Iraqi Journal of Statistical Science (12) 2007
P.P. [1-12]

Generalized Implicit-Update in Multi-Step QN Methods

Abbas Y. AL-Bayati[*] Ban A. Metras[**]

Abstract

In this paper, we have generalized the implicit update of the Quasi-Newton's condition. We have here investigated a four; five and n-step update algorithm. We have applied the cases at four and five- step numerically and we have compared these cases with other QN-algorithms. The numerical results of the proposed algorithm show that the new algorithm was better than others.

تعميم التحسين الضمني في طرق أشباه نيوتن متعدد الخطوات

الملخص

في هذا البحث تم تعميم التحسين الضمني على شرط (أشباه نيوتن) في الخوارزميات الشبيهة بخوارزمية نيوتن. إذ قمنا بتعميم التحسين الضمني المكون من n من الحدود حيث بدأنا بتحسين ضمني مكون من أربعة حدود ثم من خمسة حدود وعممناها إلى n من الحدود. تم معالجة الحالتين عند الحدود الأربعة والخمسة عددياً وتم مقارنة النتائج مع الخوارزميات السابقة لأشباه نيوتن وأثبتت الخوارزمية المقترحة كفاءتها من بين تلك الخوارزميات.

---

* Prof./ Dept. of Mathematics/ College of Computers Sciences and Mathematics/University of Mosul
** Ass. Prof. / Dept. of Statistics/ College of Computers Sciences and Mathematics/University of Mosul

Received: 18/ 7 /2006 ____________________ Accepted: 5/ 6 / 2007

1. Introduction:

One problem, which is more widely used, is QN method, where approximate Hessian or inverse Hessian is updated at each iteration, while the gradients are supplied. The basic requirement for the updating formula is that the QN condition (secant condition i.e $B_{k+1}s_k = y_k$) where s_k, y_k are defined as:

$s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$.

where x_k is the point at k iteration, g_k is the gradient at k iteration and B_{k+1} is the inverse Hessian.

We consider QN methods for unconstrained optimization problems (min f(x), $x \in R^n$), where the basic idea behind the QN formulas is to update B_{k+1} from B_k in some computational cheap ways while ensure secant condition, and the computation of the update should be relatively cheap.

2. One Step Method (BFGS Method):

The BFGS method is one of the most efficient QN methods for unconstrained optimization. This algorithm was proposed by Broyden, Fletcher, Goldfarb and Shanno in (1970). BFGS method has a search direction computed by:

$$d_k = -H_k g_k \tag{1}$$

where H_k is a symmetric and positive definite matrix at the k-th iteration. The next iterate is given by:

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

where λ_k is the step size that satisfies the strong Wolfe condition ($f(x_{k+1}) < f(x_k)$) [Ahmed, 2005].

The approximation matrix is updated by:

$$\begin{aligned} H_{k+1} &= H_k - \frac{H_k s_k s_k^T H_k}{s_k^T H_k s_k} + \frac{y_k y_k^T}{y_k^T s_k} + w_k w_k^T \\ &= BFGS(H_k, s_k, y_k) \end{aligned} \tag{3}$$

where $w_k = (y_k^T H_k y_k) s_k - (y_k^T s_k) H_k y_k$. We call eq.(3) by one step method. [Dai, 2002] and [Nocedal et al., 1987].

2.2 BFGS (One Step Algorithm):

Step 1: Given $x_0 \in R^n$, set H_0=I, compute $g_0=\nabla f(x_0)$. If $\|g_0\| \le 10^{-5}$ then

stop, Otherwise, set k=1 and continue.

Step 2: Set $d_k = -H_k g_k$.

Step 3: Compute $x_{k+1} = x_k + \lambda_k d_k$.

Step 4: If $\|g_{k+1}\| < \varepsilon$ then stop else continue.

Step 5: Update H_k by the correction matrix to get H_{k+1} defined by:

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{s_k^T y_k} + w_k w_k^T$$

Where

$$w_k = (y_k^T H_k y_k) s_k - (y_k^T s_k) H_k y_k$$

Step 6: Set k=k+1 and go to step 2.

3. Implicit update method:

Ford in 2001 developed a two-step implicit algorithm denoted as two-step QN method which are very similar to the standard (one-step) method in very respect, except that the Hessian approximation H_{k+1} in the standard method is constrained to satisfy the relation ($B_{k+1} s_k = y_k$). Where in the two-step methods, it must satisfy a modified relation of the form:

$$B_{k+1}(s_k - \alpha s_{k-1}) = (y_k - \alpha y_{k-1}) \quad (4)$$

where α is positive scalar and defined by:

$$\alpha = \hat{\delta}(\hat{\delta} + 2) \quad (5)$$

So we can rewrite (4) by:

$$B_{k+1} r_k = w_k \quad (6)$$

where $r_k = s_k - \hat{\delta}(\hat{\delta} + 2) s_{k-1}$ (7)

$$w_k = y_k - \hat{\delta}(\hat{\delta} + 2) y_{k-1} \quad (8)$$

The relation (4) or (6) is dependant on cubic interpolating curve $\{x(\tau)\}$ and $\{h(\tau)\}$, where $\{x(\tau)\}$ interpolates the two latest iterates x_{k+1}, x_k, while $\{h(\tau)\}$ interpolates the corresponding gradient value $g(x_{k+1})$ and $g(x_k)$ i.e.

$$x(\tau_j)=x_{i+j-1}, \qquad j=0,1 \tag{9}$$

$$g(\tau_j)=g(x_{i+j-1}), \; j=0,1 \tag{10}$$

A Suitable matrix B_{k+1} satisfying (4) or (6) may then be obtained by using the BFGS formula. [Tharmlikit, 2001].

Now we define $\hat{\delta}$ by:

$$\begin{aligned}\hat{\delta} &= \tau_1 - \tau_0 \\ &= \left\| x(\tau_1) - x(\tau_0) \right\|_M \\ &= \left\| x_{k+1} - x_k \right\|_M \\ &= \left\| s_k \right\|_M \end{aligned} \tag{11}$$

where $\left\| s_k \right\|_M = \{ s_k^{\;T} M s_k \}^{0.5}$ for general s_k. [Ford and Moghrabi, 1993; 1994]

Ford proved that two-step iterations are alternated with standard one-step iterations, so that $B_k s_{k-1} = y_{k-1}$ on every two-step iteration that satisfies the QN condition of the one-step ($B_{k+1} s_k = y_k$): Substitute the value of r_k and w_k by (7) and (8) then we obtain:

$$\hat{B}_{k+1}(s_k - \hat{\delta}(\hat{\delta}+2)s_{k-1}) = y_k - \hat{\delta}(\hat{\delta}+2)y_{k-1}$$

$$\hat{B}_{k+1}\, s_k - \hat{\delta}(\hat{\delta}+2)\,\hat{B}_{k+1}\, s_{k-1} = y_k - \hat{\delta}(\hat{\delta}+2)y_{k-1}$$

$$\begin{aligned}\hat{B}_{k+1}\, s_k &= y_k - \hat{\delta}(\hat{\delta}+2)y_{k-1} + \hat{\delta}(\hat{\delta}+2)\,\hat{B}_{k+1}\, s_{k-1} \\ &= y_k - \hat{\delta}(\hat{\delta}+2)[y_{k-1} - \hat{B}_{k+1}\, s_{k-1}]\end{aligned}$$

Since $(\hat{B}_{k+1}\, s_{k-1} = y_{k-1})$ then

$$\hat{B}_{k+1}\, s_k = y_k$$

[Ford and Moghrabi, 1996]

4. Ford and Moghrabi (The Two - Step Implicit Update) Algorithm:

Step 1: Given $x_0 \in R^n$, set $H_0=I$, compute $g_0=\nabla f(x_0)$. If $\|g_0\|\le 10^{-5}$ then

stop, Otherwise, set k=1 and continue

Step 2: Set $d_k = -H_k g_k$.

Step 3: Compute $x_{k+1} = x_k + \lambda_k d_k$.

Step 4: If $\|g_{k+1}\| < \varepsilon$ then stop else continue.

Step 5: If k=1 then $r_k=s_k$ and $w_k=y_k$, i.e. we use standard BFGS formula,

Else calculate $\{\tau_j\}_{j=0}^{1}$ and compute $\hat{\delta}$ from Eq.(11) and compute r_k,

w_k from $r_k = s_k - \hat{\delta}(\hat{\delta}+2)s_{k-1}$

$$w_k = y_k - \hat{\delta}(\hat{\delta}+2)y_{k-1}$$

Step 6: Update H_k by using:

$$H_{k+1} = H_k - \left(1 + \frac{w_k^T H_k w_k}{r_k^T w_k}\right)\frac{r_k r_k^T}{r_k^T w_k} - \left(\frac{H_k w_k r_k^T + r_k w_k^T H_k}{r_k^T w_k}\right)$$

that satisfying $H_{k+1} r_k = w_k$

Step 7: Set k=k+1 and go to step 2.

5. The 3-Step Implicit Update Algorithm:

Al-Bayati and Ahmed in 2005 developed a new implicit QN methods which the matrix M is the result of alternate three-step update of B_k :

The 3-Step Implicit Update (Al-Bayati and Ahmed, 2005) Algorithm:

Step 1: Given $x_0 \in R^n$, set H_o=I, compute $g_0=\nabla f(x_0)$. If $\|g_0\| \leq 10^{-5}$ then

stop, Otherwise, set k=1 and continue

Step 2: Set $d_k = -H_k g_k$.

Step 3: Compute $x_{k+1} = x_k + \lambda_k d_k$.

Step 4: If $\|g_{k+1}\| < \varepsilon$ then stop else continue.

Step 5: If k=1 then $r_k=s_k$ and $w_k=y_k$, i.e. we use standard BFGS formula,

Else calculate $\{\tau_j\}_{j=0}^{1}$ and compute $\hat{\delta}$ from Eq.(11) and compute

r_k, w_k from

$$r_k = s_k - \left(\frac{\hat{\delta}}{\hat{\delta}-2}\right)s_{k-1} - \left(\frac{\hat{\delta}}{\hat{\delta}-2}\right)s_{k-2}$$

$$w_k = y_k - \left(\frac{\hat{\delta}}{\hat{\delta}-2}\right)y_{k-1} - \left(\frac{\hat{\delta}}{\hat{\delta}-2}\right)y_{k-2}$$

Step 6: Update H_k by using:

$$H_{k+1} = H_k - \left(1 + \frac{w_k^T H_k w_k}{r_k^T w_k}\right)\frac{r_k r_k^T}{r_k^T w_k} - \left(\frac{H_k w_k r_k^T + r_k w_k^T H_k}{r_k^T w_k}\right)$$

that satisfying $H_{k+1} r_k = w_k$

Step 7: Set k=k+1 and go to step 2.

6. Generalized Implicit-Update in QN Methods:

Here we extended the three-step update to the four-step update by using four terms and we extended the four-step update to the five-step update by using five terms and hence we generalize the process to n-terms as the following:

6.1 4-Step Implicit Update Forms:

Let $\psi = \dfrac{(\hat{\delta}+2)}{\hat{\delta}}$, then:

$$\hat{B}_{k+1}(s_k - \psi s_{k-1} - \psi s_{k-2} - \psi s_{k-3}) = y_k - \psi y_{k-1} - \psi y_{k-2} - \psi y_{k-3}$$

$$\hat{B}_{k+1} s_k - \psi \hat{B}_{k+1} s_{k-1} - \psi \hat{B}_{k+1} s_{k-2} - \psi \hat{B}_{k+1} s_{k-3}) = y_k - \psi y_{k-1} - \psi y_{k-2} - \psi y_{k-3}$$

$$\hat{B}_{k+1} s_k = y_k - \psi(y_{k-1} - \hat{B}_{k+1} s_{k-1}) - \psi(y_{k-2} - \hat{B}_{k+1} s_{k-2}) - \psi(y_{k-3} - \hat{B}_{k+1} s_{k-3})$$

Since $(\hat{B}_{k+1} s_{k-1} = y_{k-1})$ then

$$\hat{B}_{k+1} s_k = y_k$$

6.2 5-Step Implicit Update Forms:

Let $\psi = \dfrac{(\hat{\delta}+2)}{\hat{\delta}}$, then:

$$\hat{B}_{k+1}(s_k - \psi s_{k-1} - \psi s_{k-2} - \psi s_{k-3} - \psi s_{k-4}) = y_k - \psi y_{k-1} - \psi y_{k-2} - \psi y_{k-3} - \psi y_{k-4}$$

$$\hat{B}_{k+1} s_k - \psi \hat{B}_{k+1} s_{k-1} - \psi \hat{B}_{k+1} s_{k-2} - \psi \hat{B}_{k+1} s_{k-3} - \psi \hat{B}_{k+1} s_{k-4})$$
$$= y_k - \psi y_{k-1} - \psi y_{k-2} - \psi y_{k-3} - \psi y_{k-4}$$

$$\hat{B}_{k+1} s_k = y_k - \psi(y_{k-1} - \hat{B}_{k+1} s_{k-1}) - \psi(y_{k-2} - \hat{B}_{k+1} s_{k-2}) - \psi(y_{k-3} - \hat{B}_{k+1} s_{k-3})$$
$$- \psi(y_{k-4} - \hat{B}_{k+1} s_{k-4})$$

Since $(\hat{B}_{k+1} s_{k-1} = y_{k-1})$ then

$$\hat{B}_{k+1} s_k = y_k$$

Note:

6.3 N-Step Implicit Update Forms:

We prove that N-step implicit form by using mathematical induction:

Assume that we extended all-step to (n-1)-step as:

$$\hat{B}_{k+1}(s_k - \psi s_{k-1} - \psi s_{k-2} - \ldots - \psi s_{k-(n-1)}) = y_k - \psi y_{k-1} - \psi y_{k-2} - \ldots - \psi y_{k-(n-1)}$$

$$\hat{B}_{k+1}\, s_k - \psi \hat{B}_{k+1}\, s_{k-1} - \psi \hat{B}_{k+1}\, s_{k-2} - \ldots - \psi \hat{B}_{k+1}\, s_{k-(n-1)})$$

$$= y_k - \psi y_{k-1} - \psi y_{k-2} - \ldots - \psi y_{k-(n-1)}$$

$$\hat{B}_{k+1}\, s_k = y_k - \psi(y_{k-1} - \hat{B}_{k+1}\, s_{k-1}) - \psi(y_{k-2} - \hat{B}_{k+1}\, s_{k-2})$$

$$- \ldots - \psi(y_{k-(n-1)} - \hat{B}_{k+1}\, s_{k-(n-1)})$$

Since $(\hat{B}_{k+1}\, s_{k-(n-1)} = y_{k-(n-1)})$ then

$$\hat{B}_{k+1}\, s_k = y_k$$

7. The Generalized N-Step Algorithm:

Step 1: Given $x_0 \in R^n$, set $H_0=I$, compute $g_0=\nabla f(x_0)$. If $\|g_0\| \le 10^{-5}$ then

stop, Otherwise, set k=1 and continue

Step 2: Set $d_k = -H_k g_k$.

Step 3: Compute $x_{k+1} = x_k + \lambda_k d_k$.

Step 4: If $\|g_{k+1}\| < \varepsilon$ then stop else continue.

Step 5: If k=1 then $r_k=s_k$ and $w_k=y_k$, i.e. we use standard BFGS formula,

Else calculate $\{\tau_j\}_{j=0}^{1}$ and compute $\hat{\delta}$ from Eq.(11) and compute

r_k, w_k from

$$r_k = s_k - \psi s_{k-1} - \psi s_{k-2} - \ldots - \psi s_{k-(n-1)}$$

$$w_k = y_k - \psi y_{k-1} - \psi y_{k-2} - \ldots - \psi y_{k-(n-1)}$$

Where: $\psi = \dfrac{(\hat{\delta} + 2)}{\hat{\delta}}$

Step 6: Update H_k by using:

$$H_{k+1} = H_k - \left(1 + \frac{w_k^T H_k w_k}{r_k^T w_k}\right)\frac{r_k r_k^T}{r_k^T w_k} - \left(\frac{H_k w_k r_k^T + r_k w_k^T H_k}{r_k^T w_k}\right)$$

that satisfying $H_{k+1} r_k = w_k$

Step 7: Set k=k+1 and go to step 2.

8. Numerical Results:

In order to assess the performance of the new implicit N-step QN methods for the cases N=4, N=5, we tested these cases by using (5) nonlinear test functions with dimension N=100.

All results are obtained using Pentium 3. All programs are written in FORTRAN 90 language and for all cases the stopping criterion taken to be $\|g_{k+1}\| < \varepsilon$, where $\varepsilon = 10^{-5}$.

The comparative performance for all of these methods are evaluated by considering NOF, NOI, where NOF is the number of function evaluations and NOI is the number of iterations.

All the methods, in this search use the same exact line search strategy which is the quadratic interpolation technique directly adapted from [Bunday, 1984].

In table (1), we have compared our new methods 4-step and 5-step with BFGS and 3-step methods. The numerical results of the proposed methods show that the new methods were better than others (i.e. when the steps are increasing, the results be good).

Table (1)
Comparison Among the BFGS; 2-step; 3-Step; 4-Step and 5-Step Methods at N=100

Test f.	BFGS NOF(NOI)	2-step NOF(NOI)	3-step NOF(NOI)	4-step NOF(NOI)	5-step NOF(NOI)
1	122(44)	280 (111)	198 (37)	177 (40)	125 (39)
2	310 (92)	334 (98)	299 (81)	253 (76)	222 (69)
3	27 (9)	33 (12)	24 (10)	25 (12)	20 (9)
4	62 (33)	82 (45)	67 (37)	53 (31)	48 (29)
5	113 (44)	263 (116)	60 (19)	65 (17)	23 (19)
Total	634(222)	992(382)	648(184)	573(176)	438(165)

9. Appendix:

1- Generalized Powell Function:

$$f=\sum_{i=1}^{n/4}[(x_{4i-3}-10x_{4i-2})^2+5(x_{4i-1}-x_{4i})^2+(x_{4i-2}-2x_{4i-1})^4+10(x_{4i-3}-x_{4i})^4],$$

$x_0=(3,-1,0,1;\dots)^T$.

2- Generalized Rosenbrock Function:

$$f=\sum_{i=1}^{n/2}[100(x_{2i}-x_{2i-1}^2)^2+(1-x_{2i-1})^2], \quad x_0=(-1.2,1;\dots)^T.$$

3- Generalized Sum of Quadratics Function:

$$f=\sum_{i=1}^{n}(x_i-i)^4, \quad x_0=(2;\dots)^T.$$

4- Generalized Dixon Function:

$$f=\sum_{i=1}^{n}[(1-x_i)^2+(1-x_n)^2+\sum_{i=1}^{n-1}(x_i^2-x_{i-1})^2], \quad x_0=(-1;\dots)^T.$$

5- Generalized Cubic Function:

$$f = \sum_{i=1}^{n/2} [100(x_{2i} - x_{2i-1}^3)^2 + (1 - x_{2i-1})^2], \quad x_0=(-1.2,1;\ldots)^T$$

10. References:

[1] Ahmed, H.I. (2005), "Performance of CG-Type Methods with ILS for Unconstrained Minimization", ph.D Thesis. University of Mosul.

[2] Broyden, C.W. (1970), " The Convergence of Class of Double Rank Minimization Algorithms: 2-The New Algorithm", J. Inst. Math. Appl., Vol. 6, pp. 222-231.

[3] Bunday, B.D. (1984), "Basic Optimization Methods", Edward Arnold, London.

[4] Dai, Y.H. (2002), "Convergence Properties of the BFGS Algorithm", Society for Industrial and Applied Mathematics (SIAM. J. Optm.), Vol. 13, pp. 693-701.

[5] Fletcher, R. (1970), "A New Approach to Variable Metric Algorithm", Computer J., Vol. 13, pp.317-322.

[6] Ford, A.J. and Moghrabi, A.I. (1993), "Alternate Parameter Choices for Multi-Step Quasi-Newton Methods" Optimization Methods and Software, Vol. 2, pp. 357-370.

[7] Ford, A.J. and Moghrabi, A.I. (1994), "Multi-Step Quasi-Newton Methods for Optimization", J. Comput. Math. Appl., Vol. 50, pp. 305-323.

[8] Ford, A.J. and Moghrabi, A.I. (1996), "Minimum Curvature Multi-Step Quasi-Newton Methods", J. Comput. Math. Appl., Vol. 31, pp. 179-186.

[9] Ford, A.J. (2001), "Implicit Update in Multi-Step Quasi-Newton

Method", J. Comput. Math. Appl., Vol. 42, pp. 1083-1091.

[10] Goldfarb, D. (1970), "A Family of Variable Metric Methods Derived by Variational Means", Math. Comp., Vol. 24, pp. 23-26.

[11] Nocedal, J.; Byrd, H.R. and Yuan, Y. (1987), " Global Convergence of a Class of Quasi-Newton Methods on Convex Problems", SIAM. J. Numer. Anal. Vol. 24, pp. 1171-1190.

[12] Shanno, D.F. (1970), "Conditioning of QN Methods for Function Minimization", Math. Comp., Vol. 24, pp.647-650.

[13] Tharmlikit, S. (2001), "Further Investigation of Multi-Step QN Method for Unconstrained Optimization", Ph. D. Thesis, University of Essex.