Aplikace matematiky

Ladislav Lukšan

Conjugate gradient algorithms for conic functions

Aplikace matematiky, Vol. 31 (1986), No. 6, 427–440

Persistent URL: http://dml.cz/dmlcz/104222

31 (1986) APLIKACE MATEMATIKY No. 6, 427—440

CONJUGATE GRADIENT ALGORITHMS FOR CONIC FUNCTIONS

LADISLAV LUKŠAN

(Received June 27, 1985)

Summary. The paper contains a description and an analysis of two modifications of the conjugate gradient method for unconstrained minimization which find a minimum of the conic function after a finite number of steps. Moreover, further extension of the conjugate gradient method is given which is based on a more general class of the model functions.

Keywords: Unconstrained optimization, conjugate gradient methods, conic functions, algorithms.

AMS classification: 65K10.

1. INTRODUCTION

The conjugate gradient method was introduced by Hestenes and Stiefel [9] for finding solutions of systems of linear equations with symmetric positive definite matrices and, lately, by Fletcher and Reeves [7] for unconstrained minimization. Since then it has been frequently modified and improved by many authors. Dixon [6] and Sloboda [13] have proposed conjugate gradient methods which use no perfect line search. Beale [2] and Powell [11] have described conjugate gradient methods with improved restart procedures. Fried [8], Boland et al. [4] and Kowalik et al. [10] have proposed further modifications of the conjugate gradient method which are based on some nonquadratic models. Most recent papers by Sloboda [14], Shirey [12] and Abaffy and Sloboda [1] generalize the previous results and give conjugate gradient methods which minimize the so called l-quadratic functions after a finite number of steps.

Another class of functions, which generalize the quadratic function, contains the so called conic functions. These functions were introduced by Bjørstad and Nocedal [3] for line search and by Davidon [5] and Sorensen [15], who used them for the construction of a new class of variable metric methods. In this paper, we propose new modifications of the conjugate gradient method, which minimize conic functions after a finite number of steps. Section 2 contains some results concerning conic functions. Section 3 is devoted to the derivation and analysis of a basic modification of the conjugate gradient method. It also contains a detailed description of a new algorithm. Section 4 is devoted to the investigation of an imperfect version of the conjugate gradient method. Section 5 proposes a further extension of the conjugate gradient method, which is based on a more general class of the model functions.

2. CONIC FUNCTIONS AND CONIC INTERPOLATIONS

Let R_n be an n-dimensional vector space. Let $\tilde{F}: R_n \to R$ be a quadratic function and $l: R_n \to R$ a linear function, both defined in the space R_n. Then the function

$$F(x) = \frac{\tilde{F}(x)}{l^2(x)} \tag{2.1}$$

defined in the open halfspace $X = \{x \in R_n: l(x) > 0\}$ is called a conic function.

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 $x \in X$. Furthermore, we denote by l abd c the value and the gradient of the function $l(x)$. Note that $\tilde{G}$ is a constant matrix and c is a constant vector. We assume throughout this paper that $\tilde{G}$ is a positive definite matrix.

Using (2.1) we get the formulae

$$\begin{aligned} F &= \frac{\tilde{F}}{l^2}, \\ g &= \frac{1}{l^2}\,\tilde{g} - \frac{2}{l^3}\,\tilde{F}c = \frac{1}{l^2}\,\tilde{g} - \frac{2}{l}\,Fc\,. \end{aligned} \tag{2.2}$$

Many properties of the conic functions have been described in the paper of Davidon [5]. We summarize one of his results in the following lemma.

Lemma 2.1. *Let $x \in X$ and $x_2 = x + \alpha_2 s \in X$ be two different points. Then*

$$\frac{l_2}{l} = \frac{\alpha_2 g^{\mathrm{T}} s}{F_2 - F - \varrho_2}, \tag{2.3}$$

where

$$\varrho_2^2 = (F_2 - F)^2 - \alpha_2^2 g^{\mathrm{T}} s g_2^{\mathrm{T}} s\,.$$

Lemma 2.1 gives the possibility of determining the ratio l_2/l from the values F and F_2 and from the gradients g and g_2 computed at two different points x and x_2.

Now we will prove a lemma, which will allow us to determine the vector c from the values F, F_1, and F_2 and the gradients g, g_1, and g_2 computed at three different points x, x_1, and x_2 lying on a line.

Lemma 2.2. *Let* $x \in X$, $x_1 = x + \alpha_1 s \in X$, and $x_2 = x + \alpha_2 s \in X$ *be three different points. Then*

$$c = -\frac{1}{2}\frac{(l_2^2 g_2 - l^2 g)\,\alpha_1 - (l_1^2 g_1 - l^2 g)\,\alpha_2}{(l_2 F_2 - lF)\,\alpha_1 - (l_1 F_1 - lF)\,\alpha_2}\,. \tag{2.4}$$

Proof. Since $\tilde{g}$, $\tilde{g}_1$, and $\tilde{g}_2$ are gradients of the quadratic function which has the Hessian matrix $\tilde{G}$, we can write, by (2.2),

$$\alpha_i \tilde{G} s = \tilde{g}_i - \tilde{g} = l_i^2 g_i - l^2 g + 2c(l_i F_i - lF)$$

for $1 \leqq i \leqq 2$. Therefore

$$\frac{l_2^2 g_2 - l^2 g}{\alpha_2} + 2c\,\frac{l_2 F_2 - lF}{\alpha_2} = \frac{l_1^2 g_1 - l^2 g}{\alpha_1} + 2c\,\frac{l_1 F_1 - lF}{\alpha_1}$$

which implies (2.4). □

Lemma 2.2 together with Lemma 2.1 offer the possibility of determining all parameters of the linear function $l(x)$. Therefore, we can compute the value and the gradient of the quadratic function $\tilde{F}(x)$ from the value and the gradient of the conic function $F(x)$, which is necessary for developing the conjugate gradient method.

We suppose throughout this paper that the conic function (2.1) is cupped (see [5]) which means that $g(x) = 0$ if and only if $x \in R_n$ is a minimizer of $F(x)$.

3. THE BASIC CONJUGATE GRADIENT METHOD

The conjugate gradient method for minimizing a conic function $F: X \to R$ over the open halfspace $X \subset R_n$ is based on the iterative scheme

$$x_{i+1} = x_i + \alpha_i s_i\,, \tag{3.1}$$

$i \in N = \{1, 2, \ldots\}$, where s_i is a direction vector and α_i is a steplength. We assume in this section that the steplengths are chosen by the perfect line searches so that

$$s_i^{\mathrm{T}} g_{i+1} = 0 \tag{3.2}$$

for $i \in N$. The following lemma is essential for conjugate direction methods.

Lemma 3.1. *Let* $F: X \to R$ *be a conic function. Consider the iterative scheme* (3.1) *with the steplengths chosen by the perfect line searches. Let the direction vectors satisfy the conditions* $s_i^{\mathrm{T}} \tilde{G} s_j = 0$ *for* $1 \leqq i < j \leqq k$ *and* $s_i^{\mathrm{T}} c = 0$ *for* $1 \leqq$ $\leqq i < k$ *with* $k \leqq n$. *Then*

$$s_i^{\mathrm{T}} g_{k+1} = 0 \tag{3.3}$$

for $1 \leqq i \leqq k$.

Proof. The equality $s_k^T g_{k+1} = 0$ follows from (3.2). Using (2.2) we get

$$s_i^T g_{k+1} = s_i^T \left(\frac{1}{l_{k+1}^2} \tilde{g}_{k+1} - \frac{2}{l_{k+1}} F_{k+1} c \right) = \frac{1}{l_{k+1}^2} s_i^T \tilde{g}_{k+1}$$

for $1 \leqq i < k$, since $s_i^T c = 0$ for $1 \leqq i < k$ by the assumption. Therefore $s_i^T g_{k+1} = 0$ if and only if $s_i^T \tilde{g}_{k+1} = 0$. Let $\tilde{y}_j = \tilde{g}_{j+1} - \tilde{g}_j = \alpha_j \tilde{G} s_j$ for $i < j \leqq k$. Then

$$s_i^T \tilde{g}_{k+1} = s_i^T \tilde{g}_{i+1} + \sum_{j=i+1}^{k} s_i^T \tilde{y}_j =$$

$$= l_{i+1}^2 s_i^T g_{i+1} + 2 l_{i+1} F_{i+1} s_i^T c + \sum_{j=i+1}^{k} \alpha_j s_i^T \tilde{G} s_j = 0$$

by (2.2) since $s_i^T c = 0$ and $s_i^T \tilde{G} s_j = 0$ for $i < j \leqq k$ by the assumption and $s_i^T g_{i+1} = 0$ by (3.2). □

Lemma 3.1 shows that the conjugate directions have to be generated in such a way that the first $n - 1$ of them lie in the subspace which is orthogonal to the vector c. Then $s_i^T g_{n+1} = 0$ for $1 \leqq i \leqq n$. If in addition $s_i \neq 0$ for $1 \leqq i \leqq n$, then $g_{n+1} = 0$ and, consequently, x_{n+1} is a minimizer of the conic function F. Let

$$P = I - \frac{cc^T}{c^T c} \tag{3.4}$$

be the orthogonal projection matrix associated with the subspace which is orthogonal to the vector c. It is advantageous to generate the direction vectors s_i, $1 \leqq i < n$ by orthogonalizing the projected gradients Pg_i, $1 \leqq i < n$. Since

$$Pg_i = P\left(\frac{1}{l_i^2} \tilde{g}_i - \frac{2}{l_i} Fc \right) = \frac{1}{l_i^2} P\tilde{g}_i$$

for $1 \leqq i < n$, we can derive the formulae for the direction vectors in the same way as in the case of minimization of a quadratic function with a single linear constraint. Thus we obtain

$$s_1 = -Pg_1 \tag{3.5}$$

and

$$s_i = -Pg_i + \frac{\tilde{y}_{i-1}^T P g_i}{\tilde{y}_{i-1}^T s_{i-1}} s_{i-1}$$

for $1 < i < n$. These vectors are different from zero in the regular case when $Pg_i \neq 0$ for $1 \leqq i < n$.

The direction vector s_n cannot be determined by the above scheme since $Pg_n = 0$ in the regular case (as follows from Lemma 3.1). Therefore g_n is parallel to the vector c and we have to use the general formula

$$s_n = c - \sum_{i=1}^{n-1} \frac{\tilde{y}_i^T c}{\tilde{y}_i^T s_i} s_i . \tag{3.6}$$

The following algorithm summarizes our results.

Algorithm 3.1.

Step 1: Determine an initial point x 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 set $s := -g$, $k := 1$, $l := 1$ and go to Step 4 else go to Step 5.

Step 4: Use an imperfect line search procedure to determine two points $x_1 := x + \alpha_1 s$, $x_2 := x + \alpha_2 s$. Compute the values $F_1 := F(x_1)$, $F_2 := F(x_2)$ and the gradients $g_1 := g(x_1)$, $g_2 := g(x_2)$ (suppose that $F_2 = \min(F, F_1, F_2)$). Compute the values

$$\varrho_1 := \sqrt{((F_1 - F)^2 - \alpha_1^2 g^T s g_1^T s)}\,,$$

$$\varrho_2 := \sqrt{((F_2 - F)^2 - \alpha_2^2 g^T s g_2^T s)}\,,$$

$$\tilde{l}_1 := \frac{\alpha_1 g^T s}{F_1 - F - \varrho_1}\,,$$

$$\tilde{l}_2 := \frac{\alpha_2 g^T s}{F_2 - F - \varrho_2}\,,$$

and the vector

$$c := -\frac{l}{2}\,\frac{(\tilde{l}_2^2 g_2 - g)\,\alpha_1 - (\tilde{l}_1^2 g_1 - g)\,\alpha_2}{(\tilde{l}_2 F_2 - F)\,\alpha_1 - (\tilde{l}_1 F_1 - F)\,\alpha_2}\,.$$

Go to Step 9.

Step 5: If $k = 1$ then set $u := c$ else set

$$u := u - \frac{y^T c}{y^T s}\, s\,.$$

Step 6: Set $v := (c^T g / c^T c)\, c - g$. If either $k = n$ or $v^T v \leqq \varepsilon g^T g$, set $s := -\operatorname{sgn}(g^T u)\, u$, $k := 0$ and go to Step 8.

Step 7: If $k = 1$ then set $s := v$ else set

$$s := v - \frac{y^T v}{y^T s}\, s\,.$$

Set $k := k + 1$ and continue.

Step 8: Use a perfect line search procedure to determine the point $x_2 := x + \alpha_2 s$ such that $s^T g(x_2) = 0$. Compute the value $F_2 := F(x_2)$ and the gradient $g_2 := g(x_2)$.

If $k \neq 0$, compute the values

$$\varrho_2 := \sqrt{((F_2 - F)^2 - \alpha_2^2 g^{\mathrm{T}} s g_2^{\mathrm{T}} s)}\,,$$

$$\bar{l}_2 := \frac{\alpha_2 g^{\mathrm{T}} s}{F_2 - F - \varrho_2}\,,$$

and the vector

$$y := (\bar{l}_2^2 g_2 - g) + \frac{2c}{l}(\bar{l}_2 F_2 - F)\,.$$

Step 9: Set $x := x_2$, $F := F_2$, $g := g_2$, $l := \bar{l}_2 l$ and go to Step 2.

Comments. 1) The algorithm is invariant under the initial scaling. Therefore we set $l = 1$ in Step 3.

2) We use the ratios $\bar{l}_1 = l_1/l$, $\bar{l}_2 = l_2/l$ instead of the values l_1, l_2 in Step 4 and Step 8.

3) The vector $\tilde{y} = \tilde{g}_2 - \tilde{g}$ appears both in the numerator and in the denominator of the formula for deriving the direction vector (see (3.5) and (3.6)). Therefore we use $y = \tilde{y}/l^2$ instead of $\tilde{y}$ in Step 7 of the algorithm.

4) Step 4 serves for the determination of the vector c only. Therefore it can be reduced to the computation of two values F_1, F_2 and two gradients g_1, g_2 provided ϱ_1, ϱ_2 exist and $l_1 > 0$, $l_2 > 0$.

Modifications. 1) The value of the linear function $l(x)$ remains unchanged in Step 8 of the algorithm when it is used for the conic function. Therefore we can easily set $\bar{l}_2 = 1$. However, the value $\bar{l}_2$ computed by means of the parameter ϱ_2 is useful for checking the suitability of the conic function as a model for the general objective function. In the case of perfect line search, we have $g_2^{\mathrm{T}} s = 0$ so that $\varrho_2 = |F_2 - F|$.

2) The vector c can be recomputed in Step 8 of the algorithm. In this case, both Step 4 and Step 8 are replaced by their combination which use a perfect line search procedure and compute the vector c from three values and three gradients of the objective function.

The following theorem is an immediate consequence of (3.5), (3.6), and Lemma 3.1.

Theorem 3.1. *Algorithm* 3.1 *finds a minimum of the conic function* $F: X \to R$ *with* $\tilde{G}$ *positive definite after* n *perfect steps in the regular case.*

Now we are analysing the singular case when $Pg_i = 0$ for some $i < n$. If this is the case then g_i is parallel to c and, consequently, x_i is a minimizer of the conic function $F(x)$ with the constraint $l(x) = l_i$. Therefore, it is also a minimizer of the quadratic function $\tilde{F}(x)$ with the same constraint and we can use the following lemma.

Lemma 3.2. *Let* $\tilde{F}(x)$ *be a quadratic function with a positive definite Hessian matrix* $\tilde{G}$. *Let* $\tilde{g}_i = \tilde{g}(x_i)$, $1 \leqq i \leqq 3$, *be the gradients of the function* $\tilde{F}(x)$ *at the* *points $x_i \in R_n$, $1 \leqq i \leqq 3$. Then $\tilde{g}_i$, $1 \leqq i \leqq 3$, are parallel only if x_i, $1 \leqq i \leqq 3$, lie on a line.*

Proof. We can write $\tilde{g}_i = \tilde{G}(x_i - \tilde{x})$, $1 \leqq i \leqq 3$, where $\tilde{x}$ is a minimizer of the function $\tilde{F}(x)$. The gradients $\tilde{g}_i$, $1 \leqq i \leqq 3$, are parallel only if

$$\tilde{g}_2 - \tilde{g}_1 = \tilde{G}(x_2 - x_1) = \lambda_2 c\,,$$
$$\tilde{g}_3 - \tilde{g}_1 = \tilde{G}(x_3 - x_1) = \lambda_3 c$$

for some vector $c \in R_n$. Therefore

$$x_2 - x_1 = \lambda_2 \tilde{G}^{-1} c\,,$$
$$x_3 - x_1 = \lambda_3 \tilde{G}^{-1} c$$

and the points x_i, $1 \leqq i \leqq 3$ lie on a line. □

Lemma 3.2 can be used in the singular case. Let $Pg_1 = 0$ and $Pg_2 = 0$ at two different points x_1 and x_2, respectively. Then $\tilde{g}_1 = \tilde{\lambda}_1 c$ and $\tilde{g}_2 = \tilde{\lambda}_2 c$ by (2.2) and (3.4). Let x_3 be a minimizer of the conic function. Then $g_3 = 0$ and, consequently, $\tilde{g}_3 = \tilde{\lambda}_3 c$ by (2.2). Therefore, using Lemma 3.2, we can write

$$x_3 = x_2 + \alpha(x_2 - x_1) \tag{3.7}$$

for some steplength α. The points x_1 and x_2 such that $Pg_1 = 0$ and $Pg_2 = 0$ can be obtained in two immediately consecutive cycles of the algorithm. Therefore we can find a minimizer of the conic function in the second cycle by the special step (3.7).

4. THE IMPERFECT CONJUGATE GRADIENT METHOD

Algorithm 3.1, described in the previous section, uses a perfect line search procedure in Step 8. If the objective function $F(x)$ is conic then the line search function $\varphi(\alpha) = F(x + \alpha s)$ is also conic and we can use the interpolation formula described in [3]. Thus we can find a minimizer α_2 of the function $\varphi(\alpha)$ by the formula

$$\alpha_2 = -\frac{\alpha_1}{\left(\dfrac{l_1}{l}\right)^3 \dfrac{g_1^{\mathrm{T}} s}{g^{\mathrm{T}} s} - 1} \tag{4.1}$$

so that the perfect line search procedure requires only two values F, F_1 and two gradients g, g_1 of the conic function computed at two different points x and $x_1 = x + \alpha_1 s$. Note that $l_1/l = 1$ when $s^{\mathrm{T}} c = 0$ so that (4.1) reduces to the quadratic interpolation formula in this case.

Now we are describing the algorithm which is based on the idea used in [13] and which does not use a perfect line search procedure when $s^{\mathrm{T}} c = 0$. Note that it allows us to save the computation of the value F_1 and the gradient g_1 when $s^{\mathrm{T}} c = 0$.

Theorem 4.1. *Let $F: X \to R$ be a conic function. Consider the iterative scheme*

$$x_{i+1} = x_i + \alpha_i s_i \tag{4.2}$$

$i \in N$, such that

$$s_1 = -h_1 \tag{4.3}$$

and

$$s_i = -h_i + \frac{\tilde{y}_{i-1}^{\mathrm{T}} h_i}{\tilde{y}_{i-1}^{\mathrm{T}} s_{i-1}} s_{i-1}$$

for $1 < i < n$ where

$$h_1 = Pg_1 \tag{4.4}$$

and

$$h_i = P\tilde{y}_{i-1} - \frac{s_{i-1}^{\mathrm{T}} \tilde{y}_{i-1}}{s_{i-1}^{\mathrm{T}} h_{i-1}} h_{i-1}$$

for $1 < i < n$ with

$$\tilde{y}_{i-1} = l_i^2 g_i - l_{i-1}^2 g_{i-1} + 2c(l_i F_i - l_{i-1} F_{i-1}) \tag{4.5}$$

for $1 < i < n$. Then the direction vectors s_i, $1 \leqq i < n$, are nonzero and mutually conjugate provided $Pg_i \neq 0$ and $h_i \neq 0$ for $1 \leqq i < n$ (regular case). Moreover, $s_i^{\mathrm{T}} c = 0$ for $1 \leqq i < n$.

Proof. We prove this theorem by induction. Suppose that $s_k \neq 0$ and $s_k^{\mathrm{T}} c = 0$ and, moreover, $s_i^{\mathrm{T}} \tilde{G} s_k = 0$, $s_i^{\mathrm{T}} h_k = 0$ and $h_i^{\mathrm{T}} h_k = 0$ for $1 \leqq i < k$ where $k \leqq n - 2$, which certainly holds for $k = 1$ provided $Pg_1 \neq 0$ and $h_1 \neq 0$.

(a) Using (4.4) we get

$$s_k^{\mathrm{T}} h_{k+1} = s_k^{\mathrm{T}} \tilde{y}_k - \frac{s_k^{\mathrm{T}} \tilde{y}_k}{s_k^{\mathrm{T}} h_k} s_k^{\mathrm{T}} h_k = 0$$

since $s_k^{\mathrm{T}} c = 0$, and

$$s_i^{\mathrm{T}} h_{k+1} = s_i^{\mathrm{T}} \tilde{y}_k - \frac{s_k^{\mathrm{T}} \tilde{y}_k}{s_k^{\mathrm{T}} h_k} s_i^{\mathrm{T}} h_k = 0$$

since $s_i^{\mathrm{T}} c = 0$, $s_i^{\mathrm{T}} h_k = 0$ and $s_i^{\mathrm{T}} \tilde{y}_k = \alpha_i s_i^{\mathrm{T}} \tilde{G} s_k = 0$ for $1 \leqq i < k$ by the assumption.

(b) Using (4.3) we get

$$h_1^{\mathrm{T}} h_{k+1} = -s_1^{\mathrm{T}} h_{k+1} = 0$$

and

$$h_i^{\mathrm{T}} h_{k+1} = -s_i^{\mathrm{T}} h_{k+1} + \frac{\tilde{y}_{i-1}^{\mathrm{T}} h_i}{\tilde{y}_{i-1}^{\mathrm{T}} s_{i-1}} s_{i-1}^{\mathrm{T}} h_{k+1} = 0$$

for $1 < i \leqq k$ since $s_i^{\mathrm{T}} h_{k+1} = 0$ for $1 \leqq i \leqq k$ by (a).

(c) We have $g_i^T h_{k+1} = 0$ for $1 \leqq i \leqq k$ by (b) since the vector Pg_i is a linear combination of the vectors h_j, $1 \leqq j \leqq i$, by (4.3) and (4.5) and since $h_{k+1}^T c = 0$ by (4.4).

(d) The condition $s_{k+1}^T c = 0$ follows from (4.3) since $h_{k+1}^T c = 0$ by (4.4) and $s_k^T c = 0$ by the assumption. Moreover, $s_{k+1} \neq 0$ since $s_k^T h_{k+1} = 0$ by (a) and $h_{k+1} \neq 0$ by the assumption so that $h_{k+1}^T s_{k+1} = -h_{k+1}^T h_{k+1} \neq 0$ by (4.3). Using (4.3) we get

$$s_k^T \tilde{G} s_{k+1} = -s_k^T \tilde{G} h_{k+1} + \frac{\tilde{y}_k^T h_{k+1}}{\tilde{y}_k^T s_k} s_k^T \tilde{G} s_k = -s_k^T \tilde{G} h_{k+1} + \frac{s_k^T \tilde{G} h_{k+1}}{\tilde{y}_k^T s_k} \tilde{y}_k^T s_k = 0$$

and

$$s_i^T \tilde{G} s_{k+1} = -s_i^T \tilde{G} h_{k+1} + \frac{\tilde{y}_k^T h_{k+1}}{\tilde{y}_k^T s_k} s_i^T \tilde{G} s_k = -\frac{1}{\alpha_i} \tilde{y}_i^T h_{k+1} =$$

$$= -\frac{1}{\alpha_i} (l_{i+1}^2 g_{i+1} - l_i^2 g_i)^T h_{k+1} - \frac{2}{\alpha_i} (l_{i+1} F_{i+1} - l_i F_i) c^T h_{k+1} = 0$$

for $1 \leqq i < k$ since $s_i^T \tilde{G} s_k = 0$ for $1 \leqq i < k$ by the assumption, $g_i^T h_{k+1} = 0$ for $1 \leqq i \leqq k$ by (c) and $h_{k+1}^T c = 0$ by (4.4). Note that $\alpha_i \neq 0$ if $h_i \neq 0$. $\square$

Theorem 4.1 shows that the vectors s_i, $1 \leqq i < n$, generated by the formula (4.3), are nonzero and mutually conjugate in the regular case. These vectors span a subspace which is orthogonal to the vector c so that

$$\sum_{i=1}^{n-1} \frac{s_i s_i^T}{s_i^T \tilde{G} s_i} = \tilde{G}^{-1} - \frac{\tilde{G}^{-1} c c^T \tilde{G}^{-1}}{c^T \tilde{G}^{-1} c}. \tag{4.6}$$

This equality can be easily verified by multiplying it by the linearly independent vectors $\tilde{G} s_i$, $1 \leqq i \leqq n - 1$, and c. Using (4.6) we can find a minimizer of both the quadratic function $\tilde{F}(x)$ and the conic function $F(x)$ subject to the linear constraint $l(x) = l_n$. It is given by the formula

$$x_{n+1} = x_n + s_n \tag{4.7}$$

where

$$s_n = -\sum_{i=1}^{n-1} \frac{s_i s_i^T}{s_i^T \tilde{G} s_i} \tilde{g}_n = -\sum_{i=1}^{n-1} \alpha_i l_{i+1}^2 \frac{s_i^T g_{i+1}}{s_i^T \tilde{y}_i} s_i. \tag{4.8}$$

The point $x_{n+1} \in X$ given by (4.7) is the same as that obtained by means of $n - 1$ perfect steps with the directions (3.5). Therefore, we can continue in the same way as in the previous section. Let

$$x_{n+2} = x_{n+1} + \alpha_{n+1} s_{n+1} \tag{4.9}$$

where

$$s_{n+1} = c - \sum_{i=1}^{n-1} \frac{\tilde{y}_i^T c}{\tilde{y}_i^T s_i} s_i \tag{4.10}$$

and where the steplength α_{n+1} is chosen by the perfect line search so that $s_{n+1}^{T}g_{n+1} = 0$. Then $s_i^T\tilde{y}_{n+1} = 0$ for $1 \leqq i < n$ by (4.10) and $s_i^T g_{n+1} = 0$ for $1 \leqq i < n$ since g_{n+1} is parallel to the vector c. Therefore

$$s_i^T\tilde{y}_{n+1} = s_i^T(l_{n+2}^2 g_{n+2} - l_{n+1}^2 g_{n+1}) + \\ + 2s_i^T c(l_{n+2}F_{n+2} - l_{n+1}F_{n+1}) = l_{n+2}^2 s_i^T g_{n+2} = 0$$

for $1 \leqq i < n$, which together with $s_{n+1}^T g_{n+1} = 0$ implies $g_{n+2} = 0$ and, consequently, x_{n+1} is a minimizer of the conic function $F(x)$.

The following algorithm summarizes the above results.

Algorithm 4.1

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

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

Step 3: If $k < 0$ then set $s := -g$, $k := 1$, $l := 1$ and go to Step 4. If $k = 0$, go to Step 6. If $k > 0$, go to Step 5.

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 $g_1 := g(x_1)$, $g_2 := g(x_2)$. Compute the values

$$\varrho_1 := \sqrt{((F_1 - F)^2 - \alpha_1^2 g^T s g_1^T s)},$$

$$\varrho_2 := \sqrt{((F_2 - F)^2 - \alpha_2^2 g^T s g_2^T s)},$$

$$\tilde{l}_1 := \frac{\alpha_1 g^T s}{F_1 - F - \varrho_1},$$

$$\tilde{l}_2 := \frac{\alpha_2 g^T s}{F_2 - F - \varrho_2},$$

and the vector

$$c := -\frac{l}{2}\frac{(\tilde{l}_2^2 g_2 - g)\alpha_1 - (\tilde{l}_1^2 g_1 - g)\alpha_2}{(\tilde{l}_2 F_2 - F)\alpha_1 - (\tilde{l}_1 F_1 - F)\alpha_2}.$$

Go to Step 9.

Step 5: If $k = 1$ then set $h := g - (c^T g / c^T c)\, c$, $u := c$, and $v := 0$ else set

$$h := y - \frac{c^T y}{c^T c} c - \frac{s^T y}{s^T h} h, \quad u := u - \frac{y^T c}{y^T s} s, \quad v := v - \alpha \tilde{l}_2^2 \frac{s^T g}{s^T y} s.$$

If either $k = n$ or $h^T h \leqq \varepsilon g^T g$ then set $k := 1 - k$.

Step 6: If $k < 0$ then set $s := -\operatorname{sgn}(g^{\mathrm{T}}v)\, v$, $k := 0$, and go to Step 8. If $k = 0$ then set $s := -\operatorname{sgn}(g^{\mathrm{T}}u)\, u$, $k := 1$ and go to Step 4. If $k > 0$ then continue.

Step 7: If $k = 1$ then set $s := -h$ else set

$$s := -h + \frac{y^{\mathrm{T}}h}{y^{\mathrm{T}}s}\, s\,.$$

Set $s := -\operatorname{sgn}(s^{\mathrm{T}}g)\, s$ and $k := k + 1$.

Step 8: Use an imperfect line search procedure to determine the point $x_2 := x + \alpha_2 s$. Compute the value $F_2 := F(x_2)$ and the gradient $g_2 := g(x_2)$. If $k > 0$ then compute the values

$$\varrho_2 := \sqrt{((F_2 - F)^2 - \alpha_2^2 g^{\mathrm{T}} s g_2^{\mathrm{T}} s)}\,,$$

$$\tilde{l}_2 := \frac{\alpha_2 g^{\mathrm{T}} s}{F_2 - F - \varrho_2}\,,$$

and the vector

$$y := (\tilde{l}_2^2 g_2 - g) + \frac{2c}{l}(\tilde{l}_2 F_2 - F)\,.$$

Step 9: Set $x := x_2$, $F := F_2$, $g := g_2$, $\alpha := \alpha_2$, $l := \tilde{l}_2 l$, and go to Step 2.

Comments. 1) The algorithm is invariant under the initial scaling. Therefore we set $l = 1$ in Step 3.

2) We use the ratios $\tilde{l}_1 = l_1/l$, $\tilde{l}_2 = l_2/l$ instead of the values l_1, l_2 in Step 4 and Step 8.

3) We use $y = \tilde{y}/l^2$ instead of $\tilde{y}$ in Step 7 of the algorithm. It only changes the absolute values of the vectors s_i, $1 \leqq i < n$, but it has no effect on their conjugacy.

4) The value of the linear function $l(x)$ remains unchanged in Step 8 of the algorithm when it is used for the conic function. Therefore we can easily set $\tilde{l}_2 = 1$. However, the value $\tilde{l}_2$ computed by means of the parameter ϱ_2 is useful for checking the suitability of the conic function as a model for the general objective function.

5) The algorithm uses a perfect line search procedure in Step 4. It can be reduced to the determination of two points $x_1 = x + \alpha_1 s$ and $x_2 = x + \alpha_2 s$ only, when the objective function is conic and when we use the interpolation formula (4.1).

6) The imperfect line search procedure which is used in Step 8 of the algorithm can be reduced to the determination of the only point $x_2 = x + \alpha_2 s$.

5. FURTHER EXTENSION

Consider the objective function of the form

$$F(x) = \varphi(\tilde{F}(x), l(x)) \tag{5.1}$$

such that $\partial\varphi(\tilde{F}(x), l(x))/\partial\tilde{F} > 0$ for all $x \in R_n$, where $\tilde{F}(x)$ is a quadratic function with a constant positive definite Hessian matrix $\tilde{G}$, and $l(x)$ is a linear function with a constant gradient c. Using the same notation as in Section 2, we can write

$$\begin{aligned} F &= \varphi(\tilde{F}, l)\,, \\ g &= \sigma\tilde{g} + \tau c\,, \end{aligned} \tag{5.2}$$

where $\sigma = \partial\varphi/\partial\tilde{F}$ and $\tau = \partial\varphi/\partial l$. Comparing (5.2) with (2.2), we can see that both Algorithm 3.1 and Algorithm 4.1 can be easily generalized for the objective function (5.1) provided it is possible to compute $\sigma(x)$ and $\tau(x)$ at an arbitrary point $x \in R_n$. The following lemma allows us to determine the vector c from the values F, F_1, and F_2 and the gradients g, g_1, and g_2 computed at three different points x, x_1, and x_2 lying on a line.

Lemma 5.1. *Let $x \in R_n$, $x_1 = x + \alpha_1 s \in R_n$, and $x_2 = x + \alpha_2 s \in R_n$ be three different points. Then*

$$c = \frac{\left(\dfrac{g_2}{\sigma_2} - \dfrac{g}{\sigma}\right)\alpha_1 - \left(\dfrac{g_1}{\sigma_1} - \dfrac{g}{\sigma}\right)\alpha_2}{\left(\dfrac{\tau_2}{\sigma_2} - \dfrac{\tau}{\sigma}\right)\alpha_1 - \left(\dfrac{\tau_1}{\sigma_1} - \dfrac{\tau}{\sigma}\right)\alpha_2}\,. \tag{5.3}$$

Proof. See proof of Lemma 2.2. □

The most complicated problem associated with the function (5.1) is the determination of the values σ and τ. We confine ourselves to the objective function of the form

$$F(x) = \tilde{F}(x)\, l^k(x)\,, \tag{5.4}$$

defined in the open halfspace $X = \{x \in R_n, l(x) > 0\}$, which is a generalization of the conic function (2.1). Using (5.4) we get the formulae

$$\begin{aligned} F &= \tilde{F}\; l^k\,, \\ g &= \tilde{g}\; l^k + \frac{k}{l} Fc\,, \end{aligned} \tag{5.5}$$

so that $\sigma = l^k$ and $\tau = kF/l$, and (5.3) implies

$$c = \frac{\left(\dfrac{g_2}{l_2^k} - \dfrac{g}{l^k}\right)\alpha_1 - \left(\dfrac{g_1}{l_1^k} - \dfrac{g}{l^k}\right)\alpha_2}{\left(\dfrac{kF_2}{l_2^{k+1}} - \dfrac{kF}{l^{k+1}}\right)\alpha_1 - \left(\dfrac{kF_1}{l_1^{k+1}} - \dfrac{kF}{l^{k+1}}\right)\alpha_2}\,. \tag{5.6}$$

The following lemma offers the possibility of determining the ratio l_2/l from the values F and F_2 and the gradients g and g_2 computed at two different points x and x_2.

Lemma 5.2. *Let $x \in X$ and $x_2 = x + \alpha_2 s \in X$ be two different points. Then the ratio l_2/l is a solution of the equation*

(5.7)
$$kF\left(\frac{l_2}{l}\right)^{k+2} - ((2+k)F + \alpha_2 g^T s)\left(\frac{l_2}{l}\right)^{k+1} + ((2+k)F_2 - \alpha_2 g_2^T s)\left(\frac{l_2}{l}\right) - kF_2 = 0\,.$$

Proof. Using (5.5) we get

$$\tilde{F} = l^{-k}F\,, \quad \tilde{g} = l^{-k}g - kFl^{-(k+1)}c\,.$$

Since the quadratic function has to satisfy the equality

$$\tilde{F}_2 - \tilde{F} = \frac{\alpha_2}{2}(\tilde{g}_2^T s + \tilde{g}^T s)$$

and since $\alpha_2 c^T s = l_2 - l$, we get after substitution

$$l_2^{-k}F_2 - l^{-k}F = \frac{1}{2}\left(\alpha_2 l_2^{-k} g_2^T s + \alpha_2 l^{-k} g^T s - kF_2 l_2^{-k}\frac{l_2 - l}{l_2} - kFl^{-k}\frac{l_2 - l}{l}\right),$$

which gives (5.7) after rearrangements. □

Using the above investigation we can see that both Algorithm 3.1 and Algorithm 4.1 find a minimum of the function (5.4) after a finite number of steps if we replace (2.3) and (2.4) by (5.7) and (5.6), respectively, and set

(5.8)
$$\tilde{y} = \tilde{g}_2 - \tilde{g} = \frac{g_2}{l_2^k} - \frac{g}{l^k} - k\left(\frac{F_2}{l_2^{k+1}} - \frac{F}{l^{k+1}}\right)c\,.$$

Note that the equation (5.5) has a real solution $l_2/l \geqq 0$ if $FF_2 > 0$, which is usually satisfied for x_2 sufficiently close to x. The following table contains some special models.

Table 1

k	model	degree of (5.7)
-3	$F(x) = \tilde{F}(x)/l^3(x)$	3
-2	$F(x) = \tilde{F}(x)/l^2(x)$	2
-1	$F(x) = \tilde{F}(x)/l(x)$	1
1	$F(x) = \tilde{F}(x)\, l(x)$	3
2	$F(x) = \tilde{F}(x)\, l^2(x)$	4

Note that k in (5.4) can be any real number.

References

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

[2] *E. M. L. Beale:* A derivation of conjugate gradients. In: Nonlinear Optimization (Lootsma, F.A., ed.) New York: Academic Press 1972.

[3] *P. Bjørstad, J. Nocedal:* Analysis of a new algorithm for one-dimensional minimization. Computing 22, 93—100 (1979).
[4] *W. R. Boland, E. R. Kamgnia, J. S. Kowalik:* A conjugate-gradient optimization method invariant to nonlinear scaling. J. Optimization Theory Appl. 27, 221—230 (1979).
[5] *W. C. Davidon:* Conic approximations and collinear scalings for optimizers. SIAM J. Numer. Anal. 17, 268—281 (1980).
[6] *L. C. W. Dixon:* Conjugate gradient algorithms: Quadratic termination properties without line searches. J. Inst. Math. Appl. 15, 9—18 (1975).
[7] *R. Fletcher, C. M. Reeves:* Function minimization by conjugate gradients. Comput. J. 7, 149—154 (1964).
[8] *I. Fried:* N-step conjugate gradient minimization scheme for nonquadratic functions, AIAA J. 9, 2286—2287 (1971).
[9] *M. R. Hestenes, E. Stiefel:* The method of conjugate gradients for solving linear systems. J. Res. Nat. Bur. Standards, Section B 49, 409—436 (1952).
[10] *J. S. Kowalik, E. R. Kamgnia, W. R. Boland:* An exponential function as a model for a conjugate gradient optimization method. J. Math. Anal. Appl. 67, 476—482 (1979).
[11] *M. J. D. Powell:* Restart procedures for the conjugate gradient method. Math. Programming 12, 241—254 (1977).
[12] *J. E. Shirey:* Minimization of extended quadratic functions. Numer. Math. 39, 157—161 (1982).
[13] *F. Sloboda:* An imperfect conjugate gradient algorithm. Aplikace matematiky 27, 426—434 (1982).
[14] *F. Sloboda:* A generalized conjugate gradient algorithm for minimization. Numer. Math. 35, 223—230 (1980).
[15] *D. C. Sorensen:* The Q-superlinear convergence of a collinear scaling algorithm for unconstrained optimization. SIAM J. Numer. Anal. 17, 84—114 (1980).

Souhrn

ALGORITMY SDRUŽENÝCH GRADIENTŮ PRO KONICKÉ FUNKCE

LADISLAV LUKŠAN

Příspěvek obsahuje popis a analýzu dvou modifikací metody sdružených gradientů pro nepodmíněnou minimalizaci, které naleznou minimum konické funkce po konečném počtu kroků. Navíc je prezentováno další zobecnění metody sdružených gradientů založené na obecnější třídě modelových funkcí.

Резюме

АЛГОРИФМЫ СОПРЯЖЕННЫХ ГРАДИЕНТОВ ДЛЯ КОНИЧЕСКИХ ФУНКЦИЙ

LADISLAV LUKŠAN

Статья содержит описание и анализ двух модификаций метода сопряженных градиентов для минимизации без ограничений, которые находят минимум конической функции после конечного числа шагов. Кроме того указано дальнейшее обобщение метода сопряжённых градиентов, основанное на более общем класе модельных функций.

Author's address: Ing. *Ladislav Lukšan,* CSc., Středisko výpočetní techniky ČSAV, Pod vodárenskou věží 2, 182 07 Praha 8.