JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 69, No. 1, APRIL 1991

Efficient Generalized Conjugate Gradient Algorithms, Part 1: Theory

Y. LIU[1] AND C. STOREY[2]

Communicated by L. C. W. Dixon

Abstract. The effect of inexact line search on conjugacy is studied in unconstrained optimization. A generalized conjugate gradient method based on this effect is proposed and shown to have global convergence for a twice continuously differentiable function with a bounded level set.

Key Words. Unconstrained optimization, hybrid and restart conjugate gradient methods, inexact line search.

1. Introduction

Conjugate gradient methods are some of the most useful algorithms for unconstrained optimization of large problems by virtue of their storage saving properties. The general routine is given below.

Suppose that f is a twice continuously differentiable function on its domain containing a bounded level set L,

$$L=\{x\in R^n: f(x)\le f(x_1)\}, \qquad n\ge 2, \tag{1}$$

where x_1 is an initial point. Then, the minimum point of f is to be found by a sequence of line searches on directions s_k, $k=1,2,\dots$,

$$x_{k+1}=x_k+a_k s_k, \tag{2}$$

where

$$a_k=\arg\min_a f(x_k+as_k), \tag{3}$$

$$s_{k+1}=-g_{k+1}+b_k s_k. \tag{4}$$

[1] Visiting Scholar, Department of Mathematical Sciences, Loughborough University of Technology, Loughborough, Leicestershire, England.

[2] Professor, Department of Mathematical Sciences, Loughborough University of Technology, Loughborough, Leicestershire, England.

0022-3239/91/0400-0129$06.50/0

In (4), $g_k = f'(x_k)$ is the gradient of f at x_k, $s_1 = -g_1$ is the steepest descent direction, and b_k is determined by the conjugacy condition

$$s_k^T H_{k+1} s_{k+1} = 0, \qquad k = 1, 2, \dots, \tag{5}$$

where $H_{k+1} = f''(x_{k+1})$ is the Hessian of f at x_{k+1} and s_k^T is the transpose of s_k. Thus, we could have b_k of the form suggested by Daniel (Ref. 1),

$$b_k = g_{k+1}^T H_{k+1} s_k / s_k^T H_{k+1} s_k. \tag{6}$$

But in practice, in order to avoid computation of second derivatives and storage of matrices, condition (5) is changed to its difference form,

$$s_{k+1}^T (g_{k+1} - g_k) = 0, \qquad k = 1, 2, \dots. \tag{7}$$

Accordingly, we have b_k of the form suggested by Sorenson (Ref. 2),

$$b_k = g_{k+1}^T (g_{k+1} - g_k) / s_k^T (g_{k+1} - g_k), \qquad k = 1, 2, \dots. \tag{8}$$

On the other hand, by (3) we have

$$s_k^T g_{k+1} = 0, \qquad k = 1, 2, \dots; \tag{9}$$

then, (8) becomes

$$b_k = -g_{k+1}^T (g_{k+1} - g_k) / s_k^T g_k, \qquad k = 1, 2, \dots. \tag{10}$$

We note that s_{k+1} in (4) is independent of the length of s_k when b_k takes the form of (6), (8), or (10). This property is useful in computations. On the other hand, if b_k takes the Polak-Ribière form (Ref. 3)

$$b_k = g_{k+1}^T (g_{k+1} - g_k) / g_k^T g_k, \qquad k = 1, 2, \dots, \tag{11}$$

which is obtained from (10) considering (4) and (9), this property is lost.

We know that, if f is a quadratic function, then

$$g_{k+1}^T g_k = 0, \qquad k = 1, 2, \dots,$$

and (11) takes the form suggested by Fletcher-Reeves (Ref. 4),

$$b_k = g_{k+1}^T g_{k+1} / g_k^T g_k, \qquad k = 1, 2, \dots. \tag{12}$$

Fletcher-Reeves' method is the simplest of all conjugate gradient methods and its convergence is proved by Powell (Ref. 5), where f is not restricted to being quadratic. Furthermore, Al-Baali (Ref. 6) extends this result to show the convergence of Fletcher-Reeves' method with inexact line search when a_k satisfies the conditions

$$|g_{k+1}^T s_k| < c_1 |g_k^T s_k|, \qquad 0 < c_1 < 1/2, \tag{13}$$

and

$$f(x_{k+1}) \leq f(x_k) + c_2 a_k g_k^T s_k, \qquad 0 < c_2 < 1/2. \tag{14}$$

The finite termination property of conjugate gradient methods does not hold in general for nonquadratic f (and even for quadratic f because of round-off errors), and so restart procedures are necessary. Fletcher and Reeves suggest restarting after every $n+1$ iterations (Ref. 4), but when n is very large, this has little effect. Shanno (Ref. 7) gives an angle test to determine when conjugate gradient algorithms should be restarted with a steepest-descent direction. Touati-Ahmed and Storey (Ref. 8) give three efficient hybrid conjugate gradient methods in which (11), (12), or a restart is used according to certain switching criteria.

2. Generalized Conjugate Gradient Method

In this paper, we develop a new kind of method which we call generalized conjugate gradient method.

In practice, exact line search is generally impossible and, in any case, would be very uneconomical. When line search is inexact, conjugate directions are not the best directions as we will see later. For this reason, we modify b_k by considering the effect of inexact linear search.

First, we write the Newton approximation of $f(x_{k+1})$,

$$F(x_{k+1}) = F(x_k + a_k s_k) = f(x_k) + a_k g_k^T s_k + a_k^2 s_k^T H_k s_k / 2, \qquad k \geq 2.$$

Assume that $s_k^T H_k s_k > 0$; then,

$$\min_{a_k}(F(x_k + a_k s_k) - f(x_k)) \leq -(g_k^T s_k)^2/(2 s_k^T H_k s_k) = -V_k/2. \qquad (15)$$

Now, we find the direction s_k on the $g_k - s_{k-1}$ plane,

$$s_k = -t g_k + u s_{k-1}, \qquad (16)$$

such that V_k takes its maximum. Clearly, we can write

$$\begin{aligned} V_k &= (t g_k^T g_k - u g_k^T s_{k-1})^2/(t^2 g_k^T H_k g_k - 2tu g_k^T H_k s_{k-1} + u^2 s_{k-1}^T H_k s_{k-1}) \\ &= [(g_k^T g_k)^2 s_{k-1}^T H_k s_{k-1} - 2 g_k^T s_{k-1} g_k^T g_k g_k^T H_k s_{k-1} \\ &\qquad + (g_k^T s_{k-1})^2 g_k^T H_k g_k]/[s_{k-1}^T H_k s_{k-1} g_k^T H_k g_k - (g_k^T H_k s_{k-1})^2] \\ &\quad - [t(g_k^T g_k g_k^T H_k s_{k-1} - g_k^T s_{k-1} g_k^T H_k g_k) \\ &\qquad - u(g_k^T g_k s_{k-1}^T H_k s_{k-1} - g_k^T s_{k-1} g_k^T H_k s_{k-1})]^2 \\ &\quad /[(t^2 g_k^T H_k g_k - 2tu g_k^T H_k s_{k-1} \\ &\qquad + u^2 s_{k-1}^T H_k s_{k-1})(s_{k-1}^T H_k s_{k-1} g_k^T H_k g_k - (g_k^T H_k s_{k-1})^2)]. \end{aligned}$$

Hence, if

$$s_{k-1}^T H_k s_{k-1} > 0, \qquad k \geq 2, \tag{17}$$

$$D = s_{k-1}^T H_k s_{k-1} g_k^T H_k g_k - (g_k^T H_k s_{k-1})^2 > 0, \qquad k \geq 2, \tag{18}$$

then

$$t^2 g_k^T H_k g_k - 2tu g_k^T H_k s_{k-1} + u^2 s_{k-1}^T H_k s_{k-1} = s_k^T H_k s_k > 0, \tag{19}$$

if $t \neq 0$ or $u \neq 0$. Thus, V_k takes its maximum on a nonnull direction s_k,

$$\begin{aligned} s_k = [(&g_k^T s_{k-1} g_k^T H_k s_{k-1} - g_k^T g_k s_{k-1}^T H_k s_{k-1}) g_k \\ &+ (g_k^T g_k g_k^T H_k s_{k-1} - g_k^T s_{k-1} g_k^T H_k g_k) s_{k-1}]/D, \end{aligned} \tag{20}$$

and the descent property holds,

$$\begin{aligned} g_k^T s_k = -[(&g_k^T g_k)^2 s_{k-1}^T H_k s_{k-1} - 2 g_k^T g_k g_k^T s_{k-1} g_k^T H_k s_{k-1} \\ &+ (g_k^T s_{k-1})^2 g_k^T H_k g_k]/D < 0, \end{aligned}$$

when $g_k^T g_k \neq 0$, since (19) is a positive-definite quadratic form.

We note that (17) is satisfied if f is strictly convex on the line $x = x_{k-1} + a s_{k-1}$ at the point x_k and that (18) is satisfied if H_k is positive definite and g_k and s_{k-1} are linearly independent.

Before we give the new algorithms, we first establish the following general convergence theorem.

Theorem 2.1. Let the line search direction be

$$s_k = [(u_k g_k^T s_{k-1} - t_k g_k^T g_k) g_k + (u_k g_k^T g_k - v_k g_k^T s_{k-1}) s_{k-1}]/w_k, \tag{21}$$

where t_k, u_k, v_k, w_k satisfy

$$t_k > 0, \qquad v_k > 0, \tag{22}$$

$$1 - u_k^2/(t_k v_k) \geq 1/(4 r_k), \qquad \infty > r_k > 0, \tag{23}$$

$$(v_k / g_k^T g_k)/(t_k / s_{k-1}^T s_{k-1}) \leq r_k, \qquad \infty > r_k > 0, \tag{24}$$

$$w_k = t_k v_k - u_k^2, \tag{25}$$

or

$$t_k = 1, \qquad u_k = v_k = 0, \qquad w_k = g_k^T g_k \neq 0, \qquad r_k = 0. \tag{26}$$

Then, the following descent property holds:

$$g_k^T s_k < 0.$$

If in addition there exists $p_k > 0$ such that

$$f(x_{k+1}) - f(x_k) \leq -p_k (g_k^T s_k)^2 / s_k^T s_k, \tag{27}$$

then if

$$\sum_{k=1}^{\infty} p_k/(1+r_k^2)=\infty, \tag{28}$$

it follows that

$$\lim_{k\to\infty} \inf g_k^T g_k = 0;$$

and if

$$\lim_{k\to\infty} \inf p_k/(1+r_k^2) > 0, \tag{29}$$

then we have the convergence property

$$\lim_{k\to\infty} g_k^T g_k = 0.$$

Proof. First, by (22), (23), and (24), we can see that

$$w_k > 0, \qquad g_k^T g_k > 0.$$

Therefore,

$$g_k^T s_k = -[(g_k^T g_k)^2 t_k - 2 g_k^T g_k g_k^T s_{k-1} u_k + (g_k^T s_{k-1})^2 v_k]/w_k < 0,$$

or by (26),

$$g_k^T s_k = -g_k^T g_k < 0;$$

thus, the descent property is proved. Secondly, since

$$\begin{aligned}(g_k^T s_k)^2 = [&(g_k^T g_k)^2 (g_k^T g_k t_k - g_k^T s_{k-1} u_k)^2 \\ &- 2 g_k^T g_k g_k^T s_{k-1} (g_k^T g_k t_k - g_k^T s_{k-1} u_k)(g_k^T g_k u_k - g_k^T s_{k-1} v_k) \\ &+ (g_k^T g_k u_k - g_k^T s_{k-1} v_k)^2 (g_k^T s_{k-1})^2]/w_k^2,\end{aligned} \tag{30}$$

and

$$\begin{aligned}s_k^T s_k = [&(g_k^T g_k t_k - g_k^T s_{k-1} u_k)^2 g_k^T g_k \\ &- 2(g_k^T g_k t_k - g_k^T s_{k-1} u_k)(g_k^T g_k u_k - g_k^T s_{k-1} v_k) g_k^T s_{k-1} \\ &+ (g_k^T g_k u_k - g_k^T s_{k-1} v_k)^2 s_{k-1}^T s_{k-1}]/w_k^2,\end{aligned} \tag{31}$$

then comparing (30) and (31) and letting

$$q_k = g_k^T s_{k-1}/g_k^T g_k,$$

we have

$$\begin{aligned}&(g_k^T s_k)^2/s_k^T s_k \\ &= g_k^T g_k/[1+(u_k - v_k q_k)^2 (s_{k-1}^T s_{k-1}/g_k^T g_k - q_k^2)/(t_k - 2u_k q_k + v_k q_k^2)^2];\end{aligned} \tag{32}$$

and since

$$s_{k-1}^T s_{k-1}/g_k^T g_k - q_k^2 \geq 0$$

and

$$\begin{aligned}
&(u_k - v_k q_k)^2/(t_k - 2u_k q_k + v_k q_k^2)^2 \\
&= v_k^2/4(t_k v_k - u_k^2) - (2u_k^2 - t_k v_k - 2u_k v_k q_k + v_k^2 q_k^2)^2 \\
&/[4(t_k v_k - u_k^2)(t_k - 2u_k q_k + v_k q_k^2)^2] \\
&\leq v_k^2/4(t_k v_k - u_k^2),
\end{aligned} \tag{33}$$

then, by (30), (31), (22), (23), and (24), we have

$$\begin{aligned}
(g_k^T s_k)^2/s_k^T s_k &\geq g_k^T g_k/[1 + v_k^2 s_{k-1}^T s_{k-1}/4g_k^T g_k(t_k v_k - u_k^2)] \\
&\geq g_k^T g_k/(1 + r_k^2),
\end{aligned} \tag{34}$$

or by (26),

$$(g_k^T s_k)^2/s_k^T s_k = g_k^T g_k. \tag{35}$$

Thus, by (27) and (34) or (35),

$$f(x_{k+1}) - f(x_k) \leq -g_k^T g_k p_k/(1 + r_k^2).$$

By the fact that f is bounded on the bounded level set L, the series

$$\sum_{k=1}^{\infty} g_k^T g_k p_k/(1 + r_k^2)$$

is convergent. Therefore, by (28), we have

$$\liminf_{k \to \infty} g_k^T g_k = 0,$$

and by (29), we have

$$\lim_{k \to \infty} g_k^T g_k = 0.$$

This completes the proof of the theorem. □

3. Algorithms

Suppose that, in the line search,

$$f(x_{k+1}) - f(x_k) \leq -p(g_k^T s_k)^2/s_k^T s_k, \qquad p > 0. \tag{36}$$

Then, the algorithms are as follows.

Algorithm A1.

Step 1. Set $k=1$, $s_1=-g_1$.

Step 2. Line Search. Compute $x_{k+1}=x_k+a_k s_k$; set $k=k+1$.

Step 3. If $g_k^T g_k<\epsilon$, then stop; otherwise, go to Step 4.

Step 4. If $k>n>2$, go to Step 8; otherwise, go to Step 5.

Step 5. Let $t_k=s_{k-1}^T H_k s_{k-1}$, $v_k=g_k^T H_k g_k$, and $u_k=g_k^T H_k s_{k-1}$.

Step 6. If $t_k>0$, $v_k>0$, $1-u_k^2/(t_k v_k)\geq 1/(4r)$, and

$$(v_k/g_k^T g_k)/(t_k/s_{k-1}^T s_{k-1})\leq r, \qquad r>0,$$

then go to Step 7; otherwise, go to Step 8.

Step 7. Let

$$s_k=[(u_k g_k^T s_{k-1}-t_k g_k^T g_k)g_k+(u_k g_k^T g_k-v_k g_k^T s_{k-1})s_{k-1}]/w_k,$$

where $w_k=t_k v_k-u_k^2$; go to Step 2.

Step 8. Set x_k to x_1; go to Step 1.

Algorithm A2. Algorithm A2 is only different from Algorithm A1 in Step 5 in order to avoid the computation of H_k and the storage of matrices. Step 5 is changed as follows.

Step 5. Let

$$t_k=s_{k-1}^T[g_k-f'(x_k-\delta s_{k-1})]/\delta,$$

$$u_k=g_k^T[g_k-f'(x_k-\delta s_{k-1})]/\delta,$$

$$v_k=g_k^T[g_k-f'(x_k-\gamma g_k)]/\gamma,$$

where δ and γ are suitably small positive numbers.

4. Discussion on Algorithms

First, we see that the descent property and convergence of the two algorithms hold by Theorem 2.1. Secondly, the hypothesis (36) is satisfied if the line search is exact. When the line search is inexact, (36) holds when (13) with $0<c_1<1$ and (14) hold; for the proof, see Ref. 5 and Ref. 8.

Next, suppose that the Hessian H_k of f is positive definite with condition number N_k. Then, in Algorithm A1,

$$(v_k/g_k^T g_k)/(t_k/s_{k-1}^T s_{k-1})\leq N_k,$$

by the definition of N_k, and

$$1-u_k^2/(t_k v_k) \geq [1-(g_k^T s_{k-1})^2/(s_{k-1}^T s_{k-1} g_k^T g_k)]/N_k.$$

If we take $r \geq N_k$ and the angle between g_k and $s_{k-1} > 30°$, then the conditions in Step 6 are satisfied. Thus, when f is a positive-definite quadratic function with the condition number N of the Hessian H less than r, Algorithms A1 and A2 are equivalent to the usual conjugate gradient method with n-step restart when the line search is exact.

Finally, we discuss the parameters used in the algorithms. Let η denote the accuracy of the computer. That is, η is the smallest positive number such that, when $1.0+\eta$ is computed, the result is greater than 1.0. Then, we suggest taking

$$\epsilon=\sqrt{\eta}, \qquad r=1/\sqrt{\eta}, \qquad \delta=\sqrt{\eta}/\sqrt{s_{k-1}^T s_{k-1}}, \qquad \gamma=\sqrt{\eta}/\sqrt{g_k^T g_k}.$$

A preliminary numerical comparison of Algorithm A2 with the Fletcher-Reeves method and the BFGS method indicated its relative efficiency. A more detailed examination of a number of different implementations of Algorithm A2 and comparison with other modified conjugate gradient methods will be reported in Part 2 of this paper (Ref. 9).

Remark 4.1. On setting

$$\bar{s}_{k-1}=(I-g_k g_k^T/g_k^T g_k)s_{k-1}, \tag{37}$$

(20) simplifies to

$$s_k=[(-\bar{s}_{k-1}^T H_k \bar{s}_{k-1})g_k+(\bar{s}_{k-1}^T H_k g_k)\bar{s}_{k-1}]g_k^T g_k/D, \tag{38}$$

with

$$D=(\bar{s}_{k-1}^T H_k \bar{s}_{k-1})(g_k^T H_k g_k)-(g_k^T H_k \bar{s}_{k-1})^2, \tag{39}$$

and the numerator in (38) does not involve the term $g_k^T H_k g_k$. Consequently, taking

$$s_k=-g_k+[(\bar{s}_{k-1}^T H_k g_k)/(\bar{s}_{k-1}^T H_k \bar{s}_{k-1})]\bar{s}_{k-1} \tag{40}$$

saves a gradient evaluation when finite differences are used. Clearly, s_k is conjugate to $\bar{s}_{k-1}$, rather than to s_{k-1}. If we now let

$$H_k s_{k-1} \approx [g_k-g(x_k-\delta \bar{s}_{k-1})]/\delta=(g_k-\bar{g}_k)/\delta, \tag{41}$$

with δ a suitably small number, then since $g_k^T \bar{s}_{k-1}=0$, (40) becomes

$$s_k=-g_k-[(g_k^T(g_k-\bar{g}_k)/\bar{g}_k^T \bar{s}_{k-1})]\bar{s}_{k-1}, \tag{42}$$

which is a generalization of the Polak-Ribière formula. A disadvantage is that the s_k given by (40) is not now scaled to give the minimizer of the Newton approximation, with a steplength of 1, as is the s_k given by (20). A new algorithm, based on (40), has been formulated and preliminary numerical tests are very encouraging. We hope to report on the details later.

References

1. DANIEL, J. W., *The Conjugate Gradient Method for Linear and Nonlinear Operator Equations*, SIAM Journal on Numerical Analysis, Vol. 4, pp. 10–26, 1967.
2. SORENSON, H. W., *Comparison of Some Conjugate Direction Procedures for Function Minimization*, Journal of the Franklin Institute, Vol. 288, pp. 421–441, 1969.
3. POLAK, E., and RIBIÈRE, G., *Note sur la Convergence des Méthodes de Directions Conjuguées*, Revue Française d'Informatique et de Recherche Opérationelle, Vol. 16, pp. 35–43, 1969.
4. FLETCHER, R., and REEVES, C. M., *Function Minimization by Conjugate Gradients*, Computer Journal, Vol. 7, pp. 149–154, 1964.
5. POWELL, M. J. D., *Nonconvex Minimization Calculations and the Conjugate Gradient Method*, Report No. DAMTP 1983/NA14, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, England, 1983.
6. AL-BAALI, M., *Descent Property and Global Convergence of the Fletcher–Reeves Method with Inexact Line Search*, IMA Journal of Numerical Analysis, Vol. 5, pp. 121–124, 1985.
7. SHANNO, D. F., *Globally Convergent Conjugate Gradient Algorithms*, Mathematical Programming, Vol. 33, pp. 61–67, 1985.
8. TOUATI-AHMED, D., and STOREY, C., *Efficient Hybrid Conjugate Gradient Techniques*, Journal of Optimization Theory and Applications, Vol. 64, No. 2, pp. 379–397, 1990.
9. HU, Y. F., and STOREY, C., *Efficient Generalized Conjugate Gradient Algorithms, Part 2: Implementation*, Journal of Optimization Theory and Applications, Vol. 69, No. 1, pp. 139–152, 1991.