IMA Journal of Numerical Analysis (1991) **11**, 325–332

A Modified BFGS Algorithm for Unconstrained Optimization

YA-XIANG YUAN

Computing Centre, Academia Sinica, Beijing 100080, China

[Received 25 April 1989 and in revised form 16 October 1990]

In this paper we present a modified BFGS algorithm for unconstrained optimization. The BFGS algorithm updates an approximate Hessian which satisfies the most recent quasi-Newton equation. The quasi-Newton condition can be interpreted as the interpolation condition that the gradient value of the local quadratic model matches that of the objective function at the previous iterate. Our modified algorithm requires that the function value is matched, instead of the gradient value, at the previous iterate. The modified algorithm preserves the global and local superlinear convergence properties of the BFGS algorithm. Numerical results are presented, which suggest that a slight improvement has been achieved.

1. Introduction

VARIABLE metric algorithms for unconstrained optimization are a class of numerical algorithms for solving the following problem:

$$\min_{x\in\mathbb{R}^n} f(x). \tag{1.1}$$

They are iterative. On the kth iteration an approximation point x_k and an $n \times n$ matrix B_k are available. A search direction

$$d_k = -B_k^{-1}\,\nabla f(x_k) \tag{1.2}$$

is calculated, then a step-length $\alpha_k > 0$ is calculated to satisfy certain line search conditions, and the next iterate x_{k+1} is set to be $x_k + \alpha_k d_k$. One of the important features of the method is the choice of matrices B_k. Variable metric algorithms require B_k positive definite and satisfying the quasi-Newton equation

$$B_{k+1}\delta_k = \gamma_k, \tag{1.3}$$

where

$$\delta_k = \alpha_k d_k, \tag{1.4}$$

$$\gamma_k = \nabla f(x_k + \delta_k) - \nabla f(x_k). \tag{1.5}$$

The search direction d_k in (1.2) is the solution of the following quadratic subproblem:

$$\min_{d\in\mathbb{R}^n} \phi_k(d) = f(x_k) + d^{\mathrm{T}}\,\nabla f(x_k) + \tfrac{1}{2}d^{\mathrm{T}}B_k d, \tag{1.6}$$

which is an approximation to problem (1.1) near the current iterate x_k, since $\phi_k(d) \simeq f(x_k + d)$ for small d. In fact, the definition of $\phi_k(\cdot)$ in (1.6) implies that

$$\phi_k(0) = f(x_k), \tag{1.7}$$

$$\nabla\phi_k(0) = \nabla f(x_k), \tag{1.8}$$

and condition (1.3) is equivalent to

$$\nabla\phi_k(x_{k-1} - x_k) = \nabla f(x_{k-1}). \tag{1.9}$$

Thus, $\phi_k(x - x_k)$ is a quadratic interpolation of $f(x)$ at x_k and x_{k-1}, satisfying conditions (1.7)–(1.9). Davidon (1980) introduced 'conic models' where a nonquadratic function $\phi_k(d)$ is constructed and $\phi_k(d)$ satisfies conditions (1.7)–(1.9) and the interpolation condition

$$\phi_k(x_{k-1} - x_k) = f(x_{k-1}). \tag{1.10}$$

More details can be found in Schnabel (1983). Since we only consider the quadratic model (1.6), we require $\phi_k(d)$ to satisfy conditions (1.7)–(1.8) and (1.10). One reason for motivating us to investigate the case when $\phi_k(d)$ satisfies equation (1.10) is that we believe conditions (1.7)–(1.8) and (1.10) are very natural interpolation conditions.

In Section 2, a variant of the secant method for one-dimensional optimization is discussed, a modified BFGS algorithm for the multivariable case based on interpolation condition (1.10) is given in Section 3, and finally in Section 4 numerical results are presented and we also give a brief discussion.

2. A variant of the secant method for one-dimensional problems

In a one-dimensional optimization problem, a nonlinear function of only one variable is minimized:

$$\min_{x \in \mathbb{R}^1} f(x). \tag{2.1}$$

The secant method for (2.1) chooses the new iterate x_{k+1} as follows,

$$x_{k+1} = x_k - \frac{f'(x_k)(x_k - x_{k-1})}{f'(x_k) - f'(x_{k-1})}. \tag{2.2}$$

Let $d_k = x_{k+1} - x_k$. By comparing equations (1.2) and (2.2), it can be easily seen that the secant method has $B_{k+1} = [f'(x_k) - f'(x_{k-1})]/(x_k - x_{k-1})$, which is the unique solution of (1.3). The secant method (2.2) is Q-superlinearly convergent with the Q-order $\tau = \frac{1}{2}(\sqrt{5} + 1) \simeq 1{\cdot}618$. More exactly, if x_k converges to a point x^* at which $f'(x^*) = 0$, $f''(x^*) \neq 0$, and $f'''(x^*) \neq 0$, it can be shown that

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^\tau} = \left| \frac{f'''(x^*)}{2f''(x^*)} \right|^{1/\tau}. \tag{2.3}$$

Details can be found in Ostrowski (1966).

We can derive a variant of (2.2) if we consider the approximation

$$\phi_k(d) = f(x_k) + f'(x_k)d + \tfrac{1}{2}c_k d^2. \tag{2.4}$$

$\phi_k(\cdot)$ satisfies conditions (1.7)–(1.8). Forcing $\phi_k(d)$ to satisfy (1.10) yields

$$c_k = \frac{2[f(x_{k-1}) - f(x_k) - f'(x_k)(x_{k-1} - x_k)]}{(x_{k-1} - x_k)^2}. \tag{2.5}$$

Assuming $c_k > 0$, because $-f'(x_k)/c_k$ is the minimum of $\phi_k(\cdot)$, we let the next iterate be

$$\begin{aligned} x_{k+1} &= x_k - f'(x_k)/c_k \\ &= x_k - \frac{f'(x_k)(x_k - x_{k-1})}{2\{f'(x_k) - [f(x_k) - f(x_{k-1})]/(x_k - x_{k-1})\}} \end{aligned} \tag{2.6}$$

which is a variant of formula (2.2). It is always true that $c_k > 0$ if the objective function $f(x)$ is strictly convex. Both iteration formulae (2.2) and (2.6) can be viewed as approximations to Newton's iteration

$$x_{k+1} = x_k - f'(x_k)/f''(x_k). \tag{2.7}$$

In other words, (2.2) is a modified Newton's method where the approximation

$$\frac{f'(x_k) - f'(x_{k-1})}{x_k - x_{k-1}} \simeq f''(x_k) \tag{2.8}$$

is used, and in (2.6) one uses

$$\frac{2\{f'(x_k) - [f(x_k) - f(x_{k-1})]/(x_k - x_{k-1})\}}{x_k - x_{k-1}} \simeq f''(x_k). \tag{2.9}$$

The leading errors in (2.8) and (2.9) are $-\frac{1}{2}f'''(x_k)(x_k - x_{k-1})$ and $-\frac{1}{3}f'''(x_k)(x_k - x_{k-1})$ respectively. Hence (2.6) is a better approximation to (2.7) than (2.2). It can be shown that the iterate scheme (2.6) also has local Q-superlinear convergence properties. Assuming x_k converges to x^* at which $f'(x^*) = 0$, $f''(x^*) \neq 0$, and $f'''(x^*) \neq 0$, from (2.6) we can prove that

$$|x_{k+1} - x^*| = \left|\frac{f'''(x^*)}{3f''(x^*)}\right| |x_k - x^*|\, |x_{k-1} - x^*| + o(|x_k - x^*|\, |x_{k-1} - x^*|), \tag{2.10}$$

which implies that

$$\lim_{k\to\infty} |x_{k+1} - x^*|/|x_k - x^*|^{\tau} = \left|\frac{f'''(x^*)}{3f''(x^*)}\right|^{1/\tau}, \tag{2.11}$$

where again $\tau = \frac{1}{2}(\sqrt{5} + 1)$. The convergence rates (2.11) and (2.3) indicate that (2.6) is slightly faster than (2.2). We consider a simple example that minimizes the function $f(x) = -xe^{-x}$, which has the unique minimum $x^* = 1$. Initial points $x_1 = 0{\cdot}0$ and $x_2 = 0{\cdot}1$ are chosen, and the numerical results of both methods (2.2) and (2.6) are presented in Table 1. Only the errors $1{\cdot}0 - x_k$ $(k \leq 10)$ are given to compare the performances of both methods.

TABLE 1
The methods (2.2) *and* (2.6) *for* $f(x) = -xe^{-x}$

Error	(2.2)	(2.6)
$1.0 - x_1$	1.000000000×10^0	1.000000000×10^0
$1.0 - x_2$	0.900000000×10^0	0.900000000×10^0
$1.0 - x_3$	0.461341340×10^0	0.450000000×10^0
$1.0 - x_4$	0.244721116×10^0	0.211038490×10^0
$1.0 - x_5$	$0.832019761 \times 10^{-1}$	$0.606665134 \times 10^{-1}$
$1.0 - x_6$	$0.174604885 \times 10^{-1}$	$0.881302355 \times 10^{-2}$
$1.0 - x_7$	$0.138265830 \times 10^{-2}$	$0.373191911 \times 10^{-3}$
$1.0 - x_8$	$0.239160474 \times 10^{-4}$	$0.223267244 \times 10^{-5}$
$1.0 - x_9$	$0.330444768 \times 10^{-7}$	$0.557076149 \times 10^{-9}$
$1.0 - x_{10}$	$0.790284505 \times 10^{-12}$	$0.111022302 \times 10^{-14}$

The results in Table 1 show that in this simple example the numerical performance of (2.6) is better than that of (2.2). This motivates us to investigate updating formulae based on (1.10) for n-dimensional unconstrained optimization.

3. A modified BFGS algorithm

The BFGS algorithm for unconstrained optimization problem (1.1) uses the search direction (1.2), and the matrices B_k $(k = 1, 2, \ldots)$ are updated by the BFGS formula

$$B_{k+1} = B_k - \frac{B_k \delta_k \delta_k^{\mathrm{T}} B_k}{\delta_k^{\mathrm{T}} B_k \delta_k} + \frac{\gamma_k \gamma_k^{\mathrm{T}}}{\delta_k^{\mathrm{T}} \gamma_k}, \tag{3.1}$$

which satisfies the quasi-Newton equation (1.3). The BFGS algorithm is one of the most efficient algorithms for solving the unconstrained optimization problem (1.1). More details can be found in Fletcher (1980). This algorithm is globally convergent when the objective function $f(x)$ is convex, if the inexact line search conditions

$$f(x_k + \alpha_k d_k) \leq f(x_k) + c_1 \alpha_k d_k^{\mathrm{T}} \nabla f(x_k), \tag{3.2}$$

$$d_k^{\mathrm{T}} \nabla f(x_k + \alpha_k d_k) \geq c_2 d_k^{\mathrm{T}} \nabla f(x_k) \tag{3.3}$$

are satisfied, where c_1 and c_2 are two constants such that $0 < c_1 \leq c_2 < 1$ and $c_1 < 0.5$. Assume the sequence x_k $(k = 1, 2, \ldots)$ generated by the BFGS algorithm converges to a solution x^* where $\nabla^2 f(x^*)$ is positive definite. If the stepsize $\alpha_k = 1$ is chosen whenever it satisfies the line search conditions (3.2)–(3.3), it can be shown that x_k converges to x^* Q-superlinearly, that is

$$\|x_{k+1} - x^*\| / \|x_k - x^*\| \to 0. \tag{3.4}$$

More details can be found in Powell (1976) and Dennis & Moré (1977).

We consider the case when the search direction d_k is a solution of the quadratic subproblem (1.6), but this subproblem satisfies the interpolation condition (1.10), instead of (1.9). (We recall from Section 1 that (1.9) is equivalent to the quasi-Newton equation (1.3).) Using definition (1.6), the interpolation condition (1.10) with k increased by one is

$$\delta_k^{\mathrm{T}} B_{k+1} \delta_k = 2[f(x_k) - f(x_{k+1}) + \delta_k^{\mathrm{T}} \nabla f(x_{k+1})]. \tag{3.5}$$

The positive definiteness of B_{k+1} requires

$$f(x_k) - f(x_{k+1}) > -\delta_k^{\mathrm{T}} \nabla f(x_{k+1}). \tag{3.6}$$

Inequality (3.6) is trivial if the objective function $f(x)$ is strictly convex, and it is also true if the step-length α_k is chosen by an exact line search which requires $\delta_k^{\mathrm{T}} \nabla f(x_{k+1}) = 0$. We update B_{k+1} by the formula

$$B_{k+1} = B_k - \frac{B_k \delta_k \delta_k^{\mathrm{T}} B_k}{\delta_k^{\mathrm{T}} B_k \delta_k} + t_k \frac{\gamma_k \gamma_k^{\mathrm{T}}}{\delta_k^{\mathrm{T}} \gamma_k}, \tag{3.7}$$

which is a slight modification of the BFGS formula. Now relation (3.5) implies

$$t_k = \frac{2}{\delta_k^{\mathrm{T}} \gamma_k} [f(x_k) - f(x_{k+1}) + \delta_k^{\mathrm{T}} \nabla f(x_{k+1})]. \tag{3.8}$$

It is easily seen that $t_k \in [0, 2]$ if $f(x)$ is convex. Furthermore, $t_k = 1$ if $f(x)$ is quadratic on the line segment between x_k and x_{k+1}. Since B_{k+1} is positive definite if and only if $t_k > 0$, we require $t_k > 0$ which is equivalent to condition (3.6). For a general nonlinear objective function $f(x)$, one can modify the line search conditions so that (3.6) is satisfied. Assuming the line search conditions (3.2)–(3.3) are used, we restrict t_k to the interval

$$0{\cdot}01 \leq t_k \leq 100. \tag{3.9}$$

Now we can state a modified BFGS algorithm with inexact line searches as follows.

ALGORITHM 3.1. (A modified BFGS algorithm)
Step 0: Given $x_1 \in \mathbb{R}^n$, B_1 positive definite, $k := 1$.
Step 1: Calculate $\nabla f(x_k)$,

$$d_k = -B_k^{-1} \nabla f(x_k).$$

Step 2: Calculate $\alpha_k > 0$ such that conditions (3.2)–(3.3) are satisfied,

$$x_{k+1} = x_k + \alpha_k d_k.$$

Step 3: Calculate t_k by (3.8),

if $t_k < 0{\cdot}01$ then set $t_k = 0{\cdot}01$,

if $t_k > 100$ then set $t_k = 100$,

update B_{k+1} by (3.7), set $k := k + 1$, and go to Step 1.

Because t_k is truncated so that inequality (3.9) is satisfied, by slightly modifying the proof in Powell (1976), it can be shown that Algorithm 3.1 converges globally for convex objective functions with inexact line searches. Assume x_k converges to a strict local minimum x^* where $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite, and that $f(x)$ is twice continuously differentiable. Then it can be proved that

$$\lim_{k\to\infty} t_k = 1. \tag{3.10}$$

Thus it is reasonable to hope that local Q-superlinear convergence of the BFGS algorithm can be extended to the modified algorithm where updating formula (3.7) is used. Details of local analyses of the BFGS algorithm can be found in Dennis & Moré (1977).

4. Numerical results and discussions

A FORTRAN subroutine was programmed to test the modified BFGS algorithm presented in the previous section. The following test problems are used.

Problem 1. (Rosenbrook's function)

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \tag{4.1}$$

starting point: $(-1{\cdot}2, 1{\cdot}0)^{\mathrm{T}}$,
solution: $(1, 1)^{\mathrm{T}}$.

Problem 2. (Powell's function of four variables)

$$f(x_1, x_2, x_3, x_4) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4, \tag{4.2}$$

starting point: $(3, -1, 0, 1)^{\mathrm{T}}$,
solution: $(0, 0, 0, 0)^{\mathrm{T}}$.

Problem 3. (Wood's function)

$$f(x_1, x_2, x_3, x_4) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10{\cdot}1[(x_2 - 1)^2 + (x_4 - 1)^2] - 19{\cdot}8(x_2 - 1)(x_4 - 1), \tag{4.3}$$

starting point: $(-3, -1, -3, -1)^{\mathrm{T}}$,
solution: $(1, 1, 1, 1)^{\mathrm{T}}$.

Problem 4. (A quartic function)

$$f(x_1, x_2, x_3, x_4) = \sum_{i=1}^{4} (10^{i-1}x_i^4 + x_i^3 + 10^{1-i}x_i^2), \tag{4.4}$$

starting point: $(1, 1, 1, 1)^{\mathrm{T}}$,
solution: $(0, 0, 0, 0)^{\mathrm{T}}$.

Problem 5. (*A sine-valley function*)

$$f(x_1, x_2) = 100[x_2 - \sin(x_1)]^2 + 0{\cdot}25x_1^2, \tag{4.5}$$

starting point: $(\tfrac{3}{2}\pi, -1)^{\mathrm{T}}$,
solution: $(0, 0)^{\mathrm{T}}$.

Problems 1–3 are well-known test problems for unconstrained optimization (for example, see Fletcher & Freeman, 1977; Fletcher, 1980). Problem 4 is so constructed that the Hessian of the objective function at the starting point is very different from that at the solution, namely

$$\nabla^2 f(1, 1, 1, 1) = \text{DIAG}\,[20, 126{\cdot}2, 1206{\cdot}02, 12006{\cdot}002], \tag{4.6}$$

but

$$\nabla^2 f(0, 0, 0, 0) = \text{DIAG}\,[2, 0{\cdot}2, 0{\cdot}02, 0{\cdot}002]. \tag{4.7}$$

Problem 5 is a sine-valley function having a valley along the curve $x_2 = \sin(x_1)$.

The calculations were carried out on an IBM 4341 machine with double-precision arithmetic. The convergence criterion is

$$\|\nabla f(x_k)\| \leq \varepsilon. \tag{4.8}$$

For each problem we run for both $\varepsilon = 10^{-8}$ and $\varepsilon = 10^{-12}$ and the initial matrix B_1 is chosen to be the unit matrix I. The step-length $\alpha_k > 0$ satisfying conditions (3.2)–(3.3), with $c_1 = 0{\cdot}01$ and $c_2 = 0{\cdot}9$, is calculated by quadratic approximation with bracketing techniques. More details can be found in Fletcher (1980). All problems are solved successfully, and the numbers of iterations and function evaluations are given in Table 2. We also solved these problems by the BFGS algorithm, and the numerical results of the BFGS algorithm are also given in Table 2.

In Table 2, NI and NF are the numbers of iterations and function evaluations respectively. The numerical results show that the modified algorithm is slightly better than the original BFGS algorithm on this collection of test problems.

One possible disadvantage of the modified algorithm is that the calculation of t_k in formula (3.8) may lose accuracy due to rounding errors. That is, when x_k converges to a solution x^* superlinearly, it follows that

$$f(x_k) - f(x_{k+1}) = O(\|x_k - x_{k+1}\|^2). \tag{4.9}$$

But, if the Hessian at the solution $\nabla^2 f(x^*)$ is positive definite, the gradient difference γ_k used in the BFGS algorithm is of the order of $\|x_k - x_{k+1}\|$ in magnitude since it can be verified that

$$\nabla f(x_k) - \nabla f(x_{k+1}) \sim \|x_k - x_{k+1}\|. \tag{4.10}$$

TABLE 2
Comparison of the modified algorithm with BFGS

Problem number	ε	BFGS algorithm NI/NF	Modified BFGS NI/NF
1	10^{-8}	33/45	34/45
1	10^{-12}	34/46	35/46
2	10^{-8}	59/65	45/51
2	10^{-12}	79/85	68/74
3	10^{-8}	57/71	54/66
3	10^{-12}	59/73	55/67
4	10^{-8}	59/65	55/61
4	10^{-12}	63/69	57/63
5	10^{-8}	40/57	39/54
5	10^{-12}	41/58	40/55

The BFGS update formula (3.1) only requires $\nabla f(x_{k+1}) - \nabla f(x_k)$ and $x_{k+1} - x_k$, but the modified formula (3.7) computes the number t_k which requires the evaluation of $f(x_k) - f(x_{k+1})$. Hence, when $\|x_{k+1} - x_k\|$ is very small, it can be seen from (4.9) and (4.10) that the modified algorithm may have numerical underflow more easily than the BFGS algorithm.

There are other updating formulae that satisfy condition (1.10). The main reason that we take the simple form (3.7) is that it is easy to see that the updating formula (3.7) possesses the global convergence property of the BFGS algorithm if t_k is bounded above and bounded below from zero. Another way of constructing other quadratic subproblems is to require the function $\phi_k(d)$ to satisfy the weighted least square condition

$$\min \mu_k \| \nabla \phi_k(x_{k-1} - x_k) - \nabla f(x_{k-1}) \|_2^2 + \nu_k \| \phi_k(x_{k-1} - x_k) - f(x_{k-1}) \|_2^2, \quad (4.11)$$

where μ_k and ν_k are nonnegative weight parameters. Though normally a solution of (4.11) does not satisfy the quasi-Newton equation, it can be viewed as a quadratic approximation function trying to satisfy interpolation conditions (1.7)–(1.10).

Acknowledgement

I am very grateful to the associate editor and two referees for their valuable comments and suggestions on the early version of the paper. The author is supported by a grant from the Academia Sinica.

References

Davidon, W. C. 1980 Conic approximations and collinear scalings for optimizers. *SIAM J. Numer. Anal.* **17**, 268–281.

Dennis, J. E., & Moré, J. J. 1977 Quasi-Newton methods, motivation and theory. *SIAM Review* **19**, 46–89.

Fletcher, R. 1980 *Practical Methods of Optimization: Vol 1, Unconstrained Optimization.* New York: John Wiley and Sons.

Fletcher, R., & Freeman, T. L. 1977 A modified Newton method for minimization. *J. Optimization Theory and Applications* **23**, 357–372.

Ostrowski, A. M. 1966 *Solution of Equations and Systems of Equations*, 2nd edn. New York: Academic Press.

Powell, M. J. D. 1976 Some global convergence properties of a variable metric algorithm for minimization without exact line searches. In: *Nonlinear Programming SIAM-AMS Proceeding Vol 9* (R. W. Cottle & C. E. Lemke, Eds). SIAM Publications. Pp. 53–72.

Schnabel, R. B. 1983 Conic methods for unconstrained optimization and tensor methods for nonlinear equations. In: *Mathematical Programming, The State of the Art* (A. Bachem, M. Grotschel, & B. Korte, Eds). Berlin: Springer-Verlag. Pp. 417–438.