Discussion Paper #235

A GENERAL CLASS OF SYMMETRIC QUASI-NEWTON UPDATES

Avinoam Perry
Assistant Professor
Department of Decision Sciences

August 1976

Abstract

In this report we introduce a general class of symmetric quasi-Newton updates. We show that the DFP and the BFGS updates are special cases of this class and we present some computational experience with several new specific quasi-Newton algorithms.

I. Introduction

Families of quasi-Newton algorithms were first introduced by Huang [9] who generalized the updating formula of the unconstrained minimization algorithm:

$$(1) \quad x_{k+1} = x_k - \alpha_k S_k g_k$$

where $x_k \in E^n$ is the n dimensional starting point, α_k is a scalar minimizing the one dimensional program:

$$(2) \quad \min t(\alpha) = f(x_k - \alpha S_k g_k)$$

S_k is a matrix approximating the inverse Hessian of $f(x)$ and $g_k \equiv \nabla f(x_k)$.

The matrix S_k is updated from one stage to another as more information about the function and its first order derivatives become available. Huang generalization of the updating formula is presented below:

$$(3) \quad S_{k+1} = S_k + \beta_k p_k p'_k + \gamma_k S_k q_k q'_k S_k + \delta_k [p_k q'_k S_k + S_k q_k p'_k]$$

where p_k and q_k are the vectors

$$(4) \quad \begin{cases} p_k \equiv x_{k+1} - x_k \\ q_k \equiv g_{k+1} - g_k \end{cases}$$

Different choices of the parameters β, γ, and δ give the different updating formula. However, three conditions are mandatory for (3) to be classified as a quasi-Newton update:

a. $S_{k+1}q_k = p_k$

b. All direction vectors $p_k (k=1,2,\ldots n)$ are mutually conjugate with respect to the Hessian of f(x).

c. $S_{k+1}q_i = \lambda_i p_i$ for $i=1,2,\ldots k-1$ for a given scalar λ_i.

Imposing these conditions on (3) implies:

$$(5)\quad 1 + \gamma_k (q_k' S_k q_k) + \delta_k (p_k' q_k) = 0$$

It can be shown that updating formulas such as the DFP [3,5] and the BFGS [2,4,7,17] procedures are special cases of the Huang family. For example: by letting

$$(6)\begin{cases} \beta_k = 1/p_k' q_k \\ \gamma_k = -1/q_k' S_k q_k \\ \delta_k = 0 \end{cases}$$

one can obtain the DFP updating formula, and by setting:

$$(7)\begin{cases} \beta_k = \left[1 + \dfrac{q_k' S_k q_k}{p_k' q_k}\right] \dfrac{1}{p_k' q_k} \\ \gamma_k = 0 \\ \delta_k = -\dfrac{1}{p_k' q_k} \end{cases}$$

one obtains the BFGS update.

Different choices of β_k, γ_k, and δ_k may be used in order to generate unsymmetric updates such as the ones introduced by Pearson [14].
A somewhat different approach to classifying quasi-Newton methods was introduced by Goldfarb [7] who explored the interrelationships among

Broyden's rank one [1], DFP, BFGS and an additional update introduced by Greenstadt [8]. He also showed that there exists a one parameter family of correction terms which contains all four updates mentioned above. Fletcher [4] obtained equivalent results to the ones introduced by Goldfarb [7]. His one parameter family is defined as a linear combination of the DFP and the BFGS updates which are, naturally, special cases of Fletcher's class. The expression representing the above family as introduced by Fletcher is:

(8) $S_{BFGS} = S_{DFP} + \theta vv'$

where θ is the parameter in question and v_k is the vector:

(9) $v_k = (q_k' S_k q_k)^{1/2} \left[\frac{p_k}{p_k' q_k} - \frac{S_k q_k}{q_k' S_k q_k} \right]$

In this paper we introduce a new expression representing a family of quasi-Newton updates denoted by S_k. We show that if S_k is symmetric and positive definite, then the resulted updated matrix S_{k+1} is always symmetric and positive definite. We show that the BFGS and the DFP updates are special cases of this family and we compare the performance of these two well known updates with two arbitrary members of the family. The family defined above is shown to have a dual family of quasi-Newton updates, where the complementary updates of the DFP and the BFGS (BFGS and DFP respectively) are, again, special members of that dual family.

II. Derivation of the Quasi-Newton Family of Updates.

Given a positive definite symmetric matrix S_k we define the following relationships as the updating formula of a family of quasi-Newton algorithms

$$(10) \quad S_{k+1} = \left[I - \frac{y_k q_k'}{y_k' q_k} \right] S_k \left[I - \frac{q_k y_k'}{y_k' q_k} \right] + \frac{p_k p_k'}{p_k' q_k}$$

where:

I is a n x n identity matrix

$p_k = x_{k+1} - x_k$

$q_k = g_{k+1} - g_k \equiv \nabla f(x_{k+1}) - \nabla f(x_k)$

The vector y_k is the one vector which determines the specific algorithm in question. By letting $y_k = p_k$ (10) becomes the BFGS algorithm, and by letting $y_k = S_k q_k$ (10) becomes the DFP updating formula.

An interesting feature of the family in (10) is that it retains the property $S_{k+1} q_k = p_k$ even if y_k is a nonnull random vector. This property is accompanied by the facts that if S_k is positive definite, then S_{k+1} is assured of positive definiteness, and if the line search (associated with the one dimensional program minimize $h(\alpha) = f(x_k - \alpha S_{k+1} g_{k+1})$) is accurate, then p_{k+1} is conjugate to p_k. However, if y_k is a random vector, S_k does not necessarily converge to F (the Hessian of f(x)), and properties (2) and (3) of the Huang family are not part of an algorithm relying on a random vector y_k.

In order to ensure the properties:

$$\left.\begin{array}{ll} \text{a.} & p_{k+1}' F p_i = 0 \\ \text{b.} & S_{k+1} q_i = p_i \end{array}\right\} \text{ for } i=1,2,\ldots k-1$$

y_k must be equal to either p_k, or $S_k q_k$ or any linear combination of these two vectors. The above is proved by induction using the fact that $p_{k+1}' F p_k = 0$ and that

(11) $$S_{k+1}q_{k-1} = \left[S_k - \frac{S_k q_k y_k'}{q_k' y_k} - \frac{y_k q_k' S_k}{y_k' q_k} + \frac{y_k q_k' S_k q_k y_k'}{(y_k' q_k)^2} + \frac{p_k p_k'}{p_k' q_k} \right] q_{k-1}$$

It follows immediately that if y_k is any linear combination, of p_k and $S_k q_k$, and if $S_k q_{k-1} = p_{k-1}$, then:

(12) $$S_{k+1}q_{k-1} = S_k q_{k-1} = p_{k-1}.$$

Based on the above, we introduce two arbitrary members of the family. The first member, S1, and the second member, S2, are, given in (13) and (14) respectively:

(13) $$S1 = \left[I - \frac{(p_k + S_k q_k) q_k'}{(p_k + S_k q_k)' q_k} \right] S_k \left[I - \frac{q_k (p_k + S_k q_k)'}{q_k' (p_k + S_k q_k)} \right] + \frac{p_k p_k'}{p_k' q_k}$$

(14) $$S2 = \left[I - \frac{(p_k - S_k q_k) q_k'}{(p_k - S_k q_k)' q_k} \right] S_k \left[I - \frac{q_k (p_k - S_k q_k)'}{q_k' (p_k - S_k q_k)} \right] + \frac{p_k p_k'}{p_k' q_k}$$

Computational performance of these two arbitrary quasi-Newton updates is contrasted with the more well known members of the family; the DFP and the BFGS updates in another section of this paper.

III. A Dual Family of Quasi-Newton Updates

Duality in unconstrained nonlinear optimization was first introduced by Fletcher [4] who referred to the inverse transformations of the DFP and BFGS updates as dual updates. Oren and Spedicato [13] applied the duality concept to the self scaling variable metric procedure [11,12] and showed that dual updates of either the DFP or the BFGS procedures can be obtained directly upon replacing p_k by q_k, S_k by T_k, $\theta=0$ by $\bar{\theta}=1$ and vice versa, where $T_k = S_k^{-1}$

and

$$(14)\quad S_{k+1} = S_k - \frac{S_k q_k q_k' S_k}{q_k' S_k q_k} + \frac{p_k p_k'}{p_k' q_k} + \theta\, v_k v_k'$$

$$(15)\quad T_{k+1} = T_k - \frac{T_k p_k p_k' T_k}{p_k' T_k p_k} + \frac{q_k q_k'}{p_k' q_k} + \bar{\theta} w_k w_k'$$

v_k is defined in (9), and

$$(16)\quad w_k = (p_k' T_k p_k)^{1/2} \left[\frac{q_k}{p_k' q_k} - \frac{T_k p_k}{p_k' T_k p_k} \right]$$

Using the above updating procedures they showed that if $S_k T_k = I$, then it follows that $S_{k+1} T_{k+1} = I$ as well.

Applying the same principles as those of the above authors, we define a dual family of quasi-Newton updates:

$$(17)\quad T_{k+1} = \left[I - \frac{z_k p_k'}{z_k' p_k} \right] T_k \left[I - \frac{z_k p_k'}{p_k' z_k} \right] + \frac{q_k q_k'}{q_k' p_k}$$

where the vector z_k is the one which determines the specific algorithm in question. A quasi-Newton algorithm which applies a dual update in constructing the direction vector d_{k+1} is based on the equation:

$$(18)\quad d_{k+1} = -T_{k+1}^{-1}\, g_{k+1}$$

where $g_{k+1} \equiv \nabla f(x_{k+1})$

Given that $T_k = S_k^{-1}$, then by letting $z_k = q_k$, (17) and (18) become the DFP method, and by letting $z_k = T_k p_k$, (17) and (18) become the

BFGS updating formula.

By letting z_k be any linear combination of q_k and $T_k p_k$ and applying (17) and (18) we obtain a dual family of quasi-Newton methods which possesses the properties:

a. $T^{-1}_{k+1} q_k = p_k$

b. $q_k' F^{-1} q_i = 0$ for $i=1,2,...k-1$

c. $T^{-1}_{k+1} q_i = p_i$ for $i=1,2,...k-1$

The proof for the above statement is equivalent to the one given in the previous section for the primal case and we, therefore, do not repeat it. Also, note that the second property above is equivalent to the conjugate directions property of the primal family.

Although, constructing quasi-Newton direction vectors by resorting to a dual method such as (17) - (18) is somewhat inefficient due to the fact that each iteration involves an extra matrix inversion, we explored the computational performance of two arbitrary members of the dual family above. Note that the inefficiency discussed above may be eliminated if one comes up with a method for comupting T^{-1}_{k+1} directly as done in the case of the DFP and BFGS methods. The two arbitrary members are denoted as T1 and T2 and their formulas are given in (19) and (20) respectively.

$$(19) \quad T1 = \left[I - \frac{(q_k - T_k p_k) p_k'}{(q_k - T_k p_k)' p_k} \right] T_k \left[I - \frac{p_k (q_k - T_k p_k)'}{p_k' (q_k - T_k p_k)} \right] + \frac{q_k q_k'}{q_k' p_k}$$

$$(20)\quad T2 = \left[I - \frac{(q_k+T_kp_k)p_k'}{(q_k+T_kp_k)'p_k} \right] T_k \left[I - \frac{p_k(q_k+T_kp_k)'}{p_k'(q_k+T_kp_k)} \right] + \frac{q_kq_k'}{q_k'p_k}$$

In the following section we provide results of our computational experience with five arbitrary methods. These results are contrasted with the performance of the DFP and the BFGS procedures under equivalent computational conditions.

IV. Computational Experience

Experiments with the two arbitrary members of the primal quasi-Newton family: S1 and S2, the two arbitrary members of the dual quasi-Newton family: T1 and T2,and an update constructed by letting y_k in (10) be a random vector, involved three well known test problems which are denoted here as functions I through III respectively (see appendix for a detailed description of each function and its source). All problems were solved by each one of the algorithms above under two different line search accuracy measures. The line search technique applied is the well known quadratic interpolation method. In order to insure successful implementation of the line search procedure an effort was made to secure a unimodal region before the first interpolation was performed. We also define "number of iterations per line search" as the number of times the quadratic interpolation was performed along a given direction.

Another measure of accuracy used in our study is δ where:

$$(21)\quad \delta > \frac{|p(\alpha^*) - f(\alpha^*)|}{|p(\alpha^*)|}$$

and where:

$p(\alpha^*)$ is the value of the interpolating polynomial at the point $x_k+\alpha^* d_k$, and $f(\alpha^*)$ is the value of the function at the same point.

The term "stage" is defined as the step carrying a point x_k along a direction d_k to a new point $x_{k+1}= x_k+\alpha_k d_k$. It follows that the total number of stages per algorithm is equal to the total number of gradient evaluations.

The statistics "total number of function evaluations" includes the number of gradient evaluations multiplied by n (if one is interested in number of function evaluations not including gradient evaluations, the total number of stages multiplied by n should be subtracted from the total number of function evaluations). Each time a direction vector pointed upwards rather than downwards it was replaced by the direction of steepest descent.

The stopping rule applied throughout was

(22) $|\nabla f(x^*)| \leq 0.0001$

If an experimental run exceeded a given number of stages before reaching the point x* in (20) the run was terminated by the operator.

All computer programs were coded in APL using interactive mode [15] and were run on the CDC6400 computer at Northwestern University.

In the following tables we present our computational results under two measures of line search accuracy denoted as mode 1 (N=5, δ=.01) and mode 2 (N=1) where the variable N stands for the maximum number of iterations per one dimensional search and δ is defined as in (21). The one dimensional search was terminated whenever one of these constraints became active.

Conclusions

As reflected by our computational results the two new primal quasi-Newton updates, S1 and S2 exhibited a reasonable average performance which could be regarded as competitive with the DFP and BFGS updates respectively. The dual methods T1 and T2 were somewhat inferior to S1 and S2, a fact that might be explained by the extra matrix inversion which contributed to an increased magnitude of rounding off errors. The performance of the random update was better than expected. It resembled computational results obtained by conjugate gradient codes under equivalent conditions of line search accuracy [16]. This outcome may be explained by the fact that although this method employs a random vector in the construction of its update, the new direction is conjugate to its predecessor.

Function #1
x_o = -1.2, 1

Algorithm	# stages		# function evaluations		Reported best value	
	Mode 1	Mode 2	Mode 1	Mode 2	Mode 1	Mode 2
DFP	23	25	224	207	$1.91 \cdot 10^{-11}$	$6.05 \cdot 10^{-12}$
BFGS	20	25	209	216	$2.85 \cdot 10^{-13}$	$8.23 \cdot 10^{-11}$
S1	23*	25	214*	208	$1.64 \cdot 10^{-4}$	$1.90 \cdot 10^{-10}$
S2	22	21	215	183	$1.02 \cdot 10^{-12}$	$1.04 \cdot 10^{-14}$
T1	21	23	203	208	$6.05 \cdot 10^{-12}$	$2.35 \cdot 10^{-14}$
T2	23	25*	231	209	$7.89 \cdot 10^{-10}$	$2.24 \cdot 10^{-2}$
Random	23	25	241	210	$4.36 \cdot 10^{-11}$	$5.86 \cdot 10^{-10}$

Function #2
x_o = -3,-1,-3,-1

Algorithm	# stages		# function evaluations		Reported best value	
	Mode 1	Mode 2	Mode 1	Mode 2	Mode 1	Mode 2
DFP	38*	44*	425*	520*	1.59	3.55
BFGS	38	44	436	478	$4.54 \cdot 10^{-14}$	$1.11 \cdot 10^{-14}$
S1	38*	44*	438*	468*	$5.06 \cdot 10^{-9}$	$8.71 \cdot 10^{-3}$
S2	38*	44	435*	474	$1.82 \cdot 10^{-7}$	$3.69 \cdot 10^{-12}$
T1	38*	44*	441*	559*	$4.16 \cdot 10^{-8}$	$1.46 \cdot 10^{-2}$
T2	38*	44*	420*	469*	$4.77 \cdot 10^{-5}$	$1.85 \cdot 10^{-1}$
Random	38*	44*	421*	481*	$6.48 \cdot 10^{-1}$	7.23

Function #3
x_o = 1,1,1,1

Algorithm	# stages		# function evaluations		Reported best value	
	Mode 1	Mode 2	Mode 1	Mode 2	Mode 1	Mode 2
DFP	16	24	201	231	$3.33 \cdot 10^{-9}$	$1.61 \cdot 10^{-10}$
BFGS	15	19	196	197	$1.63 \cdot 10^{-9}$	$5.79 \cdot 10^{-10}$
S1	16*	18	200*	194	$1.68 \cdot 10^{-8}$	$8.08 \cdot 10^{-10}$
S2	14	17	192	177	$1.41 \cdot 10^{-9}$	$1.24 \cdot 10^{-9}$
T1	15	21	201	214	$1.08 \cdot 10^{-9}$	$1.60 \cdot 10^{-10}$
T2	15	17	199	175	$2.33 \cdot 10^{-9}$	$3.91 \cdot 10^{-8}$
Random	16*	24*	208*	235*	$3.26 \cdot 10^{-4}$	$6.01 \cdot 10^{-2}$

*Terminated by operator.

Appendix

1: $100(x_2 - x_1^2)^2 + (1-x_1)^2$ [1]

5: $100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4-x_3^2)^2$ [2]

$+ (1-x_3)^2 + (1 - x_1)^2 + 90(x_4-x_3^2)^2$

$+ 19.8\ (x_2-1)(x_4-1)$

6: $(x_1 + 10x_2)^2 + 5(x_3-x_4)^2 + (x_2-2x_3)^4 + 10(x_1-x_4)^4$ [3]

---

[1] H.H. Rosenbrock, Computer J., 3: 175 (1960).

[2] C.F. Wood, Westinghouse Research Laboratories, (1968).

[3] M.J.D. Powell, Computer J., 5: 147 (1962).

References

[1] Broyden, C.G., "Quasi-Newton Methods and Their Application to Function Minimization", Math. Comp., 21, 1967 pp. 368-381.

[2] Broyden, C.G., "The Convergence of a Class of Double Rank Algorithms, Parts I and II," J. Inst. Math. Appl., 7, 1971, pp. 76-90, 222-236.

[3] Davidon, W.C., "Variable Metric Method for Minimization", Research and Development Report ANL-5990, U.S. Atomic Energy Commission, Argonne National Laboratories, 1959.

[4] Fletcher, R. "A New Approach to Variable Metric Algorithms", Comp. J., 13, 1970, pp. 317-322.

[5] Fletcher, R. and M.J.D. Powell, "A Rapidly Convergent Descent Method for Minimization", Comp.J., 6, 1963, pp. 163-168.

[6] Fletcher, R. and C.M. Reeves, "Function Minimization by Conjugate Gradients", Comp. J., 7, 1964, pp. 149-154.

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

[8] Greenstadt J. "Variations on Variable Metric Methods", Math. Comp. 24, 1970, pp. 1-22.

[9] Huang, H.Y., "Unified Approach to Quadradically Convergent Algorithms for Function Minimization J.O.T.A. 5, 1970, pp. 405-423.

[10] Luenberger, D.G., "Introduction to Linear and Nonlinear Programming", Addison-Wesley Publishing Co., 1973.

[11] Oren, S.S., "Self Scaling Variable Metric Algorithms, Part II", Management Science, Vol. 20, No. 5, 1974, pp. 863-874.

[12] Oren, S.S. and D.G. Luenberger, "Self Scaling Variable Metric Algorithms, Part I", Management Science, Vol. 20, No. 5, 1974.

[13] Oren, S.S. and E. Spedicato, "Optimal Conditioning of Self Scaling Variable Metric Algorithm", Math Prog. Vol 10. No. 1 Feb. 1976, pp. 70-90.

[14] Pearson, J.D., "Variable Metric Methods of Minimization", Comp. J., 13, 1969, pp. 171-178.

[15] Perry, Avinoam, "UCNLP-An Interactive Package of Programs for Unconstrained Nonlinear Optimization Purposes", A Working Paper, Nov. 1975, Revised May 1976.

[16] Perry, Avinoam, "A Modified Conjugate Gradient Algorithm", to appear in Operations Research.

[17] Shanno, D.F., "Conditioning of Quasi-Newton Methods for Function Minimization", Math. Comp. 24, 1970, pp. 647-656.