Mathematical Programming 15 (1978) 343–348.
North-Holland Publishing Company

SHORT COMMUNICATION

EXTENDING THE RELATIONSHIP BETWEEN THE CONJUGATE GRADIENT AND BFGS ALGORITHMS

A. BUCKLEY

Concordia University, Montreal, Quebec, Canada

Received 21 February 1978
Revised manuscript received 14 June 1978

We wish to examine the conjugate gradient and quasi-Newton minimization algorithms. A relation noted by Nazareth is extended to an algorithm in which conjugate gradient and quasi-Newton search directions occur together and which can be interpreted as a conjugate gradient algorithm with a changing metric.

Key words: Minimization, Optimization, Variable metric, Conjugate-gradient, Quasi-Newton.

Introduction

In a recent report, Nazareth [7] has pointed out a close relationship that exists between the search directions defined by the conjugate gradient (CG) minimization algorithm and the Broyden–Fletcher–Goldfarb–Shanno (BFGS) form of quasi-Newton algorithm. It is the purpose of this paper to extend this relation and to show that it is specific to the BFGS update. In a companion paper [2] the author has discussed how the resulting theorem may be used to motivate a new conjugate gradient type of minimization algorithm.

Background

We will be discussing two types of algorithm and we wish to begin by reviewing these and thus establishing the notation to be used. The first is the CG type. Here, as preferred by Powell [8], we will choose for our work the Polak–Ribière form. Thus, given a function $f(x) = f(x_1, x_2, \dots, x_n)$ with gradient $g(x)$ and given a point x_0, we set $k = 1$ and

$$d_1 = -g_0 \equiv -g(x_0) \tag{1}$$

and then iterate on the following steps:

$$x_k = x_{k-1} + \lambda_k d_k, \tag{2a}$$

$$\beta_k = \frac{g_k^{\mathrm{T}} \gamma_k}{g_{k-1}^{\mathrm{T}} g_{k-1}}, \tag{2b}$$

$$d_{k+1} = -g_k + \beta_k d_k, \tag{2c}$$

$$k \leftarrow k + 1.$$

Here $\gamma_k = g_k - g_{k-1} \equiv g(x_k) - g(x_{k-1})$ and λ_k is determined by an exact line search.

Now it is well known that this algorithm has quadratic termination, but that if d_1 is not equal to $-g_0$ the termination property is usually lost. However, a modification of the algorithm (2) can regain termination for other choices of d_1 (see Allwright [1], Hestenes and Stiefel [5], Nazareth [7] and Powell [8]). One chooses a positive definite matrix H, and defines the transformation of variables

$$y = H^{-1/2} x.$$

In particular

$$y_0 = H^{-1/2} x_0$$

and then the CG algorithm is applied in the y-coordinates. Of course, all of the standard properties of the CG algorithm then hold in the y-coordinates, so in particular finite termination still obtains. If we now take the search directions that one obtains in this way and transform them back into the x-coordinates, it is straightforward to show that the effect is equivalent to applying a modified CG algorithm directly in the x-coordinates. To do this we set $k = 1$ and

$$d_1 = -H g_0 \tag{3}$$

and then iterate on the steps:

$$x_k = x_{k-1} + \lambda_k d_k, \tag{4a}$$

$$\hat{\beta}_k = \frac{g_k^{\mathrm{T}} H \gamma_k}{g_{k-1}^{\mathrm{T}} H g_{k-1}}, \tag{4b}$$

$$d_{k+1} = -H g_k + \hat{\beta}_k d_k, \tag{4c}$$

$$k \leftarrow k + 1.$$

In order to be practical, H must be chosen so that products of the form Hv (v a vector) are easily computed, (i.e. in $O(n)$ operations) but that is not the subject of this paper. It is discussed though in the paper [2]. Notice that this algorithm does start with a direction different from the steepest descent direction $-g_0$.

The second type of algorithm we wish to discuss is the quasi-Newton (QN) or variable metric one. For the QN update formula we will introduce the expression

$$U(x_k, H, \alpha) \equiv \left(I - \frac{\delta \gamma^{\mathrm{T}}}{\delta^{\mathrm{T}} \gamma}\right) H \left(I - \frac{\gamma \delta^{\mathrm{T}}}{\delta^{\mathrm{T}} \gamma}\right) + \frac{\delta \delta^{\mathrm{T}}}{\delta^{\mathrm{T}} \gamma} - \alpha w w^{\mathrm{T}} \tag{5}$$

where

$$\delta \equiv \delta_k = x_k - x_{k-1},$$

$$\gamma \equiv \gamma_k = g_k - g_{k-1},$$

$$w \equiv w_k = \frac{H\gamma}{\gamma^T H\gamma} - \frac{\delta}{\delta^T\gamma}. \quad (6)$$

Then, given x_0 and a positive definite matrix H_0, we start with $k = 1$ and set

$$d_1 = -H_0 g_0 \quad (7)$$

and write the quasi-Newton iteration as

$$x_k = x_{k-1} + \lambda_k d_k, \quad (8a)$$

$$H_k = U(x_k, H_{k-1}, \alpha_k), \quad (8b)$$

$$d_{k+1} = -H_k g_k, \quad (8c)$$

$$k \leftarrow k + 1.$$

The general form of update (5) may be found in many places (for example [3]). We will, as is customary, refer to the special case where $\alpha = 0$ as the BFGS update.

Throughout this paper we assume that line searches are exact. For the quadratic case this is of course easy to achieve.

A combined algorithm

We now wish to define an algorithm which is a combination of the CG and QN algorithms (4) and (8). The primary object is to show that, using the BFGS update, the sequence of points one obtains is the same as in the normal CG algorithm (2). Hence the property of termination in at most n steps is preserved.

We can think of the proposed algorithm in the following way. If we wish to apply the modified CG algorithm (4), it may be desirable to periodically change the metric, that is, to change H. In this case an obvious way to modify H is via the QN update formula (5). Then the quasi-Newton search direction (8c) corresponds to a restart of the CG algorithm (4) in a new metric $H = H_k$. In the paper [2] we describe how to dynamically change H in response to the observed behaviour of a general function, but here we wish to examine the effect of changes in the metric when f is quadratic. In fact our claim is that the sequence of points x_k ($k = 0, 1, 2, \ldots$) remains the same provided that the metric is revised by the BFGS formula.

So, let numbers $0 = Q_0 < Q_1 < Q_2 \ldots$ be specified and let the modified CG algorithm (4) be applied with a fixed metric from Q_i to Q_{i+1}. Furthermore, at each point Q_i let the metric be updated by the formula (5). To be more specific, we assume that an initial positive definite matrix H_0 is given (we will use $H_0 = I$, just for simplicity), that d_1 is then chosen by (7) which, with $H_0 = I$, is just (1) and that the CG algorithm (2) is applied until x_{Q_1} is reached. Let $t = Q_1$. Then we update H_0 to H_1 using (5),

$$H_1 = U(x_t, H_0, 0),$$

amd determine d_{t+1} by

$$d_{t+1} = -H_1 g_t.$$

This choice of d_{t+1} may be considered as a QN step as in (8c)—except that the index on H does not happen to be t—or it may equally well be considered as a restarting step of the CG algorithm as in (3)—except that $H = H_1$ and the index on x does not happen to be 0. So the point x_t is taken as a CG restart point and from x_{t+1} to x_{Q_2} we apply the modified CG algorithm (4) with $H = H_1$.

It is clear that this restart procedure may be repeated at x_{Q_2} and indeed at any point x_{Q_i}. Thus at each point x_{Q_i} we define a matrix H_i using a QN update and we restart from x_{Q_i} with a QN search direction.

In order to prove the claim that changes to the metric done in this way have no effect on the sequence of points obtained when f is quadratic, we define two sequences of points. We obtain $x_0, x_1, x_2, \ldots$ by the normal CG algorithm (2). (We may ignore Fletcher's device [4] of setting $d_k = -g_k$ for $k = n+1, 2n+1, \ldots$ because here we are interested in f quadratic, and termination will occur at or before x_n.) We also set $\bar{x}_0 = x_0$ and let $\bar{x}_0, \bar{x}_1, \bar{x}_2, \ldots$ be determined as described above by changing the metric at $\bar{x}_{Q_1}, \bar{x}_{Q_2}, \ldots$. To simplify the notation we let H denote the matrix defined when a step is taken from the point x_{k-1} and we let H^* denote the matrix defined at x_k. Thus H^* is just H if $k \neq Q_i$ for any i so that no update is done at x_k, or H^* is obtained from H by (5) with $\alpha = 0$. We let d_{k+1} denote the normal CG direction from x_k and we use $\bar{d}_{k+1}$ for the direction obtained with the changing metric algorithm. Then we obtain

Theorem 1. *Suppose $x_i = \bar{x}_i$ for $i = 0, 1, \ldots, k$, assume*

$$g_{k-1}^{\mathrm{T}} H g_{k-1} = g_{k-1}^{\mathrm{T}} g_{k-1} \tag{9}$$

and

$$H g_j = g_j \quad \text{for } j \geq k, \tag{10}$$

where $g_j = g(x_j)$ and $x_{k+1}, x_{k+2}, \ldots$ are the points which would be reached from x_k ($= \bar{x}_k$) by the normal CG algorithm (2). Then

$$\bar{d}_{k+1} = d_{k+1},$$

$$g_k^{\mathrm{T}} H^* g_k = g_k^{\mathrm{T}} g_k, \tag{11}$$

$$H^* g_j = g_j \quad \text{for } j > k. \tag{12}$$

Proof. We will use some well-known properties of the CG algorithm with exact line searches. Such occasions are noted by†. Now, if no update is done at $\bar{x}_k$, then $H^* = H$. Thus (11) and (12) are immediate from (10). From (4c),

$$\bar{d}_{k+1} = -Hg_k + \hat{\beta}_k d_k$$

with

$$\hat{\beta}_k = \frac{g_k^T H \gamma_k}{g_{k-1}^T H g_{k-1}}.$$

But by (9) and (10), $\hat{\beta}_k$ reduces to β_k defined by (2b); then (10) applied again reduces $\bar{d}_{k+1}$ to d_{k+1} defined by (2c).

On the other hand, if we update H to H^* at x_k, we have

$$\bar{d}_{k+1} = -H^* g_k \tag{13}$$

with

$$H^* = \left(I - \frac{\delta\gamma^T}{\delta^T\gamma}\right) H \left(I - \frac{\gamma\delta^T}{\delta^T\gamma}\right) + \frac{\delta\delta^T}{\delta^T\gamma}. \tag{14}$$

Now $\delta = \lambda_k d_k$, $\gamma = g_k - g_{k-1}$ and $\delta^T g_k = 0$† so from (10) and (14)

$$H^* g_k = Hg_k - \frac{\gamma^T H g_k}{\delta^T \gamma}\delta \tag{15}$$

$$= g_k - \frac{\gamma^T g_k}{d_k^T \gamma} d_k. \tag{16}$$

But the sequences are identical up to x_k, so it is easily shown that (16) reduces $\bar{d}_{k+1}$ in (13) to (2c)†. Now, continuing with (15), we have, using (10) and because $\delta^T g_k = 0$†,

$$g_k^T H^* g_k = g_k^T H g_k - \frac{\gamma^T H g_k}{\delta^T\gamma}\delta^T g_k$$

$$= g_k^T g_k$$

and (11) has been shown. Finally, for $j > k$, $\delta^T g_j = 0$† and $g_{k-1}^T g_j = g_k^T g_j = 0$† so (14) and (10) give

$$H^* g_j = Hg_j - \frac{\gamma^T H g_j}{\delta^T\gamma}\delta$$

$$= g_j.$$

Applying this theorem inductively then shows that the algorithms generate identical points from start to finish.

The case $Q_i = i$

We simply wish to note the fact that Theorem 1 still applies when an update is done on every step. Thus as a special case we obtain Myers' result [6] that the normal CG algorithm is identical to the normal BFGS quasi-Newton algorithm when f is quadratic and line searches are exact.

The case $\alpha \neq 0$

Perhaps the most interesting feature of this paper is that Theorem 1 is specific to the BFGS algorithm, i.e. to the case $\alpha = 0$. It is simple to construct an example (almost any one will do) to see that the theorem does not hold when $\alpha \neq 0$. For example, with the DFP update ($\alpha = \gamma^T H \gamma$),

$$f(x) = \tfrac{1}{2} x^T \begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & 1 \\ 0 & 1 & 4 \end{pmatrix} x,$$

$x_0^T = (1, 1, -1)$ and $Q_1 = 2$, one finds that $x_i = \bar{x}_i$ for $i = 0, 1$ and 2, but that

$$d_3^T = \tfrac{3}{7}(-1 \quad -\tfrac{4}{7} \quad \tfrac{1}{7}), \qquad \bar{d}_3^T = \tfrac{3}{65}(7 \quad -4 \quad 0).$$

In fact, we see here that using the DFP update prevents termination at x_3, so that this example also demonstrates that the changing metric CG algorithm does not retain the property of n-step finite termination when α is chosen to be non-zero.

To see why it is so easy to construct this example, one only has to examine step (16) of the proof of Theorem 1. One finds that $H^* g_k$ contains components along directions other than g_k and d_k when α is non-zero.

References

[1] J.C. Allwright, "Improving the conditioning of optimal control problems using simple models", in: D.J. Bell, ed., *Recent mathematical developments in control* (Academic Press, London, 1972).

[2] A.G. Buckley, "A combined conjugate-gradient quasi-Newton minimization algorithm", *Mathematical Programming*, to appear.

[3] R. Fletcher, "A new approach to variable metric algorithms", *Computer Journal* 13 (1970) 317–322.

[4] R. Fletcher, "Conjugate direction methods", in: W. Murray, ed., *Numerical methods for unconstrained optimization*, (Academic Press, London, 1972) pp. 73–86.

[5] M.R. Hestenes and E.L. Stiefel, "Methods of conjugate gradients for solving linear systems", *Journal of Research of the National Bureau of Standards* 49 (1952) 409–436.

[6] G.E. Myers, "Properties of the conjugate gradient and Davidon methods", *Journal of Optimization Theory and Applications* 2 (1968) 209–219.

[7] L. Nazareth, "A relationship between the BFGS and conjugate gradient algorithms", AMD Tech. Memo 282, Argonne National Laboratory (1977).

[8] M.J.D. Powell, "Restart procedures for the conjugate gradient method", *Mathematical Programming* 12 (1977) 241–254.