\min_{\alpha \geqq 0} F(x + \alpha s)$ and compute the value $F^* := F(x^*)$ and the gradient $g^* := g(x^*)$.

Step 7: Set $x := x^*$, $F := F^*$, $g := g^*$ and go to Step 2.

Comments.

1) The vector c is computed by means of Lemma 2.1. If the computations are not exact then usually $\dim C \neq 1$. Therefore we have to use some tolerances in order to exclude the almost linearly dependent vectors. Note that $\dim C = 1$ and, consequently, $c \neq 0$ indicates that $F : R_n \to R$ is probably an extended conic function. If the objective function has a general form then usually $c = 0$ and the algorithm reduces to one for minimizing an extended quadratic function.

2) The vectors s_1 and s_2 which appears in Lemma 2.1 can be chosen arbitrarily. But it is advantageous to use them in the minimization process. Therefore we can set $c = 0$ initially and use a conjugate direction subalgorithm as in Step 4 to generate the points x, $x_{11} = x + \alpha_{11} s_1$, $x_{12} = x + \alpha_{12} s_1$, $x_{21} = x_{12} + \alpha_{21} s_2$, $x_{22} = x_{12} +$

$+ \alpha_{22}s_2$. Then $c \in C$ where C is given by (2.2) while g_{32} is the gradient computed at the additional point $x_{32} = (x_{12} + x_{22})/2$. If $c = 0$, then an extended conic function is not a good model for the objective function and we continue in a conjugate direction subalgorithm with $c = 0$. If $c \neq 0$ then $F: R_n \to R$ is probably an extended conic function and we go to Step 4.

3) The minimizer $x^* \in R_n$ desired in Step 4 always exists and it is unique for an extended conic function since this function has the same behaviour as an extended quadratic function in the manifold $\mathcal{M}(x, c)$ and since $\tilde{G} > 0$ and $\sigma(x) > 0$.

4) Step 5 serves for the determination of the point $x^* \in R_n$ such that $l(x^*) \neq l(x)$. It is advantageous to use an imperfect line search procedure for this purpose since it reduces the value of the objective function.

5) The result of Step 6 depends on the behaviour of the extended conic function. If the extended conic function has several critical points then all these points lie on the line defined by x ans s. On the other hand, if the line search procedure fails in Step 6 then the extended conic function has no minimizer. This is the only case when the line search can fail in the algorithm since it cannot fail in the manifold $\mathcal{M}(x, c)$.

Lemma 2.4 shows that Algorithm 2.1 finds a minimizer of an extended conic function (if it exists) after at most two cycles provided the conjugate direction subalgorithm used in Step 4 gives the required extremal points. The conjugate direction subalgoritms which can be used in Step 4 are described in the next two sections. Note that Algorithm 2.1 uses the values and the gradients of the objective function only and it need no information of the actual form of the function $\varphi: R_n \to R$.

3. BASIC CONJUGATE DIRECTION METHODS

Let $\mathcal{M}(x_1, c)$ be a linear manifold which contains the point $x_1 \in R_n$ and which is orthogonal to the vector c. Denote by m the dimension of the manifold $\mathcal{M}(x_1, c)$. Note that $m = n$ if $c = 0$ and $m = n - 1$ if $c \neq 0$. The basic conjugate direction methods for minimizing extended conic functions over the manifold $\mathcal{M}(x_1, c)$ are based on the iterative scheme

(3.1) $$x_{i+1} = x_i + \alpha_i s_i\,, \quad i \in N$$

where s_i, $i \in N$, are direction vectors orthogonal to the vector c so that

(3.2) $$s_i^{\mathrm{T}} c = 0\,, \quad i \in N$$

and where α_i, $i \in N$, are steplengths chosen by perfect line searches so that

(3.3) $$s_i^{\mathrm{T}} g_{i+1} = 0\,, \quad i \in N$$

(N is the set of natural numbers).

The following lemma is essential for the basic conjugate direction methods.

Lemma 3.1. Let $F: R_n \to R$ be an extended conic function. Consider the iterative scheme (3.1)–(3.3). Let the direction vectors s_i, $1 \leqq i \leqq m$, be nonzero and mutually conjugate (i.e. $s_i \neq 0$, $s_j \neq 0$ and $s_i^T \tilde{G} s_j = 0$ for $1 \leqq i < j \leqq m$). Then $Pg_{m+1} = 0$ where $P = I$ for $c = 0$ or $P = I - cc^T/c^Tc$ for $c \neq 0$.

Proof. See [16]. □

Conjugate directions, assumed in lemma 3.1, can be generated by the conjugate gradient method described in [16] which is a generalization of the methods proposed in [13] and [11]. In this case,

(3.4)
$$s_1 = -Pg_1$$

and

$$s_i = -Pg_i + \frac{y_{i-1}^T P g_i}{y_{i-1}^T d_{i-1}} d_{i-1}$$

for $1 < i \leqq m$ where $P = I$ for $c = 0$ or $P = I - cc^T/c^Tc$ for $c \neq 0$ and where

(3.5)
$$d_{i-1} = x_i - x_{i-1} = \alpha_{i-1} s_{i-1}$$

$$y_{i-1} = \frac{g_i}{\sigma_i} - \frac{g_{i-1}}{\sigma_{i-1}}$$

Therefore we can use the following algorithm.

Algorithm 3.1. (Conjugate gradient method.)

Step 1: Having the point $x \in R_n$, the value $F = F(x)$ and the gradient $g = g(x)$, choose an initial value $\sigma = \sigma(x)$ (usually set $\sigma := 1$). If $c = 0$ then set $m := n$ and $P := I$, else set $m := n - 1$ and $P := I - cc^T/c^Tc$. Set $k := 0$ and $l := 0$.

Step 2: If the termination criteria are satisfied (for example if $\|Pg\|$ is sufficiently small) then set $l := 1$. If $l \geqq 1$ then stop.

Step 3: If $k = 0$ then set $s := -Pg$ else set

$$s := -Pg + \frac{y^T P g}{y^T d} d .$$

Step 4: Use a perfect line search procedure to determine two points $x_1 := x + \alpha_1 s$, $x_2 := x + \alpha_2 s$ such that $s^T g(x_2) = 0$. Compute the values $F_1 := F(x_1)$, $F_2 := F(x_2)$ and the gradients $g_1 := g(x_1)$, $g_2 := g(x_2)$.

Step 5: If the gradients g, g_1, and g_2 are linearly dependent then set $l = 2$ else compute the value $\sigma_2 := \sigma(x_2)$ by Lemma 2.2 (see (2.3)) and set

$$d := x_2 - x ,$$

$$y := \frac{g_2}{\sigma_2} - \frac{g}{\sigma} .$$

Step 6: Set $k := k + 1$. If $k \geqq n$ then set $k := 0$. If $l = 0$ then set $\sigma := \sigma_2$. Set $x := x_2$, $F := F_2$, $g := g_2$ and go to Step 2.

Conjugate directions orthogonal to the vector c can be also generated by the variable metric methods proposed in [12]. These methods use the direction vectors

(3.6) $$s_i = -\frac{1}{\sigma_i} H_i g_i, \quad i \in N$$

where $H_1 = P$ and

(3.7) $$H_{i+1} = H_i + \frac{d_i d_i^T}{y_i^T d_i} - \frac{H_i y_i (H_i y_i)^T}{y_i^T H_i y_i} + $$
$$+ \frac{\vartheta_i}{y_i^T H_i y_i} \left(\frac{y_i^T H_i y_i}{y_i^T d_i} d_i - H_i y_i \right) \left(\frac{y_i^T H_i y_i}{y_i^T d_i} d_i - H_i y_i \right)^T$$

for $i \in N$ (Broyden's class, see [3]). Here ϑ_i is the value of a free parameter and

(3.8) $$d_i = x_{i+1} - x_i = \alpha_i s_i,$$
$$y_i = \frac{g_{i+1}}{\sigma_{i+1}} - \frac{g_i}{\sigma_i}$$

for $i \in N$. Most frequently used variable metric methods correspond to the values $\vartheta_i = 0$ (DFP method) or $\vartheta_i = 1$ (BFGS method). Note that

$$d_i^T y_i = d_i^T \left(\frac{g_{i+1}}{\sigma_{i+1}} - \frac{g_i}{\sigma_i} \right) = \frac{\alpha_i}{\sigma_i^2} g_i^T H_i g_i > 0$$

if $Pg_i \neq 0$, which is a necessary assumption for the positive definiteness of the matrix (3.7). The variable metric methods can be realized by the following algorithm.

Algorithm 3.2. (Variable metric methods.)

Step 1: Having the point $x \in R_n$ the value $F = F(x)$ and the gradient $g = g(x)$, choose an initial value $\sigma = \sigma(x)$ (usually set $\sigma := 1$). If $c = 0$ then set $m := n$ and $P := I$ else set $m := n - 1$ and $P := I - cc^T/c^Tc$. Set $k := 0$ and $l := 0$.

Step 2: If the termination criteria are satisfied (for example if $\|Pg\|$ is sufficiently small) then set $l := 1$. If $l \geqq 1$ then stop.

Step 3: If $k = 0$ then set $H := P$ else set

$$H := H + \frac{dd^T}{y^T d} - \frac{Hy(Hy)^T}{y^T H y} + \frac{\vartheta}{y^T H y} \left(\frac{y^T H y}{y^T d} d - Hy \right) \left(\frac{y^T H y}{y^T d} d - Hy \right)^T$$

for a given value of the parameter ϑ. Set $s := -H(g/\sigma)$.

Step 4: Use a perfect line search procedure to determine two points $x_1 := x + \alpha_1 s$, $x_2 := x + \alpha_2 s$ such that $s^T g(x_2) = 0$. Compute the values $F_1 := F(x_1)$, $F_2 := F(x_2)$ and the gradients $g_1 := g(x_1)$, $g_2 := g_2(x_2)$.

Step 5: If the gradients g, g_1, and g_2 are linearly dependent then set $l := 2$ else compute the value $\sigma_2 := \sigma(x_2)$ by Lemma 2.2 (see (2.3)) and set

$$d := x_2 - x\,,$$

$$y := \frac{g_2}{\sigma_2} - \frac{g}{\sigma}.$$

Step 6: Set $k := k + 1$. If $l = 0$ then set $\sigma := \sigma_2$. Set $x := x_2$, $F := F_2$, $g := g_2$ and go to Step 2.

Both Algorithm 3.1 and Algorithm 3.2 use a perfect line search procedure in Step 4. The following lemma shows that the perfect line search is reduced to the computation of three function values and three gradients only, in the case when $\tilde{G}s$ is not parallel to the vector c and, at the same time, $Pg(x + \alpha s) \neq 0$ for all $\alpha \in R$ (see comment after Lemma 2.2).

Lemma 3.2. Let $F := R_n \to R$ be an extended conic function. Let $s^T c = 0$ and let x, $x_0 = x + \alpha_0 s$, and $x_1 = x + \alpha_1 s$ be three different points such that the gradients g, g_0, and g_1 are linearly independent. Then $s^T g_2 = 0$ where $g_2 = g(x + \alpha_2 s)$ with

$$\alpha_2 = \frac{\alpha_1}{1 - \dfrac{s^T g_1}{s^T g} \dfrac{\sigma}{\sigma_1}}. \tag{3.9}$$

Proof. Since the gradients g, g_0, and g_1 are linearly independent, we can determine the ratio σ/σ_1 by Lemma 2.2. Since

$$g_2 = \sigma_2 \tilde{g} + \alpha_2 \sigma_2 \tilde{G}s + \tau_2 c$$

by (2.1) and since $s^T c = 0$ by the assumption, we get

$$s^T g_2 = \sigma_2 s^T \tilde{g} + \alpha_2 \sigma_2 s^T \tilde{G}s = 0$$

so that

$$\alpha_2 = -\frac{s^T \tilde{g}}{s^T \tilde{G}s}.$$

But

$$\frac{s^T g_1}{s^T g} = \frac{\sigma_1}{\sigma}\left(1 + \alpha_1 \frac{s^T \tilde{G}s}{s^T \tilde{g}}\right)$$

by (2.1). Thus (3.9) follows from the last two equalities. □

Note that the formula (3.9) has been already used for extended quadratic functions (see [9]). This formula cannot be used when either $Pg(x + \alpha^* s) = 0$ for some steplength $\alpha^* \in R$ or $\tilde{G}s$ is parallel to the vector c. Therefore we have to use the general line search procedure if this situation arises. Note also that the above exceptional cases terminate both Algorithm 1.1 and Algorithm 3.2. If $Pg(x + \alpha^* s) = 0$ for some steplength $\alpha^* \in R$ then, obviously, $\alpha^* = \alpha_2$ and the subalgorithm terminates with the value $l = 1$. If $\tilde{G}s$ is parallel to the vector c then the subalgorithm terminates with the value $l = 2$, which indicates that the required extermal point has not been found. But the direction s is parallel to the vector $\tilde{G}^{-1}c$ in this case, so that we have got a vector parallel to the difference $x_2 - x_1$ used in Step 6 of Algorithm 2.1. Therefore we can consider also the situation when the subalgorithm does not find the required extremal point in either cycle of the outer algorithm. Obviously, Algorithm 2.1 has to be slightly modified in this case.

4. THE IMPERFECT CONJUGATE DIRECTION METHODS

Let $\mathcal{M}(x_1, c)$ be a linear manifold defined in the previous section. Let $m = n$ if $c = 0$ or $m = n - 1$ if $c \neq 0$. The imperfect conjugate direction methods for minimizing extended conic functions over the manifold $\mathcal{M}(x_1, c)$ are based on the iterative scheme (3.1)–(3.2) where α_i, $i \in N$, are steplengths that are chosen by imperfect line searches. The following lemma is essential for the imperfect conjugate direction methods.

Lemma 4.1. Let $F: R_n \to R$ be an extended conic function. Consider the iterative scheme (3.1)–(3.2) with $\alpha_i \neq 0$ for $1 \leqq i \leqq m$. Let the direction vectors s_i, $1 \leqq i \leqq$ $\leqq m$ be nonzero and mutually conjugate (i.e. $s_i \neq 0$, $s_j \neq 0$ and $s_i^T \tilde{G} s_j = 0$ for $1 \leqq i < j \leqq m$). Let $x_{m+2} = x_{m+1} + \alpha_{m+1} s_{m+1}$ where $\alpha_{m+1} = 1$ and

$$s_{m+1} = -\sum_{i=1}^{m} \frac{1}{\sigma_{i+1}} \frac{d_i^T g_{i+1}}{d_i^T y_i} d_i$$

with d_i, $1 \leqq i \leqq m$, and y_i, $1 \leqq i \leqq m$, given by (3.8). Then $P g_{m+2} = 0$ where $P = I$ for $c = 0$ or $P = I - cc^T / c^T c$ for $c \neq 0$.

Proof. We confine ourselves to the case when $c \neq 0$ (the proof for $c = 0$ is similar). Since the direction vectors s_i, $1 \leqq i \leqq m$, are nonzero and mutually conjugate, we get

$$\sum_{i=1}^{m} \frac{s_i s_i^T}{s_i^T \tilde{G} s_i} = \tilde{G}^{-1} - \frac{\tilde{G}^{-1} c c^T \tilde{G}^{-1}}{c^T \tilde{G}^{-1} c}.$$

This equality can be easily verified by multiplying it by the linearly independent vectors $\tilde{G} s_i$, $1 \leqq i \leqq m$, and c. Therefore, we can find a minimizer x_{m+2} of both the quadratic function $\tilde{F}(x)$ and the extended conic function $F(x)$ on the manifold $\mathcal{M}(x_1, c)$ by the Newton step. Thus

$$x_{m+2} = x_{m+1} - \left(\tilde{G}^{-1} - \frac{\tilde{G}^{-1} c c^T \tilde{G}^{-1}}{c^T \tilde{G}^{-1} c} \right) \tilde{g}_{m+1} = \tag{4.1}$$

$$= x_{m+1} - \sum_{i=1}^{m} \frac{s_i s_i^T}{s_i^T \tilde{G} s_i} \tilde{g}_{m+1} = x_{m+1} - \sum_{i=1}^{m} \frac{d_i d_i^T}{d_i^T y_i} \frac{g_{i+1}}{\sigma_{i+1}}$$

since $\alpha_i \neq 0$ and $d_i^T y_j = 0$ for $i < j \leqq m$ by the assumption. □

Conjugate directions, assumed in Lemma 4.1, can be generated by the imperfect conjugate gradient method described in [16] which is a generalization of the methods described in [8] and [20]. In this case

$$s_1 = -h_1$$

(4.2) and

$$s_i = -h_i + \frac{y_{i-1}^T h_i}{y_{i-1}^T d_{i-1}} d_{i-1}$$

for $1 < i \leqq m$ where

$$h_1 = Pg_1$$

(4.3) and

$$h_i = Py_{i-1} - \frac{d_{i-1}^T y_{i-1}}{d_{i-1}^T h_{i-1}} h_{i-1}$$

for $1 < i \leqq m$ with d_{i-1}, $1 < i \leqq m$, and y_{i-1}, $1 < i \leqq m$, given by (3.5). As above, $P = I$ for $c = 0$ or $P = I - cc^T/c^Tc$ for $c \neq 0$. Note that $s_i \neq 0$ only if $h_i \neq 0$ (regular case). Using the above considerations, we can state the following algorithm.

Algorithm 4.1. (Imperfect conjugate gradient method.)

Step 1: Having the point $x \in R_n$, the value $F = F(x)$ and the gradient $g = g(x)$ choose an initial value $\sigma = \sigma(x)$ (usually set $\sigma := 1$). If $c = 0$ then set $m := m$ and $P := I$ else set $m := n - 1$ and $P := I - cc^T/c^Tc$. Set $k := 0$ and $l := 0$.

Step 2: If the termination criteria are satisfied (for example if $\|Pg\|$ is sufficiently small) then set $l := 1$. If $l \geqq 1$ then stop.

Step 3: If $k = 0$ then set $h := Pg$ and $u := 0$ else set

$$h := Py - \frac{d^T y}{d^T h} h$$

and

$$u := u - \frac{1}{\sigma} \frac{d^T g}{d^T y} d .$$

Step 4: If either $k = m$ or $\|h\| \leqq \varepsilon \|g\|$ then set $k := -1$.

Step 5: If $k < 0$ then set $s := u$ and go to Step 6. If $k = 0$ then set $s := -h$ and go to Step 6. If $k > 0$ then set

$$s := -h + \frac{y^T h}{y^T d} d$$

and go to Step 7.

Step 6: Use an imperfect line search procedure to determine two points $x_1 := x + \alpha_1 s$, $x_2 := x + \alpha_2 s$ such that $F(x_2) < F(x)$ and, moreover, $s^T g(x_2) \neq 0$ in the case when $k \geqq 0$. Compute the values $F_1 := F(x_1)$, $F_2 := F(x_2)$ and the gradients $g_1 := g(x_1)$, $g_2 := g(x_2)$. If the gradients g, g_1, and g_2 are linearly dependent then set $l := 2$ and continue with the line search to obtain the point $x_2 := x + \alpha_2 s$ such that $s^T g(x_2) = 0$ else compute the value $\sigma_2 := := \sigma(x_2)$ by Lemma 2.2 (see (2.3)). If $l = 0$ then go to Step 8 else go to Step 9.

Step 7: Use an imperfect line search procedure to determine the point $x_2 := x + \alpha_2 s$ such that $s^T g(x_2) \neq 0$ and $F(x_2) < F(x)$. Compute the value $F_2 := F(x_2)$ and the gradient $g_2 := g(x_2)$. Compute the value $\sigma_2 := \sigma(x_2)$ by Lemma 2.3 (see (2.6)).

Step 8: If $k \geqq 0$ then set

$$d := x_2 - x\,,$$

$$y := \frac{g_2}{\sigma_2} - \frac{g}{\sigma}.$$

Step 9: Set $k := k + 1$. If $l = 0$ then set $\sigma := \sigma_2$. Set $x := x_2$, $F := F_2$, and $g := g_2$. Set $v := s$ and go to Step 2.

Conjugate directions orthogonal to the vector c can be also generated by the projection methods. These methods use the direction vectors

$$s_i = -\frac{1}{\sigma_i} Q_i g_i \tag{4.4}$$

for $1 \leqq i \leqq m$ where $Q_1 = P$ and

$$Q_{i+1} = Q_i - \frac{Q_i y_i (Q_i y_i)^T}{y_i^T Q_i y_i} + \tag{4.5}$$

$$+ \frac{\vartheta_i}{y_i^T Q_i y_i} \left(\frac{y_i^T Q_i y_i}{y_i^T d_i} d_i - Q_i y_i \right) \left(\frac{y_i^T Q_i y_i}{y_i^T d_i} d_i - Q_i y_i \right)^T$$

for $1 \leqq i < m$ (Lenard's class, see [15]). Here ϑ_i is the value of a free parameter and d_i, y_i are vectors defined by (3.8) for $1 \leqq i < m$. Most frequently used projection method corresponds to the value $\vartheta_i = 0$ (see [24]). Note that $s_i \neq 0$ only if $Q_i g_i \neq 0$ (regular case). Furthermore

$$Q_i y_i = Q_i \tilde{G} d_i = -Q_i \tilde{G} Q_i g_i$$

so that (4.5) is defined for $Q_i g_i \neq 0$ and $\alpha_i \neq 0$. The projection methods can be realized by the following algorithm.

Algorithm 4.2. (Projection methods.)

Step 1: Having the point $x \in R_n$, the value $F = F(x)$, and the gradient $g = g(x)$, choose an initial value $\sigma = \sigma(x)$ (usually set $\sigma := 1$). If $c = 0$ then set $m := := n$ and $P := I$ else set $m := n - 1$ and $P := I - cc^T/c^Tc$. Set $k := 0$ and $l := 0$.

Step 2: If the termination criteria are satisfied (for example if $\|Pg\|$ is sufficiently small) then set $l := 1$. If $l \geqq 1$ then stop.

Step 3: If $k = 0$ then set $Q := P$ and $u := 0$ else set

$$Q := Q - \frac{Qy(Qy)^{\mathrm{T}}}{y^{\mathrm{T}}Qy} + \frac{\vartheta}{y^{\mathrm{T}}Qy}\left(\frac{y^{\mathrm{T}}Qy}{y^{\mathrm{T}}d}\, d - Qy\right)\left(\frac{y^{\mathrm{T}}Qy}{y^{\mathrm{T}}d}\, d - Qy\right)^{\mathrm{T}}$$

and

$$u := u - \frac{1}{\sigma}\frac{d^{\mathrm{T}}g}{d^{\mathrm{T}}y}\, d\,.$$

Step 4: If either $k = m$ or $\|Qg\| \leqq \varepsilon\|g\|$ then set $k := -1$.

Step 5: If $k < 0$ then set $s := u$ and go to Step 6. If $k = 0$ then set $s := -Qg$ and go to Step 6. If $k > 0$ then set $s := -Qg$ and go the Step 7.

Step 6: Use an imperfect line search procedure to determine two points $x_1 := x + \alpha_1 s$, $x_2 := x + \alpha_2 s$ such that $F(x_2) < F(x)$ and, moreover, $s^{\mathrm{T}} g(x_2) \neq 0$ in the case when $k \geqq 0$. Compute the values $F_1 := F(x_1)$, $F_2 := F(x_2)$ and the gradients $g_1 := g(x_1)$, $g_2 := g(x_2)$. If the gradients g, g_1, and g_2 are linearly dependent then set $l := 2$ and continue with the line search to obtain the point $x_2 := x + \alpha_2 s$ such that $s^{\mathrm{T}} g(x_2) = 0$ else compute the value $\sigma_2 := \sigma(x_2)$ by Lemma 2.2 (see (2.3)). If $l = 0$ then go to Step 8 else go to Step 9.

Step 7: Use an imperfect line search procedure to determine the point $x_2 := x + \alpha_2 s$ such that $s^{\mathrm{T}} g(x_2) \neq 0$ and $F(x_2) < F(x)$. Compute the value $F_2 := F(x_2)$ and the gradient $g_2 := g(x_2)$. Compute the value $\sigma_2 := \sigma(x_2)$ by Lemma 2.3 (see (2.6)).

Step 8: If $k \geqq 0$ then set

$$d := x_2 - x\,,$$

$$y := \frac{g_2}{\sigma_2} - \frac{g}{\sigma}\,.$$

Step 9: Set $k := k + 1$. If $l = 0$ then set $\sigma := \sigma_2$. Set $x := x_2$, $F := F_2$ and $g := g_2$. Set $v := s$ and go to Step 2.

Both Algorithm 4.1 and Algorithm 4.2 find a minimum of an extended conic function on the manifold $\mathscr{M}(x_1, c)$ after at most $m + 1$ imperfect steps in the regular case, when either $h_i \neq 0$ (Algorithm 4.1) or $Q_i g_i \neq 0$ (Algorithm 4.2) for $Pg_i \neq 0$. However, it is necessary to use the steplength $\alpha_{m+1} = 1$ in the last step. It is interesting that these algorithms minimize an extended quadratic function with no perfect line search in the regular case. Note that both Algorithm 4.1 and Algorithm 4.2 are designed in such a way that they can be used for an arbitrary objective function. Obviously, the finite step convergence does not appear in the general case.

The algorithms are described in this section can terminate either with the value $l = 1$ or with the value $l = 2$. These two cases has been analyzed in the previous section. Note that the above approach can be used for the modification of the other imperfect conjugate direction methods such as those proposed in [18] and [14].

(Received December 19, 1984.)

REFERENCES

[1] J. Abaffy and F. Sloboda: Imperfect conjugate gradient algorithms for extended quadratic functions. Numer. Math. *42* (1983), 97—105.

[2] P. Bjorstad and J. Nocedal: Analysis of a new algorithm for one-dimensional minimization. Computing *22* (1979), 93—100.

[3] C. G. Broyden: Quasi-Newton methods and their application to function minimization. Math. Comp. *21* (1967), 368—381.

[4] W. C. Davidon: Variable Metric Method for Minimization. Report ANL-5990 Rev., Argonne National Laboratories, Argonne, IL, 1959.

[5] W. C. Davidon: Conic approximations and collinear scalings for optimizers. SIAM J. Numer. Anal. *17* (1980), 268—281.

[6] E. J. Davison and P. Wong: A robust algorithm that minimizes l-functions. Automatica *11* (1975), 287—308.

[7] L. C. W. Dixon: Conjugate directions without linear searches. Inst. Math. Appl. *11* (1973), 317—328.

[8] L. C. W. Dixon: Conjugate gradient algorithm: quadratic termination properties without line searches. Inst. Math. Appl. *15* (1975), 9—18.

[9] J. Flachs: On the convergence, invariance, and related aspects of a modification of Huang's algorithm. Optim. Theory Appl. *37* (1982), 315—341.

[10] R. Fletcher and M. J. D. Powell. A rapidly convergent descent method for minimization. Comput. J. *6* (1963), 163—168.

[11] R. Fletcher and C. M. Reeves: Function minimization by conjugate gradients. Comput. J. *7* (1964), 149—154.

[12] D. Goldfarb: Extension of Davidon's variable metric method to maximization under linear inequality and equality constraints. SIAM J. Appl. Math. *17* (1969), 739—763.

[13] M. R. Hestenes and E. Stiefel: The method of conjugate gradients for solving linear systems. J. Res. Nat. Bur. Standards *49* (1952), 409—436.

[14] H. Y. Huang: Method of Dual Matrices for Function Minimization. Aero-Astronautic Report No. 88, Rice University, Houston, TX, 1972.

[15] M. L. Lenard: Accelerated Conjugate Direction Methods for Unconstrained Optimization. Technical Report No. MRC-1591, University of Wisconsin, Madison, WI, 1976.

[16] L. Lukšan: Conjugate gradient algorithms for conic functions. Aplikace matemat. (to appear).

[17] L. Lukšan: Variable metric methods for a class of extended conic functions. Kybernetika *21* (1985), 96—107.

[18] L. Nazareth: A conjugate direction algorithm without line searches. J. Optim. Theory Appl. *23* (1977), 373—387.

[19] J. E. Shirey: Minimization of extended quadratic functions. Numer. Math. *39* (1982), 157—161.

[20] F. Sloboda: An imperfect conjugate gradient algorithm. Aplikace matematiky *27* (1982), 426—434.

[21] F. Sloboda: A generalized conjugate gradient algorithm for minimization. Numer. Math. *35* (1980), 223—230.

[22] D. C. Sorensen: The Q-superlinear convergence of a collinear scaling algorithm for unconstrained optimization. SIAM J. Numer. Anal. *17* (1980), 84—114.

[23] E. Spedicato: A variable-metric method for function minimization derived from invariancy to nonlinear scaling. J. Optim. Theory Appl. *20* (1976), 315—329.

[24] G. Zoutendijk: Methods of Feasible Directions. Elsevier, Amsterdam 1960.

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