MATHEMATICS OF COMPUTATION, VOLUME 27, NUMBER 124, OCTOBER 1973

Self-Scaling Variable Metric Algorithms Without Line Search for Unconstrained Minimization*

By Shmuel S. Oren

Abstract. This paper introduces a new class of quasi-Newton algorithms for unconstrained minimization in which no line search is necessary and the inverse Hessian approximations are positive definite. These algorithms are based on a two-parameter family of rank two, updating formulae used earlier with line search in self-scaling variable metric algorithms. It is proved that, in a quadratic case, the new algorithms converge at least weak superlinearly. A special case of the above algorithms was implemented and tested numerically on several test functions. In this implementation, however, cubic interpolation was performed whenever the objective function was not satisfactorily decreased on the first "shot" (with unit step size), but this did not occur too often, except for very difficult functions. The numerical results indicate that the new algorithm is competitive and often superior to previous methods.

1. Introduction. This paper addresses the problem of minimizing a smooth real valued function $f(x)$ depending on an n-dimensional vector x, assuming the availability of the gradients $\nabla f(x) = g(x)$ for any given x. An important class of algorithms for solving this problem is the quasi-Newton methods also known as variable metric algorithms. In these methods, the successive points are obtained by the equation

$$x_{k+1} = x_k - \alpha_k D_k q_k, \tag{1}$$

where α_k is an appropriate positive scalar and D_k is an approximation to the inverse Hessian of f, inferred from the gradients at the previous points and updated at every iteration. For notational convenience, we omit the subscripts k whenever it is possible and denote by superscripts (*) and (**) the values corresponding to $k + 1$ and $k + 2$, respectively. The first algorithm of this type and still the most popular is the DFP algorithm introduced by Davidon [5] and further developed and simplified by Fletcher and Powell [8]. In this algorithm, α_k is chosen (by a line search) to minimize $f(x - \alpha Dg)$ and D is updated using the formula

$$D^* = D - Dqq'D/q'Dq + pp'/p'q, \tag{2}$$

where $p = -\alpha Dg$ and $q = g^* - g$. The method has three properties that are felt to underline the superior convergence characteristics it often displays.

(a) The matrices D are all positive definite provided that D_0 is chosen to be positive definite.

(b) If f is a positive definite quadratic function, the algorithm is identical to the conjugate gradient method and thus converges to the solution in at most n steps.

Received October 24, 1972; revised January 29, 1973.

AMS (MOS) subject classifications (1970). Primary 90C30.

Key words and phrases. Function minimization, unconstrained minimization, quasi-Newton methods, variable metric methods, self-scaling variable metric algorithms, scaling, quasi-Newton algorithms with line search, gradient methods, Hessian matrix inverse approximation, conditioning of search methods, convergence rates.

* This paper was presented at the Joint Automatic Control Conference, June, 1973.

(c) If f is a positive definite quadratic function, and if convergence to the solution requires the full n steps, then the nth approximation D_n is identically equal to the inverse Hessian.

Unfortunately, the last two properties are very sensitive to the line search accuracy which, as a result, significantly influences the performance of the algorithm (see Oren [18] and Luenberger [16]). Alternative algorithms, that have the above three properties, were described in [2], [3], [12], and [14]. However, all of these methods require line search and none of them displayed a significant advantage over the DFP method. Furthermore, in a recent paper, Dixon [7] proved that these algorithms are actually all equivalent even in nonquadratic cases.

The inconvenience and high computational effort involved in the line search required for DFP-like algorithms motivated several researchers in the past few years to develop algorithms in which it is not necessary to find the minimum along the direction of move at every step. Such methods, which are often referred to as algorithms without line search, were described in [6], [9], [17], [22] and [23]. In these algorithms, α is fixed beforehand (usually taken as unity) and corrected only if the objective function is not sufficiently decreased to ensure convergence. The correction is usually done either according to some "rule of thumb" (for example, try $\alpha = 1$, 0.1, 0.01, $\cdots$ etc.) or by interpolation. Most of these methods are based on a rank-one updating formula for D that ensures property (c) for arbitrary step size α. However, this formula does not ensure property (a), i.e., the D's are not guaranteed to be positive definite, which may cause failure of the algorithm. The different rank-one methods that were suggested contain additional features to ensure convergence either by forcing D to be positive definite or by controlling the directions of search. Evidently, none of these algorithms was reported to be significantly more efficient than the DFP method.

Fletcher [9] suggested an algorithm without line search that retains property (a) while relinquishing properties (b) and (c). In this algorithm, D is updated by formula (1) or another formula discovered by Broyden [3]; the formula used depends on whether or not $p'q < q'Dq$. While the experimental results given for this algorithm seem quite favorable, its theoretical support is presently lacking. It was only proved that if the function to be minimized is a positive definite quadratic function with Hessian H, then the eigenvalues of the matrix $R = H^{1/2}DH^{1/2}$ move monotonically to unity, while one of the eigenvalues equals unity. This property does not ensure decrease in the condition number of R which was shown by Luenberger [16] to have a major effect on the convergence properties of the variable metric algorithm when conjugacy is destroyed. On the contrary, if the problem is poorly scaled, the condition number of R may increase on the first step and become greater than the condition number of H. Such a situation is illustrated in Fig. 1. Since the tendency of the

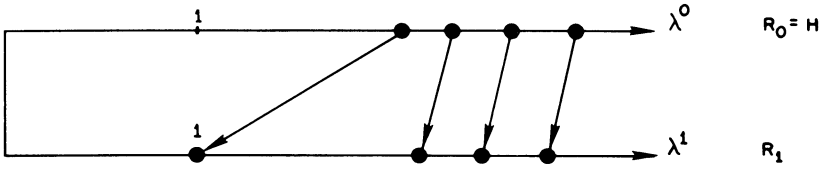

FIGURE 1. *Deterioration of the Eigenvalue Structure on the First Iteration in a Poorly Scaled Problem (Assuming $D_0 = I$)*

eigenvalues of R to unity on subsequent iterations is not strictly monotonic, this condition number may never recover, which can result in a convergence rate inferior to that of the steepest descent method. Such examples were given in [16], [18], and [19].

2. The Self-Scaling Updating Formula. The results obtained by Oren and Luenberger [20] led to developing the Self-Scaling Variable Metric Algorithms introduced by Oren ([18], [21]). In these algorithms, D is updated by the following two-parameter formula:

$$D^* = \left(D - \frac{Dqq'D}{q'Dq} + \theta vv'\right)\gamma + \frac{pp'}{p'q}, \quad \text{where} \tag{3}$$

$$v = (q'Dq)^{1/2}\left(\frac{p}{p'q} - \frac{Dq}{q'Dq}\right),$$

$$\gamma = \varphi\frac{g'p}{g'Dq} + (1-\varphi)\frac{p'q}{q'Dq} \tag{4}$$

and φ, $\theta \in [0, 1]$.

For $\gamma = 1$, (3) is identical to the convex class of updating formulae given by Fletcher [9]. For an arbitrary γ, (3) has the effect of scaling $f(x)$ by γ while using Fletcher's convex class of formulae. It can be shown (see [18] or [21]) that the γ defined by (4) guarantees that in a quadratic case the condition number of R is monotonically decreasing. Furthermore, these algorithms are invariant under scaling of the objective function or the variable. Sensitivity to such scaling was pointed out by Bard [1] to cause numerical instability of DFP-like algorithms. In addition to the new properties above, the SSVM algorithms have properties (a) and (b) but not property (c). The relinquished property (c) turns out to be important only when the number of variables is small, in which case the SSVM algorithm is less efficient than the DFP. In large-dimensional problems (above ten variables), however, the SSVM method was superior.

The relative success of Fletcher's algorithm (despite its lack of theoretical support) makes it only natural to try using the SSVM updating formulae in the same manner. Since property (c) is relinquished anyhow by eliminating the line search, it appears there is nothing to lose by using the SSVM formula while gaining the additional properties of these formulae mentioned above. This is certainly true from a theoretical point of view. We were actually able to prove (Section 4) that for a quadratic function, an algorithm in which $\alpha = 1$ and D is updated by (3) and (4) converges either in a finite number of steps or weak superlinearly.** The computational results obtained with such an algorithm (Section 5) proved to be favorable as well.

3. The Basic SSVM Algorithms Without Line Search. For notational convenience, we shall represent formula (3) by

$$D^* = \mathfrak{D}^{\theta}(D, \gamma, p, q) \tag{5}$$

and (4) by

$$\gamma = \gamma(\varphi).$$

** A sequence $\{e_k\}$ converges weak superlinearly if there exists an N such that $\lim_{k\to\infty}(e_{k+N}/e_k) = 0$.

Our basic algorithm can then be described as follows.

Algorithm 1.

Start with any x_0.

Step 1. Set $k = 0$ and choose D_0 to be a positive definite matrix.

Step 2. Set $p_k = -D_k g_k$ and obtain $x_{k+1} = x_k + p_k$, $g_{k+1} = \nabla f(x_{k+1})$ and $q_k = g_{k+1} - g_k$.

Step 3. Choose $\varphi_k \in [0, 1]$ and $\theta_k \in [0, 1]$.

Step 4. Set $D_{k+1} = \mathfrak{D}^{\theta_k}(D_k, \gamma(\varphi_k), p_k, q_k)$.

Step 5. Add 1 to k and return to Step 2.

Algorithm 1 is given in a rather crude form. In a practical implementation such as the one described in Section 5, one has to add the requirements that $p_k' q_k > 0$ (required for positive definiteness of D_{k+1}) and f is strictly monotonically decreasing to ensure convergence. Algorithm 1, however, is more convenient for analyzing convergence properties in a quadratic case where the additional features are not necessary.

4. Convergence in a Quadratic Case. In this section, we analyze the convergence properties of Algorithm 1 when applied to a positive definite quadratic function with Hessian H. The matrices of interest in this case are $R = H^{1/2} D H^{1/2}$ which characterize the closeness of the approximations D to H^{-1}. By pre- and post-multiplying (3) by $H^{1/2}$ and using the relation $q = Hp$, one can obtain an updating formula for R which turns out to have the same form as (3), only that D is replaced by R while p and q are replaced by $z = H^{1/2} p$. Thus, using the notation introduced above, we can write

$$R^* = \mathfrak{D}^{\theta}(R, \gamma, z, z). \tag{6}$$

$\gamma(\varphi)$ can also be expressed in terms of R and z in the form

$$\gamma(\varphi) = \varphi \frac{z' R^{-1} z}{z' z} + (1 - \varphi) \frac{z' z}{z' R z}. \tag{7}$$

We now define a scalar function

$$\Omega(R) = \operatorname{Tr}(R) \operatorname{Tr}(R^{-1}). \tag{8}$$

First, we show that $\Omega(R)$ is bounded below by n^2 for any positive definite $n \times n$ matrix R. We shall also prove that $\{\Omega(R_k)\}$ is monotonically decreasing over the sequence $\{R_k\}$ generated by Algorithm 1 and that

$$||x_{k+2} - x^0||/||x_k - x^0|| \leqq \beta(\Omega(R_k) - \Omega(R_{k+1}))^{1/2}, \tag{9}$$

where X^0 is the solution point and β a positive scalar. This will imply that unless the minimum is reached in a finite number of steps, $\{\Omega(R_k)\}$ is a strictly monotonically decreasing sequence. However, since it is bounded below, the right-hand side in (9) goes to zero, implying the superlinear convergence of Algorithm 1.

PROPOSITION 1.

$$n^2 \leqq \Omega(R) \leqq 4\kappa(R) n^2/[1 + \kappa(R)]^2 \tag{10}$$

for any $n \times n$ positive definite matrix R, where $\kappa(R) = ||R||\,||R^{-1}||$, is the condition number of R.

Proof.

$$\Omega(R) = \operatorname{Tr}(R)\operatorname{Tr}(R^{-1}) = \left(\sum_{i=1}^{n}\lambda_i\right)\left(\sum_{i=1}^{n}\frac{1}{\lambda_i}\right), \tag{11}$$

where λ_i are the eigenvalues of R. Defining the n-dimensional vector $r = (1, 1, 1, \cdots, 1)$, we can write

$$\Omega(R) = \frac{(r'\Lambda r)(r'\Lambda^{-1}r)}{(r'r)^2}\, n^2 \tag{12}$$

where Λ is a diagonal matrix whose elements are λ_i. The lower bound in (10) follows from (12) and the Cauchy Schwarz inequality while the upper bound follows from (12) and the Kantorovich [15] inequality.

The properties of the expressions $\mathfrak{D}^{\theta}(R, \gamma, z, z)$ and $\gamma(\varphi)$ were extensively investigated in [18], [20] and [21]. Proposition 2 summarizes without proof some of these properties that will be used in this section.

PROPOSITION 2. *For any $n \times n$ positive definite matrix R and nonzero vector $z \in E^n$, there holds*

$$1.\quad [\mathfrak{D}^0(R, \gamma, z, z)]^{-1} = \mathfrak{D}^1(R^{-1}, 1/\gamma, z, z) \quad \forall\ \gamma > 0. \tag{13}$$

$$\begin{aligned} 2.\quad \mathfrak{D}^{\theta}(R, \gamma, z, z) &= \mathfrak{D}^0(R, \gamma, z, z) + \theta\gamma uu' \\ &= \mathfrak{D}^1(R, \gamma, z, z) - \gamma(1-\theta)uu' \quad \forall\ \gamma > 0 \text{ and } \theta \in [0, 1], \end{aligned} \tag{14}$$

where

$$u = (z'Rz)^{1/2}(z/z'z - Rz/z'Rz). \tag{15}$$

3. $\mathfrak{D}^{\theta}(R, \gamma, z, z)$ *is positive definite for any positive* γ *and* θ.

$$4.\quad 0 \leqq \gamma(0) \leqq \gamma(\varphi) \leqq \gamma(1) \quad \forall\ \varphi \in [0, 1]. \tag{16}$$

$$5.\quad \kappa(\mathfrak{D}^{\theta}(R, \gamma(\varphi), z, z)) \leqq \kappa(R) \quad \forall\ \theta, \varphi \in [0, 1]. \tag{17}$$

PROPOSITION 3. *For any $n \times n$ positive definite matrix R, nonzero vector $z \in E^n$, $\gamma > 0$, and $\theta \in [0, 1]$, there holds*

$$\Omega(\mathfrak{D}^{\theta}(R, \gamma, z, z)) \leqq \operatorname{Tr}(\mathfrak{D}^1(R, \gamma, z, z))\operatorname{Tr}(\mathfrak{D}^1(R^{-1}, 1/\gamma, z, z)). \tag{18}$$

Given an $n \times n$ nonsingular symmetric matrix A, a scalar α, and an n-vector a, the rank-one modification rule (see e.g. [13]) implies that

$$(A + \alpha aa')^{-1} = A^{-1} - \frac{\alpha}{1 + \alpha a'A^{-1}a}\, A^{-1}aa'A^{-1}. \tag{19}$$

It is also well known that

$$\operatorname{Tr}(A + \alpha aa') = \operatorname{Tr}(A) + \alpha a'a. \tag{20}$$

From (14) and (20) it follows that

$$\operatorname{Tr}(\mathfrak{D}^{\theta}(R, \gamma, z, z)) \leqq \operatorname{Tr}(\mathfrak{D}^1(R, \gamma, z, z)) \tag{21}$$

and by (13), (14), (19), and (20),

$$\begin{aligned} \operatorname{Tr}([\mathfrak{D}^{\theta}(R, \gamma, z, z)]^{-1}) &\leqq \operatorname{Tr}([\mathfrak{D}^0(R, \gamma, z, z)]^{-1}) \\ &= \operatorname{Tr}(\mathfrak{D}^1(R^{-1}, 1/\gamma, z, z)). \end{aligned} \tag{22}$$

(18) follows directly from (21) and (22).

PROPOSITION 4. *For any $n \times n$ positive definite matrix R, nonzero vector $z \in E^n$, $\theta \in [0, 1]$, and $\varphi[0, 1]$, there holds*

$$\Omega(R) - \Omega(\mathfrak{D}^{\theta}(R, \gamma(\varphi), z, z)) \geqq u'R^{-1}u \geqq 0 \tag{23}$$

where u is defined by (15).

Proof. By definition

$$\mathfrak{D}^1(R, \gamma(\varphi), z, z) = \left(R - \frac{Rzz'R}{z'Rz} + uu'\right)\gamma(\varphi) + \frac{zz'}{z'z} \tag{24}$$

where u is defined by (15).

Also

$$u'u = z'Rz\left(\frac{z'R^2z}{(z'Rz)^2} - \frac{1}{z'z}\right) = \frac{z'R^2z}{z'Rz} - \frac{z'Rz}{z'z}. \tag{25}$$

Applying (20) repeatedly to (24) and then using (25) yields

$$\operatorname{Tr}(\mathfrak{D}^1(R, \gamma(\varphi), z, z)) = (\operatorname{Tr}(R) - z'Rz/z'z)\gamma(\varphi) + 1 \tag{26}$$

and consequently

$$\operatorname{Tr}(\mathfrak{D}^1(R^{-1}, 1/\gamma(\varphi), z, z)) = \left(\operatorname{Tr}(R^{-1}) - \frac{z'R^{-1}z}{z'z}\right)\frac{1}{\gamma(\varphi)} + 1. \tag{27}$$

From (26) and (27) and Proposition 3 it follows that

$$\begin{aligned}\Omega(\mathfrak{D}^{\theta}(R, \gamma(\varphi), z, z)) &\leqq \left[\operatorname{Tr}(R) - \frac{z'Rz}{z'z} + \frac{1}{\gamma(\varphi)}\right]\left[\operatorname{Tr}(R^{-1}) - \frac{z'R^{-1}z}{z'z} + \gamma(\varphi)\right] \\ &= \operatorname{Tr}(R)\operatorname{Tr}(R^{-1}) - \operatorname{Tr}(R)\frac{z'R^{-1}z}{z'z} - \operatorname{Tr}(R^{-1})\frac{z'Rz}{z'z} + \frac{(z'Rz)(z'R^{-1}z)}{(z'z)^2} + 1 \\ &\quad + \gamma(\varphi)\left[\operatorname{Tr}(R) - \frac{z'Rz}{z'z}\right] + \frac{1}{\gamma(\varphi)}\left[\operatorname{Tr}(R^{-1}) - \frac{z'R^{-1}z}{z'z}\right].\end{aligned} \tag{28}$$

The last two terms in (28) are clearly positive by virtue of the positive definiteness of R. Thus, it is possible to obtain an upper bound on the right-hand side of (28) by replacing $\gamma(\varphi)$ and $1/\gamma(\varphi)$ in the last two terms with their respective upper bounds $(z'R^{-1}z/z'z)$ and $(z'Rz/z'z)$. This yields

$$\Omega(\mathfrak{D}^{\theta}(R, \gamma(\varphi), z, z)) \leqq \Omega(R) + 1 - \frac{(z'Rz)(z'R^{-1}z)}{(z'z)^2} \tag{29}$$

and from (15) we obtain

$$u'R^{-1}u = (z'Rz)\left(\frac{z'}{z'z} - \frac{z'R}{z'Rz}\right)\left(\frac{R^{-1}z}{z'z} - \frac{z}{z'Rz}\right) = \frac{(z'R^{-1}z)(z'Rz)}{(z'z)^2} - 1. \tag{30}$$

(25) results from substituting (30) into (29).

COROLLARY 1. *Let $\{D_k\}$ and $\{p_k\}$ be the sequences of inverse Hessian approximations and updating vectors generated by Algorithm* 1 *when applied to a positive definite quadratic function with Hessian H. If we define $R_k = H^{1/2}D_kH^{1/2}$ and $z_k = H^{1/2}p_k$, then the following properties hold for all k:*

1. R_k *is positive definite.*

(31) 2. $n^2 \leqq \Omega(R_{k+1}) \leqq \Omega(R_k)$.

(32) 3. $\Omega(R_k) - \Omega(R_{k+1}) \geqq u_k' R_k^{-1} u_k \geqq 0$, *where*

$$u_k = (z_k' R_k z_k)^{1/2}\left(\frac{z_k}{z_k' z_k} - \frac{R_k z_k}{z_k' R_k z_k}\right).$$

(33) 4. $\kappa(R_{k+1}) \leqq \kappa(R_k) \leqq \kappa(R_0)$.

Proof. Since D_0 is positive definite and φ_k, $\theta_k \in [0, 1]$, it follows by Proposition 2 that R is positive definite and, by induction, it is true for all R_k.

The remaining properties follow from Propositions 2, 3, and 4.

THEOREM 1. *For a positive definite quadratic function, Algorithm* 1 *converges to the minimum either in a finite number of steps or weak superlinearly.*

Proof. First we show that

(34) $$\frac{||x_{k+2} - x^0||}{||x_k - x^0||} \leqq (\kappa(H)[\kappa(R_0) + 1][\kappa(R_0)]^3[\Omega(R_k) - \Omega(R_{k+1})])^{1/2}$$

where x^0 denotes the minimum point. Again we shall omit the subscripts and use superscripts (*) and (**) for $k + 1$ and $k + 2$. By Algorithm 1,

(35) $$q^* = Hp^* = -HD^*g^* = g^{**} - g^*.$$

Also

(36) $$g^* = q + g = (H - D^{-1})p.$$

By (35) and (36),

(37) $$g^{**} = (I - HD^*)g^* = (I - HD^*)(H - D^{-1})p.$$

Expressing (37) in terms of R and z yields

(38) $$H^{-1/2}g^{**} = (I - R^*)(I - R^{-1})z = z - R^*z - R^{-1}z + R^*R^{-1}z.$$

By (15),

(39) $$u'z = 0.$$

Thus, from (5), (3), (15), and (39), we have

(40) $$z - R^*z = z - z = 0$$

and we also obtain

(41) $$R^*R^{-1}z = \left(z - \frac{Rz(z'z)}{z'Rz} + \theta uu'R^{-1}z\right)\gamma(\varphi) + \frac{z(z'R^{-1}z)}{z'z}.$$

Again, by (15),

(42) $$u'R^{-1}z = (z'Rz)^{1/2}\left(\frac{z'R^{-1}z}{z'z} - \frac{z'z}{z'Rz}\right),$$

(43) $$z - \frac{Rz(z'z)}{z'Rz} = \frac{z'z}{(z'Rz)^{1/2}}\,u$$

and

$$R^{-1}z = \frac{z'z}{(z'Rz)}\,z + \frac{z'z}{(z'Rz)^{1/2}}\,R^{-1}u. \tag{44}$$

Substituting (40), (41), (42), (43) and (44) into (38) yields

$$\begin{aligned} H^{-1/2}g^{**} &= \gamma(\varphi)\left[\frac{z'z}{(z'Rz)^{1/2}} + \theta(z'Rz)^{1/2}\left(\frac{z'R^{-1}z}{z'z} - \frac{z'z}{z'Rz}\right)\right]u \\ &\quad + \left[\frac{z'R^{-1}z}{z'z} - \frac{z'z}{z'Rz}\right]z - \frac{z'z}{(z'Rz)^{1/2}}\,R^{-1}u. \end{aligned} \tag{45}$$

From (45) we obtain, after using (39) and (42),

$$\begin{aligned} g^{**\prime}H^{-1}g^{**} &= [\gamma(\varphi)]^2(z'Rz)\left[\frac{z'z}{z'Rz} + \theta\left(\frac{z'R^{-1}z}{z'z} - \frac{z'z}{z'Rz}\right)\right]^2(u'u) \\ &\quad - 2\gamma(\varphi)\left[\frac{(z'z)^2}{z'Rz} + \theta\left(z'R^{-1}z - \frac{(z'z)^2}{z'Rz}\right)\right](u'R^{-1}u) \\ &\quad - \left[\frac{z'R^{-1}z}{z'z} - \frac{z'z}{z'Rz}\right]^2(z'z) + \frac{(z'z)^2}{z'Rz}\,(u'R^{-2}u). \end{aligned} \tag{46}$$

By the Cauchy-Schwarz inequality,

$$z'R^{-1}z/z'z \geqq z'z/z'Rz, \tag{47}$$

from which it follows that the second and third terms in (46) are nonpositive. Thus, the right-hand side of (46) can be bounded above by dropping the second and third terms and replacing $\gamma(\varphi)$ and θ in the first term with their respective upper bounds, $(z'R^{-1}z/z'z)$ and 1. This yields

$$g^{**\prime}H^{-1}g^{**} \leqq \left(\frac{z'R^{-1}z}{z'z}\right)^4(z'Rz)u'u + \left(\frac{z'z}{z'Rz}\right)^2(z'Rz)(u'R^{-2}u). \tag{48}$$

From the definitions of z and R, we obtain

$$z'Rz = g'DH^{1/2}RH^{1/2}Dg \leqq ||R||^3\,(g'H^{-1}g), \tag{49}$$

also

$$u'u \leqq ||R||\,(u'R^{-1}u), \tag{50}$$

$$u'R^{-2}u \leqq ||R^{-1}||\,(u'R^{-1}u), \tag{51}$$

and

$$(z'z)/(z'Rz) \leqq (z'R^{-1}z)/(z'z) \leqq ||R^{-1}||. \tag{52}$$

Substituting (49) through (52) into (48) yields

$$g^{**\prime}H^{-1}g^{**}/g'H^{-1}g \leqq ([\kappa(R)]^4 + [\kappa(R)]^3)(u'R^{-1}u). \tag{53}$$

However

$$g^{**} = H(x_{k+2} - x^0) \quad \text{and} \quad g = H(x_k - x^0)$$

so that

$$(54)\qquad \frac{g^{**\prime}H^{-1}g^{**}}{g'H^{-1}g} = \frac{(x_{k+2}-x^0)'H(x_{k+2}-x^0)}{(x_k-x^0)'H(x_k-x^0)} \geqq \frac{1}{\kappa(H)}\frac{||x_{k+2}-x^0||^2}{||x_k-x^0||^2}.$$

By (53) and (33) it then follows that

$$(55)\qquad \frac{||x_{k+2}-x^0||}{||x_k-x^0||} \leqq (\kappa(H)[\kappa(R_0)]^3[\kappa(R_0)+1](u_k'R_k^{-1}u_k))^{1/2}.$$

(34) follows immediately from (55) and (32). It is clear from (34) that if, for some k, $\Omega(R_k) = \Omega(R_{k+1})$, then $x_{k+2} = x^0$, i.e., the algorithm converges in a finite number of steps. If this does not happen, then $\{\Omega(R_k)\}$ is a strictly monotonic decreasing sequence bounded below by n^2; thus $\{\Omega(R_k)\}$ converges to a limit point and hence

$$(56)\qquad \lim_{k\to\infty}(\Omega(R_k)-\Omega(R_{k+1})) = 0.$$

By (34) and (56) it then follows that

$$(57)\qquad \lim_{k\to\infty}||x_{k+2}-x^0||/||x_k-x^0|| = 0,$$

which implies weak superlinear convergence.

5. Implementation and Experiments. As mentioned earlier, an implementation of an algorithm based on Algorithm 1 that will work on a general nonquadratic function has to include additional features that will guarantee its global convergence. To ensure positive definiteness of the inverse Hessian approximation, we test at each step whether $p'q > 0$. This condition can always be satisfied by taking a step large enough to cross the minimum along the line. To guarantee convergence to a stationary point, we used the Goldstein test [11] which can be written in the form

$$(58)\qquad \sigma < [f(x^*)-f(x)]/g'p < 1-\sigma$$

with $0 < \sigma < \frac{1}{2}$.

Testing the sensitivity of an algorithm to the parameters φ and θ was out of the scope of this paper, and we confined our experiments to the case $\theta_k = \varphi_k = 0$ for all k. This yields an updating formula of the form

$$(59)\qquad D^* = \left(D - \frac{Dqq'D}{q'Dq}\right)\frac{p'q}{q'Dq} + \frac{pp'}{p'q},$$

which was also used in [18], [19], and [21] to test SSVM algorithms with line search.

The algorithm used in this paper was constructed so that a new point is obtained using $\alpha = 1$; however, if this point does not satisfy condition (58), then α is evaluated by a line search based on cubic interpolation and condition (58) is replaced by other tests within the search subroutine. This strategy proves to be much more effective than trying to modify α according to some rule of thumb. The crucial point in such an algorithm is the choice of σ which actually determines the frequency of line searches. If $\sigma = \frac{1}{2}$, for instance, there will be a line search at each step, while if σ is taken very small (0 to 0.01), almost all the iterations will be without line search. The best choice of σ clearly depends on the nature of the function being minimized. For quadratic functions, a small σ was satisfactory, while for Rosenbrock's and Wood's functions a larger σ was necessary. In general, $0.25 > \sigma > 0.15$ turned out to be a good choice

for all the functions tested, and within this range the performance of the algorithm was quite insensitive to changes in σ.

An Algol W program of the algorithm described above was tested for several test functions with different values of σ. The experiments were performed on an IBM 360/65 computer in single precision. Table 1 summarizes the results of these experiments. For comparison, we also include results from [18] obtained for the DFP and the SSVM algorithms (with line search) and some of the results given by Fletcher [9] for his algorithm.

Test Functions.

(a) Poorly Scaled Quadratic Function—6 variables.
$f(x) = \frac{1}{2}(x'Qx)$.
Q = diag (300, 280, 260, 240, 220, 200).
Minimum point: $x^* = (0, 0, 0, 0, 0, 0)$, $f(x^*) = 0$.
Starting point: $x_0 = (1, 1, 1, 1, 1, 1)$.

(b) Hilbert-Matrix Quadratic Function—N variables.

$$f(x) = \sum_{i=1}^{N} \sum_{j=1}^{N} [(x_j - 1)(x_i - 1)/(j + i - 1)].$$

Minimum point: $x^* = (1, 1, 1, \cdots, 1)$, $f(x^*) = 0$.
Starting point: $x_{0k} = -4/k$.

(c) Helical Valley [8]—3 variables.

$$f(x_1, x_2, x_3) = 100\{[x_3 - 10\theta(x_1, x_2)]^2 + [r(x_1, x_2) - 1]^2\} + x_3^2 .$$

$$\theta(x_1, x_2) = \begin{cases} \dfrac{1}{2\pi} \arctan (x_2/x_1) & \text{for } x_1 > 0, \\ \dfrac{1}{2} + \dfrac{1}{2\pi} \arctan (x_2/x_1) & \text{for } x_1 < 0. \end{cases}$$

$$r(x_1, x_2) = (x_1^2 + x_2^2)^{1/2}.$$

Minumum point: $x^* = (1, 0, 0)$, $f(x^*) = 0$.
Starting point: $x_0 = (-1, 0, 0)$.

(d) Wood's Function [4]—4 variables.

$$\begin{aligned} f(x_1, x_2, x_3, x_4) = {} & 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3)^2 \\ & + (1 - x_3)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] \\ & + 19.8(x_2 - 1)(x_4 - 1). \end{aligned}$$

Minimum point: $x^* = (1, 1, 1, 1)$, $f(x^*) = 0$.
Starting point: $x_0 = (-3, -1, -3, -1)$.

(e) Rosenbrock's Function [24]—2 variables.
$f(x_1, x_2) = 100\,(x_2 - x_1^2)^2 + (1 - x_1)^2$.
Minimum point: $x^* = (1, 1)$, $f(x^*) = 0$.
Starting point: $x_0 = (-1.2, 1)$.

(f) Squared Quadratic—N variables.
$f(x) = (x'Qx)^2$.
$Q \equiv$ diagonal matrix with diagonal elements $q_{kk} = k$ for $k = 1, 2, \cdots, N$.

Minimum point: $x^* = (0, 0, \cdots, 0)$, $f(x^*) = 0$.

Starting point: $x_0 = (1, 1, 1, \cdots, 1)$.

Comments. The squared quadratic function is a typical case that illustrates the advantage of self-scaling algorithms. In this case, the eigenvalues of the Hessian,

TABLE 1

Function	N	ACC(2)	NEW METHOD				SSVM (with line search)		DFP		FLETCHER'S [9] METHOD (1)	
			σ	IT	NF	L.S	IT	NF	IT	NF	IT	NF
(a)	6	10^{-10}	0	6	8	1	6	14	7	15	-	-
(b)	6	10^{-9}	0.01	28(4)	30	1	7	29	7	29	-	-
			0.15	17	42	10						
			0.2	7	29	7						
			0.25	7	29	7						
(c)	3	10^{-9}	0.15	53	84	14	27	69	23	52	-	-
			0.2	26	54	13						
			0.25	23	51	12						
(d)	4	10^{-9}	0.05	98	138	15	60	183	31(3)	93	122	136
			0.15	24	47	7						
			0.2	26	56	11						
			0.25	68	173	51						
(e)	2	10^{-10}	0.1	FAILED			33	137	24	74	39	47
			0.2	35	104	26						
			0.25	29	85	22						
(f)	6	10^{-9}	0.01	19	20	0	9	36	29(3)	88	-	-
			0.2	19	20	0						
			0.25	15	20	2						
	10	10^{-9}	0.01	19	20	0	12	46	42(3)	136	-	-
			0.15	17	21	2						
			0.2	17	21	2						
			0.25	12	26	5						
	20	10^{-9}	0.01	22	26	1	16	56	78(3)	236	-	-
			0.25	21	28	2						
	30	10^{-9}	0.01	25	30	1	20	60	89(3)	270	-	-
			0.25	23	36	3						
	50	10^{-9}	0.01	31	37	1	28	86	134(3)	381	-	-

N—Number of variables.

ACC—Accuracy required in $f(x^*)$.

IT—Number of iterations until $f(x) - f(x^*) \leqq$ ACC.

NF—Number of function evaluations until $f(x) - f(x^*) \leqq$ ACC.

L.S.—Number of iterations in which line search was performed.

(1)—Data taken from original paper (the numbers in the ACC column do not correspond to this data).

(2)—The accuracy required is different for various functions since the actual stopping rule in the algorithms was based on the norm of the gradient.

(3)—Accuracy obtained in $f(x^*)$ was 10^{-8}.

(4)—Accuracy obtained in $f(x^*)$ was only 10^{-6}.

and consequently of R, drift constantly toward zero and since the DFP formula always guarantees a unit eigenvalue for R, the condition number is constantly increasing, which causes the poor behavior of the DFP algorithm in this case. The SSVM formula prevents this difficulty by rescaling the D matrix and (therefore the R) at each updating.

The success of the new method with Wood's function appears to be somewhat artificial since it performed much better than we would expect on the basis of its overall performance. This unusual behavior can be attributed to the nature of the function which has a "trap" in which most algorthims are usually caught. This trap was apparently avoided by our method.

The low accuracy achieved with the new algorithm in minimizing the quadratic function based on the Hilbert matrix was caused by the excessive round-off error resulting from the poor conditioning of the Hilbert matrix. As mentioned earlier, the self-scaling updating formulae reduce the condition number of $H^{1/2}DH^{1/2}$ monotonically. This property is advantageous from a convergence rate viewpoint; however, if H is ill-conditioned, it will cause D to become ill-conditioned also, resulting in excessive round-off error. This difficulty can be overcome by an alternative implementation of the above algorithm in which the Cholesky decomposition of the Hessian approximation is updated at each step and the direction of search is obtained by solving a system of linear equations. This method, which was originally developed by Gill and Murray [10] for the DFP, and Broyden's algorithms will allow control of the conditioning of D fairly easily, and thus reduce the round-off errors.

6. Conclusion. We presented a class of idealized quasi-Newton algorithms in which the inverse Hessian approximation is updated by a two-parameter family of positive definition formulae and no line search is necessary. It was proved that in a quadratic case this algorithm converges in a finite number of steps or superlinearly. This result is much stronger than the one given by Fletcher [9] for a similar algorithm. The above result provided the motivation for implementing an algorithm with economic line searching based on the family of updating formulae mentioned above. Numerical experiments indicate a strong advantage for the proposed algorithms. The experiments in this paper were restricted to a particular updating formula that corresponds to the case where the parameters are both zero. It still remains to investigate the effect of varying these parameters. It is expected that such changes may have a significant effect, particularly since Dixon's [7] result does not apply in general to this class of formulae. Unfortunately, the theory available so far does not provide a basis for predicting such effects.

We expect that further improvement of the new methods, particularly for ill-conditioned problems, will be achieved by using an alternative implementation based on Gill's and Murray's [10] approach. Yet this remains to be experimentally verified.

Xerox Palo Alto Research Center
Palo Alto, California 94304

Department of Engineering-Economic Systems
Stanford University
Stanford, California 94304

→ Y. BARD, "On a numerical instability of Davidon-like methods," *Math. Comp.*, v. 22, 1968, pp. 665–666. MR **38** #858.

2. C. G. BROYDEN, "Quasi-Newton methods and their applications to function minimization," *Math. Comp.*, v. 21, 1967, pp. 368–381. MR **36** #7317.

3. C. G. BROYDEN, "The convergence of a class of double-rank minimization algorithms 2: The new algorithm," *J. Inst. Math. Appl.*, v. 6, 1970, pp. 222–231.

4. A. R. COLVILLE, *A Comparative Study on Nonlinear Programming Codes*, IBM Tech. Rep. No. 320-2949, 1968.

5. W. C. DAVIDON, *Variable Metric Method for Minimization*, A.E.C. Research and Development Rep. ANL-5990 (Rev.), 1959.

6. W. C. DAVIDON, "Variance algorithm for minimization," *Comput. J.*, v. 10, 1968, pp. 406–410. MR **36** #4790.

7. L. C. W. DIXON, "Variable metric algorithms: Necessary and sufficient conditions for identical behavior of non-quadratic functions," *J. Optimization Theory Appl.*, v. 10, 1972, pp. 34–40.

8. R. FLETCHER & M. J. D. POWELL, "A rapidly convergent descent method for minimization," *Comput. J.*, v. 6, 1964, pp. 163–168. MR **27** #2096.

→ R. FLETCHER, "A new approach to variable metric algorithms," *Comput. J.*, v. 13, 1970, pp. 317–322.

→ P. E. GILL & W. MURRAY, "Quasi-Newton methods for unconstrained optimization," *J. Inst. Math. Appl.*, v. 9, 1972, pp. 91–108.

11. A. A. GOLDSTEIN, "On steepest descent," *J. Soc. Indust. Appl. Math. Ser. A Control*, v. 3, 1965, pp. 147–151. MR **32** #2249.

12. J. GREENSTADT, "Variations on variable metric methods," *Math. Comp.*, v. 24, 1970, pp. 1–22. MR **41** #2895.

13. A. S. HOUSEHOLDER, *The Theory of Matrices in Numerical Analysis*, Blaisdell, New York, 1964. MR **30** #5475.

14. H. Y. HUANG, "Unified approach to quadratically convergent algorithms for function minimization," *J. Optimization Theory Appl.*, v. 5, 1970, pp. 405–423. MR **44** #6134.

15. L. V. KANTOROVIČ & G. P. AKILOV, *Functional Analysis in Normed Spaces*, Fizmatgiz, Moscow, 1959; English transl., Internat. Series of Monographs in Pure and Appl. Math., vol. 46, Macmillan, New York, 1964. MR **22** #9837; MR **35** #4699.

16. D. G. LUENBERGER, *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, Reading, Mass., 1973.

17. B. A. MURTAGH & R. W. H. SARGENT, *A Constrained Minimization Method With Quadratic Convergence*, Optimization (Sympos., Univ. Keele, 1968), Academic Press, London, 1969, pp. 215–245. MR **44** #1442.

18. S. S. OREN, *Self-Scaling Variable Metric Algorithms for Unconstrained Minimization*, Ph.D. Thesis, Department of Engineering-Economic Systems, Stanford University, Stanford, Calif., 1972.

19. S. S. OREN & D. G. LUENBERGER, *The Self-Scaling Variable Metric Algorithm*, Proc. Fifth Hawaii International Conference on System Sciences, January, 1972.

20. S. S. OREN & D. G. LUENBERGER, "Self-scaling variable metric (SSVM) algorithms. I: Criteria and sufficient conditions for scaling a class of algorithms," *Management Sci.* (To appear.)

21. S. S. OREN, "Self-scaling variable metric (SSVM) algorithms. II: Implementation and experiments," *Management Sci.* (To appear.)

22. M. J. D. POWELL, "Rank one methods for unconstrained optimization," in *Integer and Nonlinear Programming*, J. Abadie (Editor), North-Holland, Amsterdam, 1970, pp. 139–156.

23. M. J. D. POWELL, *Recent Advances in Unconstrained Optimization*, Report No. T. P., 430, A. E. R. E., Harwell, 1970.

24. H. H. ROSENBROCK, "An automatic method for finding the greatest or least value of a function," *Comput. J.*, v. 3, 1961, pp. 175–184. MR **24** #B2081.