International Journal of Applied Mathematics

Volume 34 No. 3 2021, 583-594
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
doi: http://dx.doi.org/10.12732/ijam.v34i3.12

NEW INVARIANT TO NONLINEAR SCALING QUASI-NEWTON ALGORITHMS

Issam A.R. Moghrabi

Gulf University for Science and Technology
College of Business Administration
Mishref – 1150, KUWAIT

Abstract: New Quasi-Newton methods for unconstrained optimization are proposed which are invariant to a nonlinear scaling of a strictly convex quadratic function. In specific, we examine a logarithmic scaling of some quadratic function and proceed to derive the necessary parameters for obtaining invariancy to such nonlinear scalings. The techniques considered in this work have the same convergence properties as the classical BFGS-method, when applied to a quadratic function.

AMS Subject Classification: 65K10
Key Words: unconstrained optimization; quasi-Newton methods; nonlinear scaling; invariancy

1. Introduction

The type of problems of interest in this work is of the form

$$\text{minimize } f(x), \text{ where } f : R^n \to R.$$

One well-known way to find a solution is by making use of a class of methods widely known as the quasi-Newton methods for unconstrained optimization. Those methods require only the objective function and its first partial derivatives (the gradient) to be derived and encoded. The Hessian matrix is assumed to be unavailable due to its usually complicated derivation and encoding, a process susceptible to error. However, an approximating matrix to the Hessian is rather utilized and updated on a step-wise basis to ensure the incorporation of the latest changes in both function value and the corresponding gradient.

Received: January 13, 2021

Given B_i, the most recent approximation to the Hessian, a new approximating matrix B_{i+1} needs to be built, corresponding to the newly computed iterate x_{i+1}. One way to determine a relationship that the new matrix B_{i+1} should satisfy is by using Taylor's series of first order approximation to the gradient about the newly computed point x_{i+1}. This leads to obtaining what is widely known as the Secant equation, [4],

$$B_{i+1}s_i = y_i\,,$$

where

$$s_i = x_{i+1} - x_i\,,$$

and

$$y_i = g_{i+1} - g_i\,.$$

The Secant Equation defines the basis for the derivation of possible updating formulae to find the new Hessian approximation B_{i+1} using simply the old approximation B_i and the corresponding iteration and their respective gradient vectors differences, namely the vectors, s_i and y_i. Generally, such an update has the form $B_{i+1} = B_i + C_i$, where C_i is some correction matrix to the Hessian approximation. Alternatively, in many practical situations, it may be preferable to instead utilize $H_{i+1} = H_i + D_i$, where D_i is some correction matrix to the inverse Hessian approximation and $H_{i+1} = B_{i+1}^{-1}$.

One particularly successful rank-two formula is the widely recognized BFGS formula. This formula is given by (see [3], [4], [5])

$$B_{i+1}^{BFGS} = B_i + \frac{y_i\ y_i^T}{y_i^T\ s_i} - \frac{B_i\ s_i\ s_i^T\ B_i}{s_i^T\ B_i s_i}\,,$$

$$H_{i+1}^{BFGS} = H_i + \left[1 + \frac{y_i^T\ H_i\ \ y_i}{s_i^T\ y_i}\right]\frac{s_i\ s_i^T}{s_i^T\ y_i} - \frac{s_i\ \ y_i^T\ H_i\ \ + H_i\ y_i\ s_i^T}{s_i^T\ y_i}\,.$$

The BFGS is shown in [7], [8] and [9] to be a least change update from B_i to B_{i+1} under the Frobenius norm. Numerical evidence support the superiority of this formula to other updating formulae especially. This superiority is especially visible in the case of non-exact line search (see, for example, [4], [5]), the thing which makes it an adopted standard.

We examine here the general problem considered in [2] and [22]. It is stated as $f = F(q(x))$, $df/dx > 0$ for $x = x_{\min}$, where $x_{\min}$ is the minimizer of $q(x)$ with respect to x and $q(x)$ is a quadratic function. F is some nonlinear scaling of $q(x)$ and for invariancy to nonlinear scaling of the objective funciton, Spedicato [22] suggested an extra parameter to be included in the quantity y_i as follows

$$y_i = g_{i+1} - \delta_i g_i, \text{where } \delta_i = \rho_{i+1}/\rho_i \text{ and } \rho_j = \frac{d}{dq}F(q)|_{q=q(x_j)}. \quad (1)$$

Alternatively, y_i may be defined as

$$y_i = g_{i+1}/\rho_{i+1} - g_i/\rho_i.$$

Invariancy to nonlinear scaling is obtained if the following properties hold true:

i. $\rho_j > 0$, for $x \neq x_{\min}$;
ii. If $x_{\min}$ is the minimum of $q(x)$, then it is also a minimum of f.

Similar non-linear scalings of the objective function have been investigated by various authors (see [18]). For example:

a) $f = \alpha\ q$, for $\alpha \in R$;
b) $f = -e^{-q}$;
and
c) $f = q^r$, for $r \in R$.

For option (a), the Huang-Oren class ([13], [23]) is obtained naturally. In the second case we have simply $\frac{df}{dq} = f$. For option (c), $\frac{df}{dq}$ can be obtained using the method of Fried [10] and of Spedicato [22]. Spedicato noticed that the parameter $\delta_i = \frac{\rho_{i+1}}{\rho_i}$ is determined using the Chain rule as follows (where A is the Hessian of $q(x)$)

$$G_{i+1} = \frac{d^2 f}{dx^2}\ |_{x=x_{i+1}} = 2A\rho_{i+1} + \frac{d^2 F}{dq^2}\ |_{x=x_{i+1}}\ g_{i+1}g_{i+1}^{T}. \quad (2)$$

Then on the premise of exact line search ($s_i^T g_{i+1} = 0$) and upon premultiplying and postmultiplying (2) by the fact that $s_i = \alpha_i p_i$, where p_i is a search vector and α_i is a step length along that search direction obtained by a designated line search algorithm ([11], [12]), we get

$$s_i^T G_{i+1} s_i = 2s_i^T A s_i \frac{dF}{dq}\ |_{x=x_{i+1}}\ .$$

Then, for a quadratic we have

$$\frac{g_{i+1}}{\rho_{i+1}} - \frac{g}{\rho} = 2As$$

and, therefore, $\delta_i = \frac{-s_i^T G_{i+1} s_i}{s_i^T g_i}$.

2. Derivation of δ_i with inaccurate line searches – A logaritmic mode

We consider here the model $F = F(q(x))$, for F being a non-linear scaling of q. Spedicato [22] examines the following form for q

$$q = c + b^T x + x^T A x,$$

for a positive definite Hessian A and where b is some constant vector and c is some scalar constant.

In this work, we will examine the following two quadratic models

$$q = e_i^T A e_i + c \text{ for } e_i = x - x_{\min}, \qquad \text{(model A)}$$

and

$$q = x_i^T A x_i + c, \qquad \text{(model B)}$$

where *model B* is a special case of *model A* for which $x_{\min} = 0$.

We will examine one particular scaling of q given as

$$f = ln(q).$$

To proceed with the derivation of δ_i, we start by considering the quantities:

$$g_{i+1} = \rho_{i+1} \nabla q_{x_{i+1}} \text{ or } \nabla q_{i+1} = \frac{g_{i+1}}{\rho_{i+1}}$$

and

$$g_i = \rho_i \nabla q_{x_i} \text{ or } \nabla q_i = \frac{g_i}{\rho_i}.$$

From the above, it follows that

$$\frac{\rho_{i+1} g_i}{\rho_i g_{i+1}} = \frac{\nabla q_i}{\nabla q_{i+1}}$$

or, equivalently,

$$\delta_i \frac{g_i}{g_{i+1}} = \frac{\nabla q_i}{\nabla q_{i+1}}. \qquad (3)$$

Using (4), we have (for (model A))

$$\delta_i \frac{s_i^T g_i}{s_i^T g_{i+1}} = \frac{s_i^T \nabla q_i}{s_i^T \nabla q_{i+1}},$$

which gives

$$\delta_i = \frac{\left(s_i^T g_i\right)\left(s_i^T \nabla q_{i+1}\right)}{\left(s_i^T g_{i+1}\right)\left(s_i^T \nabla q_i\right)} = \frac{\left(s_i^T A e_{i+1}\right)\left(s_i^T g_i\right)}{\left(s_i^T A e_i\right)\left(s_i^T g_{i+1}\right)}, \tag{4}$$

for $s_i^T g_{i+1} \neq 0$ (which is usually true for inexact line searches).

Similarly for (model B), we obtain (using (4)) the following expression for δ_i:

$$\delta_i = \frac{\left(s_i^T A x_{i+1}\right)\left(g_i^T s_i\right)}{\left(s_i^T A x_i\right)\left(s_i^T g_{i+1}\right)} = \frac{\left(y_i^T x_{i+1}\right)\left(g_i^T s_i\right)}{\left(y_i^T x_i\right)\left(s_i^T g_{i+1}\right)}, \tag{5}$$

using $2As_i = y_i$.

In order to complete the derivation of δ_i for *model A* (*model B)*, we need to derive an expression for $s_i^T A e_i$ and we start by premultiplying δ_i by $\dfrac{g_{i+1}}{g_i}$

$$\frac{g_{i+1}\ \rho_{i+1}}{g_i\ \rho_i} = \frac{A e_{i+1}}{A e_i} = \frac{e_{i+1}}{e_i}.$$

We, thus, get:

$$\delta_i = \frac{\rho_{i+1}}{\rho_i} = \frac{g_{i+1}^T\ e_{i+1}}{g_i^T\ e_i}.$$

But

$$g_i^T e_{i+1} = g_i^T\left(x_i - x_{\min}\right) + \alpha_i g_i^T p_i$$

and

$$g_{i+1}^T e_i = g_{i+1}^T\left(x_i - x_{\min}\right) = g_{i+1}^T e_{i+1} - \alpha_i\ g_{i+1}^T p_i.$$

Therefore,

$$\begin{aligned}
\delta_i &= \frac{g_{i+1}^T e_{i+1} - s_i^T g_{i+1}}{g_{i+1}^T e_i + s_i^T g_i} \\
&= \frac{2\rho_{i+1}\ e_{i+1}^T A e_{i+1} - s_i^T\ g_{i+1}}{2\rho_i\ e_i^T A e_i + s_i^T\ g_i} \\
&= \frac{2\rho_{i+1}\ q_{i+1} - s_i^T\ g_{i+1}}{2\rho_i\ q_i + s_i^T\ g_i} \\
&= \frac{2\rho_{i+1}\ q_{i+1} - s_i^T\left(2\rho_{i+1}\ A e_{i+1}\right)}{2\rho_i\ q_i + s_i^T\left(2\rho_i\ A e_i\right)}.
\end{aligned}$$

Hence,

$$\delta_i = \frac{\rho_{i+1}}{\rho_i}\left(\frac{q_{i+1} - s_i^T A\ e_{i+1}}{q_i + s_i^T A\ e_i}\right)$$

or, equivalently,

$$q_{i+1} - q_i = e_i^T A s_i + e_{i+1}^T A s_i,$$

which is true for any quadratic function. But

$$s_i^T A e_{i+1} = s_i^T A e_i + s_i^T A s_i \Rightarrow q_{i+1} - q_i = 2 e_i^T A s_i + s_i^T A s_i.$$

This gives

$$e_i^T A s_i = \frac{1}{2}\left(q_{i+1} - q_i - s_i^T A s_i\right).$$

Using $\nabla q_i = 2Ae_i$, we obtain

$$\nabla q_{i+1} - \nabla q_i = 2Ae_{i+1} - 2Ae_i,$$

which yields $As_i = \dfrac{y_i}{2}$.

Therefore,

$$e_i^T A s_i = \frac{1}{2}\left(q_{i+1} - q_i - \frac{s_i^T\ y_i^T}{2}\right). \tag{6}$$

The quantities q_i and q_{i+1} can be determined depending on the nature of the function $f = F(q)$. For example, if F corresponds to the natural logarithm, say (i.e., $f = ln(q)$), then $q = e^f$, provided f is available. Plugging the quantity in (6) into (5), δ_i is obtained. If f is not known or it is not possible to determine q using f, another derivation can be used instead (see the next section).

3. Another derivation for δ_i

We now give another derivation for δ_i in the case of $f = ln(q)$ as follows

$$\delta_i = \frac{g_{i+1}^T}{g_i^T}\frac{(x_{i+1} - x_{\min})}{(x_i - x_{\min})} = \frac{2\rho_{i+1}\ q_{i+1} - s_i^T g_{i+1}}{2\rho_i\ q_i + s_i^T g_i}. \tag{7}$$

But $\rho = \dfrac{df}{dq} = \dfrac{1}{q}$, which, upon substitution, in (7) yields

$$\delta_i = \frac{2 - s_i^T g_{i+1}}{2 + s_i^T g_i}. \tag{8}$$

For all δ_i it must be ensured that $\delta_i > 0$ in the actual implementation of the method. If this is not the case, we resort into using $\delta_i = |\delta_i|$. As indicated earlier, our expressions for δ_i are exact for a quadratic and do not suppose that the line searches are accurate.

We come here to determine δ_i in an expression which does not involve q for a general F. We use

$$\delta_i = \frac{e_{i+1}^T g_{i+1}}{e_i^T\ g_i},$$

or

$$\delta_i = \frac{e_{i+1}^T g_{i+1} + s_i^T\ g_{i+1}}{e_i^T\ g_i}, \tag{9}$$

using the definition of e_{i+1} in (model A).

Now from (5), we have

$$\delta_i = \frac{\left(e_{i+1}^T g_{i+1} + s_i^T g_{i+1}\right)\left(s_i^T g_i\right)}{\left(e_i^T g_i\right)\left(s_i^T g_{i+1}\right)}, \tag{10}$$

using $2As_i = y_i$ and $y_i^T\ e_{i+1} = e_i^T y_i\ + s_i^T y_i$ and, again, the definition of e_{i+1} in (6).

From (10), it follows that

$$\delta_i = \frac{e_i^T y_i + s_i^T y_i}{\sigma e_i^T y_i}, \quad \text{where} \quad \sigma = \frac{s_i^T g_{i+1}}{s_i^T g_i}.$$

This gives

$$\delta_i = \frac{1+\pi}{\sigma}, \quad \text{for} \quad \pi = \frac{s_i^T y_i}{e_i^T y_i}. \tag{11}$$

Now let us define the quantities

$$\tau = e_i^T\ y_i,$$

$$\lambda = e_i^T\ g_i,$$

$$\sigma_1 = s_i^T g_{i+1}$$

and

$$\beta = s_i^T y_i,$$

so that from (10) and (11), we obtain

$$\frac{\tau + \lambda + \sigma_1}{\lambda} = \frac{\tau + \beta}{\tau\ \sigma},$$

from which the following quadratic is achieved

$$\phi(\tau) = \sigma\tau^2 + (\lambda\sigma + \sigma\sigma_1 - \lambda)\,\tau - \beta\lambda. \tag{12}$$

We are interested in a positive δ_i and, thus, our aim is to find $\bar{\tau}$ as the stationary point of $\phi(\tau)$ so that we can determine λ from $\phi\left(\bar{\tau}\right)$ for that point using (12). We thus obtain

$$\bar{\tau} = \frac{-(\lambda\sigma + \sigma\sigma_1 - \lambda)}{2\sigma}. \tag{13}$$

The quantity $\bar{\tau}$ is a maximum point of $\phi(\tau)$ if and only if $\sigma < 0$ which, in turn, is only true if $s_i^T g_{i+1} > 0$. We are now able to determine λ in (11) using (12) and (13).

Since we are only interested in a positive δ_i, we need to determine the conditions under which this is true.

Corollary 1. *If H_i and H_{i+1} are positive definite, then $\bar{\tau}$ given by*

$$\bar{\tau} = \frac{-(|\lambda|\sigma + \sigma\sigma_1 - |\lambda|)}{2\sigma},$$

for $\sigma > 0$, and the following condition holds true

$$|\lambda|\sigma + \sigma\sigma_1 < |\lambda|, \tag{14}$$

then $\bar{\tau} > 0$ and, therefore, $\delta_i > 0$.

The proof is trivial and is, therefore, omitted.

4. Numerical results and conclusions

The comparative tests involve eighteen well-known test functions (see Table 1) obtained from [14], [15], [16], [17]. The comparative performances of the algorithms are assessed by taking into account both the total number of iterations and the number of function evaluations, in addition to computational timings. We define 'iteration' to mean the step carrying a point x_i along the direction $d_i = -H_i g_i$ to a new point x_{i+1}, and the number of function calls quoted is that required to reduce the value of $f(x)$ below 1.0^{-10}. The cubic interpolation technique, fully described in [1], [6] and [7], is used as the linear search routine to obtain the minimum along the search direction d_i.

Four algorithms were tested and compared, namely, (i) the standard BFGS algorithm, (ii) the method with the value for δ_i in (4), (iii) the algorithm defined by the formula in (5) for δ_i and (iv) the one given by (11).The four algorithms

were compared using twenty six test functions, each tested with several dimensions n ($2 \leq n \leq 100000$), whenever applicable. The total number of problems solved is 886, tested using each of the algorithms (see Table 2). Conditioning was applied to the update matrices involved in the computation of the search directions only initially on the first iteration (see [20] and [21]).

The performance of all the derived algorithms is comparable with some behaving better on certain problems and worse on others. The new algorithms save about 9% on function/gradient evaluations and 8% on the number of iterations, with negligible additional computational overheads. In order to guarantee the convergence of the BFGS and the other tested algorithms, the step size α_i in

$$x_{i+1} = x_i + \alpha_i d_i$$

is deemed acceptable provided it satisfies the Wolfe conditions (see [4], [7], [16], [18], [19], and [24])

$$f\left(x_i + \alpha_i d_i\right) - \ f\left(x_i\right) \leq \rho_1 \alpha_i d_i^T g_i, \tag{15}$$

$$g(x_i + \alpha_i d_i)^T d_i \geq \rho_2 d_i^T g_i, \tag{16}$$

where $0 < \rho_1 \leq \rho_2 < 1$.

Our numerical results show that algorithm with δ_i given by (6) beats the other three in the number of function/gradient calls, although the three algorithms perform about the same when compared by the total number of iterations. Its timing is also better than the remaining three tested methods (including the standard BFGS).

Finally, the computational results presented here show that nonlinear scaling methods generally improve the computational efficiency of the BFGS method on on large problems and the relative improvement increases monotonically with dimensionality n. The effect of changing to inexact line searches is marginal. In conclusion, for this particular set of test functions and for the chosen line search criteria, the new algorithms perform better than the well-known standard BFGS method.

The methods derived in this paper lends themselves to several applications for which the problem under consideration is nonlinearly scaled. Also, the algorithms need to be examined and tested within the context preconditioned Conjugate Gradient (CG) methods (see [3], [25]) to determine their effectiveness, as opposed to the traditional CG approach.

Table 1 : Test problems

problem	function name	size
1	Extended Rosenbrock	100000
2	Extended Powell Singular	100000
3	Trigonometric	100000
4	Oren	10000
5	Cube	2
6	Extended Wood	4-100000
7	Beale	2
8	Helical Valley	3
9	Penalty I	2
10	Watson function	4
11	Variably dimensioned function	2-100000
12	Generalized Shallow function	2-100000
13	Wood function	2-100000
14	Shallow	2-100000
15	Tridiagonal	2-100000
16	Helical Valley	2-1000
17	Dixon	2-10000
18	Oren and Spedicato Power function	2-10000

Table 2: Overall Results (886 problems)

Method	Evaluations	Iterations	Time (sec.)	Scores
BFGS	86401	73090	39171.18	101
	100.0%	100.0%	100.0%	100.0%
Alg.(4)	80364	70404	35547.68	247
	93.01%	96.32%	90.75%	
Alg.(5)	79164	67335	34874.71	317
	91.61%	92.12%	89.03%	
Alg.(11)	86426	69262	38718.99	221
	100.1%	94.77%	98.85%	

References

[1] M. Al-Baali, New property and global convergence of the Fletcher–Reeves method with inexact line searches, *IMA J. Numer. Anal.,* **5** (1985), 122-124.

[2] W.R. Boland, E.R. Kamgnie, J.S. Kowalik, A conjugate gradiant optimization method invariant to nonlinear scaling, *JOTA,* **27** (1979), 11-19.

[3] N. Anderi, A scaled BFGS preconditioned conjugate gradient algorithm for unconstrained optimization, *Applied Mathematics Letters,* **20** (2007), 645-650.

[4] C.G. Broyden, The convergence of a class of double-rank minimization algorithms - Part 2: The new algorithm, *J. Inst. Math. Applic.*, **6** (1970), 222-231.

[5] R.H. Byrd, R.B. Schnabel, G.A. Shultz, Parallel quasi-Newton methods for unconstrained optimization, *Math. Programing*, **42** (1988), 273-306.

[6] Y.H. Dai, Y. Yuan, A nonlinear conjugate gradient method with a strong global convergence property, *SIAM J. Optim.*, **10** (1999), 177-182.

[7] J.E. Dennis, R.B. Schnabel, Least change secant updates for quasi-Newton methods, *SIAM Review*, **21** (1979), 443-459.

[8] R. Fletcher, *Practical Methods of Optimization* (Second Edition), Wiley, New York (1987).

[9] R. Fletcher, A new approach to variable metric algorithms, *Comput. J.*, **13** (1970), 317-322.

[10] I.N. Fried, N-step conjugate gradient minimization scheme for non-quadratic functions, *AIAA J*, **9** (1971), 149-154.

[11] J.A. Ford, I.A.R. Moghrabi, Using function-values in multi-step quasi-Newton methods, *J. Comput. Appl. Math.*, **66** (1996), 201-211.

[12] W. Hager, H.C. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search, *SIAM J. Optim.*, **16** (2005), 170-192.

[13] H.Y. Huang, Uninfied approach to quadratically convergent algorithms for function minimization, *J. Optim. Theory Appl.*, **5** (1970), 405-423.

[14] I.A.R. Moghrabi, Numerical experience with multiple update quasi-newton methods for unconstrained optimization, *Journal of Mathematical Modeling and Algorithms*, **6** (2007), 231-238.

[15] I.A.R. Moghrabi, Implicit extra-update multi-step quasi-Newton methods, *Int. J. Operational Research*, **28** (2017), 69-81.

[16] I.A.R. Moghrabi, New two-step conjugate gradient method for unconstrained optimization, *International Journal of Applied Mathematics*, **33**, No 5 (2020), 853-866; doi: 10.12732/ijam.v33i5.8.

[17] J.J. Moré, B.S. Garbow, K.E. Hillstrom, Testing unconstrained optimization software, *ACM Trans. Math. Softw.*, **7** (1981), 17-41.

[18] M.J.D. Powell, Restart procedures for the conjugate gradient method, *Math. Program*, **12** (1977), 241-254.

[19] D. Salane, R.P. Tewarson, On symmetric minimum norm updates, *IMA Journal of Numerical Analysis*, **9** (1983), 235-240.

[20] D.F. Shanno, Conditioning of quasi-Newton methods for function minimization, *Math. Comp.,* **24** (1970), 647-656.

[21] D.F. Shanno, K.H. Phua, Matrix conditioning and nonlinear optimization, *Math. Programming*, **14** (1978), 149-160.

[22] E. A Spedicato, Variable metric method for function minimization derived from invariancy to nonlinear scaling, *JOTA*, **20** (1976), 30-42.

[23] A. Tassopoulos, C.A. Story, Conjugate direction method based on a non-quadratic model, *JOTA*, **43** (1984), 1-9.

[24] P. Wolfe, Convergence conditions for ascent methods II: Some corrections, *SIAM Rev.*, **13** (1971), 185-188.

[25] L. Wong, M. Cao, F. Xing, The new spectral conjugate gradient method for large-scale unconstrained optimisation, *J. Inequal Appl.*, **111** (2020), 28-37.