A BRIEF INTRODUCTION[†]
TO QUASI-NEWTON METHODS[†]

by

J. E. Dennis Jr.*

TR77-327

Department of Computer Science
Cornell University
Ithaca, NY 14853

[†] Prepared for AMS Shortcourse on Numerical Analysis, Atlanta,
Jan. 1978.

* Research supported by NSF Grant MCS76-0032

A BRIEF INTRODUCTION[†]
TO QUASI-NEWTON METHODS[†]

by

J. E. Dennis Jr.*

Cornell University
Ithaca, NY 14853

0. Introduction

This section was written last as most introductions are and so it also functions as a disclaimer for omissions, misconceptions and misrepresentations.

We try below to present a current, hurried, view of a field in flux, because of progress and changing standards. In this field theorems serve to document progress and point promising future routes. Our purpose is to solve problems and our goal is to develop better algorithms certified by realistic theorems and implemented in robust software. We are beginning to see young people coming into the field who have the breadth to do all these things but there is still room for specialized contributions.

1. The Problem

The problems considered here might be looked at by a novice to computation as being really only one problem since they are mathematically equivalent in a fairly meaningful sense. From our algorithmic point of view, there is a significant gain in exploitable structure as we pass from the problem of simultan-

[†] Prepared for AMS Shortcourse on Numerical Analysis, Atlanta, Jan. 1978.

* Research supported by NSF Grant MCS76-00324.

eous nonlinear equations to nonlinear least squares via unconstrained minimization. We mean by this that certain features of the problem lend themselves to more efficient computer implementations of the class of algorithms outlined in section 2.

The first problem we will consider is certainly an instance of one of the most basic in mathematics.

Let Ω be an open convex set in $\mathbb{R}^n$.

NLEQ: Given $F : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ find x^* for which $F(x^*) = 0$.

We will use the notation T for transpose and for $x \in \mathbb{R}^n$, $x = (x^1, x^2, \dots, x^n)^T$ and $F(x) = (f^1(x), f^2(x), \dots, f^n(x))^T$. For derivatives, we write $F'(x) = J(x) = (\partial f^i(x)/\partial x^j)$ and for gradients $\nabla f(x) = (\partial f(x)/\partial x^1, \dots, \partial f(x)/\partial x^n)^T$, where $f(x)$ is a real-valued function of the vector variable x . Unless otherwise specified, the Jacobian matrix $J(x)$ will be assumed to be of full rank at x^* and Lipschitz continuous in Ω .

The second problem has two additional algorithmically useful features.

UCMIN: Given $f : \Omega \rightarrow \mathbb{R}^1$, find $x^* \in \Omega$, a local minimizer of f , i.e., for some $\epsilon > 0$, $x \in N(x^*, \epsilon)$ implies $f(x) \geq f(x^*)$.

We will always assume f has a Lipschitz continuous second derivative and use the notation $f''(x) = \nabla^2 f(x) = (\partial^2 f(x)/\partial x^i \partial x^j)$. Notice that we will find x^* , if it exists, among the zeros of the system of nonlinear equations $\nabla f(x) = 0$. It will be useful that the Jacobian of this system is symmetric and is positive definite near a solution for most practical problems. We will assume that $\nabla^2 f(x)$, the Hessian, is symmetric and positive definite at x^* , as well as Lipschitz continuous in Ω .

Another very important point is the utility of f itself as a means of choosing between approximate solutions. In algorithms for NLEQ, it is often the case that some norm of F is used in this manner but this leads to the danger of finding a local minimizer of $||F(x)||$ rather than a zero. When the l_2 norm is used, this is an important practical problem in its own right.

NL2: Given $R : \Omega \subset \mathbb{R}^p \rightarrow \mathbb{R}^n$, $p < n$, minimize

$$\phi(x) = \frac{1}{2} R(x)^T R(x) = \frac{1}{2} \sum_{i=1}^n (r^i(x))^2.$$

This nonlinear least squares problem arises frequently in practice, especially in the context of best fitting n data points from a family of functions depending on p parameters. Furthermore, there are two quite different cases. Sometimes the final fit is very good in the sense that $R(x^*)$ is nearly zero. On the other hand, in many real problems data errors preclude small final residuals $R(x^*)$. It turns out to be useful to know which to expect since the small residual case can be handled with a much simpler algorithm if it is expected. We can indicate why this is so and establish some more notation at the same time.

It is easy to see that with $J(x) = R'(x)$, $\nabla \phi(x) = J(x)^T R(x)$ and $\nabla^2 \phi(x) = J(x)^T J(x) + \sum_{i=1}^n r^i(x) \nabla^2 r^i(x)$. Thus, when $J(x)$ is computed or approximated for the gradient evaluation, it is available to form the first term in the expression given above for the Hessian. In the small residual case, this term suffices for the algorithms we will consider and the savings is considerable. The gradient also has useful structure which can be exploited

in the nonzero residual case to yield a scale-free solution test. Much more fuss is made of all this in [25] where an introduction is also given to the interesting idea of exploiting linearity in some of the parameters. See [42], [45], [47], [58], [66], [75].

In practice, these three problems are sometimes posed with x restricted to lie not in an open set Ω but in some set Γ defined by inequality and equality constraints. Often this causes the significant complications which arise when x^* is on the boundary of Γ . Recently, there has been very rapid development in this area but since all that work is based on techniques for the corresponding unconstrained problem and since [34], [65] and [72] are such excellent current surveys we do not consider constrained problems here.

2. The One Basic Algorithm

The algorithms in most common usage for all these problems are in virtually all cases variations of Newton's method for NLEQ. This algorithm, also called the Newton-Raphsen method, can be written as:

$$\text{Given } x_0, x_{k+1} = x_k - J(x_k)^{-1}F(x_k), k=1,2,\dots,$$

Notice that we have elected to use subscripts for the iteration counter and superscripts for vector components.

The iteration formula above is certainly not the way Newton's method should be implemented on a computer. The following form

will be useful for our discussion and it is much more like an actual implementation. We will call such algorithms quasi-Newton.

- c) Given x_k , $F(x_k)$ and $J(x_k)$;
- i) Solve the $n \times n$ linear system $J(x_k)s_k^N = -F(x_k)$ for the Newton step s_k^N .
- ii) Using s_k^N and perhaps some other values of $F(x)$, choose x_{k+1} .
- iii) Evaluate $F(x_{k+1})$ and test for convergence and non-convergence. Either terminate the computation or proceed to (iv).
- iv) Evaluate (or approximate) $J(x_{k+1})$, set the counter to $k+1$ and return to (i).

Each of these four steps is still rather vague and the resolution of these vagaries has a profound impact on the performance of the implementation. The solution of the linear system in (i) is dealt with extensively elsewhere and briefly in Appendix A, so we will only make occasional remarks in context about particular advantages of certain algorithms in certain cases. For example, when $J(x_k) = \nabla^2 f(x_k)$ a symmetric indefinite factorization can be used to exploit symmetry and perhaps even a Cholesky factorization can be used when f is convex. It does seem appropriate to make some other remarks here however.

On the face of it, Newton's method requires the Jacobian matrix to be nonsingular at each iterate. When x_k is near x^* , the same concerns about well-posedness of the problem guide us in the case when F is an affine mapping and the same remedies such as discarding some combinations of the variables to make

the problem acceptably well-conditioned seem reasonable. When x_k is not so near x^* , it is not unusual to encounter numerical singularity in $J(x_k)$ even for the problems where $J(x^*)$ is well-conditioned. It seems certainly wrong to discard a subset of the variables because of such temporary difficulties. How to cheaply deal with a numerically singular Jacobian in the early stages of the iteration is a very important practical problem which seems closely tied to the success of some other aspects of the algorithm such as (ii). See [25].

The choice of termination criteria is fraught with peril and again we refer to [25].

The traditional areas of research on quasi-Newton methods are steps (ii) and (iv). The reason is that for real problems, evaluations of F and J dominate the cost of solution and so it is in these steps that the potential savings is greatest.

It is worth remarking that the value of $F(x_{k+1})$ will often be available as a byproduct of the choice of x_{k+1} . However, we put this in step (iii) because $F(x_{k+1})$ must be evaluated anyway to carry out the iteration and so it should not be charged to whatever technique is used in (ii) to pick x_{k+1} . Furthermore, this makes it easier to put the unvarnished Newton method, where $x_{k+1} = x_k + s_k^N$, into our framework. It is absolutely essential that step (ii) allow a full Newton step whenever possible and that step (iv) be carried out in such a way as to enable this to be done once x_k is near x^* .

Much of the remainder of this paper will be concerned with methods for the approximation of $J(x_k)$. My experience is that

$J(x_k)$ is rarely available. Even when it is, there is a need to take and code n^2 partial derivatives. Obviously an expensive and error prone task. We will call these algorithms quasi-Newton to allow for the use of approximate derivatives.

3. The Kantorovich Theorem

One of the most beautiful theorems in numerical analysis certifies the performance of Newton's method under very reasonable hypotheses on the problem. The version given here is slightly nonstandard, it comes from applying a more standard version to $J(x_0)^{-1}F(x)$ rather than $F(x)$. We will explain later why we prefer this version.

Theorem 3.1. Let $r > 0$, $x_0 \in \mathbb{R}^n$, $F : N(x_0, r) \rightarrow \mathbb{R}^n$ and assume that $J(x)$ exists on $N(x_0, r)$ with $J(x_0)$ invertible. Let $\gamma, \eta \geq 0$ exist such that $\|J(x_0)^{-1}[J(x) - J(y)]\| \leq \gamma \|x - y\|$, $\|J(x_0)^{-1}F(x_0)\| \leq \eta$ and $h = \gamma\eta \leq \frac{1}{2}$. Under these hypotheses, if $r \geq r_0 = (1 - \sqrt{1-2h})/\gamma$ then Newton's method starting from x_0 , $x_{k+1} = x_k - J(x_k)^{-1}F(x_k)$, $k=0,1,\dots$ exists and converges to x^* , a unique zero of F in $N(x_0, r_0)$. If $h < \frac{1}{2}$ then x^* is the only zero of F in $N(x_0, r_1)$ where $r_1 = \min(r, (1 + \sqrt{1-2h})/\gamma)$.

Furthermore, for each k , set $t_{k+1} = t_k - (\frac{1}{2}\gamma t_k^2 - t_k + \eta)/(\gamma t_k - 1)$ with $t_0 = 0$, and then $\|x_k - x^*\| \leq r_0 - t_k$.

A proof of this theorem and several similiar ones can be found in [22]. The basic idea is a lovely one. It involves showing that Newton's method for $q(t) = \frac{1}{2}\gamma t^2 - t + \eta$ starting from $t_0 = 0$ majorizes the vector iteration. The invertibility of $J(x_k)$ follows from the Banach Lemma [57]. The existence of

x^* follows from a completeness argument similar to the one used in the Contractive Mapping Theorem. In fact, the connection between these two results is much deeper than one might suspect at first.

There is a tendency to think in terms of applying Theorem 3.1 to a specific F and x_0 to see if F has a zero and if Newton's method starting from x_0 will be guaranteed to find it and obey the error bounds $r_0 - t_k$. In fact, it has been usefully applied this way to integral equations [13].

This author at least, now prefers a different point-of-view. This is because it is almost certainly more difficult to determine a value for γ , the Lipschitz constant, than it is to run the program and see if the iteration converges. Furthermore, a few tests dispel the notion that the error bounds are much good in general although they are exact for the majorizing scalar iteration.

We prefer to think that the theorem identifies a very broad and reasonable class of problems for which the iteration converges. The reason we like the version given here is that the sufficient condition $\gamma \cdot \eta \leq \frac{1}{2}$ can be interpreted as relating convergence to the scale of F , or the quality of the solution estimate x_0 , as measured by η and γ , a relative measure of the local nonlinearity of F . Obviously $\gamma = 0$ for an affine F and Newton's method is globally convergent in such cases. As the problem becomes more nonlinear, a better initial estimate is needed for convergence.

The error bounds are, as we have said, not very good in practice. Furthermore, they only imply r -quadratic (order 2)

convergence rather than the more useful q-quadratic convergence which makes Newton's method as fast as any method ever really need be. There is an extensive discussion of these notions of convergence order in [57] and a brief discussion in terms of step (iv) of the general algorithm of Section 2 in [22].

Basically an iteration has q-order at least q if $\limsup_k e_{k+1}/e_k^q$ is finite, where $e = ||x - x^*||$ and it has r-order r if there is some sequence of error bounds $\langle b_k \rangle$ with q-order r. Obviously a given q-order implies at least the same r-order. In [20] we called r-order "pseudo-order" and showed the two equivalent if and only if $\limsup_k b_k/e_k$ is finite. Still, it is probably better to have an r-order result than none at all.

It is easy to obtain a much more important error estimate: under the hypothesis of Theorem 3.1,

$\frac{1}{2}||x_{k+1} - x_k|| \leq e_k \leq 2||x_{k+1} - x_k||$. See [22] for a numerical comparison with $b_k = r_0 - t_k$.

4. Least Change Secant Methods

In Section 2 we mentioned the necessity for finding good cheap ways to approximate $J(x_k)$. For $n = 1$, the secant method for NLEQ is a very satisfactory way to do this. This classical quasi-Newton method uses $[F(x_{k+1}) - F(x_k)]/[x_{k+1} - x_k] \equiv y_k/s_k$ as its approximation to $J(x_{k+1})$. There is a beautiful economy here since $F(x_{k+1})$ is needed anyway and so no additional function or derivative work is done in step (iv).

4.1. Jacobian Approximations

In higher dimensions there is an affine set $Q(y_k, s_k)$ of $n \times n$ matrices that send s_k to y_k and thus generalize the secant approximation to $J(x_{k+1})$. A great deal of theoretical work has been done [57] on extensions of the secant method to n -dimensions by resolving this ambiguity in such a way as to achieve r -order $\frac{1}{2}(1 + 5^{\frac{1}{2}})$, the q -order of the scalar method. None of these methods has so far proven to be of much use in actually solving particular instances of NLEQ.

We prefer another, more computationally oriented way of selecting the next approximate Jacobian from $Q(y_k, s_k)$. The idea is simple, geometric, and elegant. When we reach step (iv), we have just made a successful step with some approximate Jacobian A_k . It seems reasonable to retain as much of that information contained in A_k as is consistent with the new secant information.

Let $\|M\|_F = (\sum_{i,j} m_{ij}^2)^{\frac{1}{2}}$ denote the Frobenius norm of a matrix.

Theorem 4.1. Let $s \neq 0$ and y be real n -vectors and let $Q(y, s) = \{Q : Q \text{ is an } n \times n \text{ real matrix and } Qs = y\}$. If A is any $n \times n$ matrix then $A_+ = A + \frac{(y - As)s^T}{s^T s}$ is the unique solution to: $\min \|A - Q\|_F$ for $Q \in Q(y, s)$.

This theorem is easy to prove [29] and since $s_k \neq 0$, motivates Broyden's [9] method: $A_{k+1} = A_k + \frac{(y_k - A_k s_k)s_k^T}{s_k^T s_k}$.

Powell [59], [60] published a general NLEQ routine based on a quasi-Newton algorithm with a Broyden secant approximation to

the Jacobians in step (iv). To the best of our knowledge, modified versions of that program have been the best general purpose routines ever since. J. J. Moré of Argonne Labs has a clean simplified version which starts with A_0 gotten from a finite difference or discrete approximation to $J(x_0)$ based on [16].

There is a Kantorovich theorem for Broyden's method in [21] but the following theorem from [12] more accurately validates the excellent performance of Broyden's algorithm.

Theorem 4.2. Let $F : \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ have a zero x^* in Ω and assume that J is Lipschitz continuous on Ω and that $J(x^*)$ is invertible. There exist positive numbers ϵ and δ such that if $\|x_0 - x^*\| < \epsilon$ and $\|A_0 - J(x^*)\| < \delta$ then Broyden's method starting from x_0 and A_0 is defined and converges to x^* .

Furthermore, $\lim_k e_{k+1}/e_k = 0$ and $\lim_k e_k/\|s_k\| = 1$.

The error analysis for this curious method is very interesting. The condition that the quotient of successive error norms go to zero is called "q-superlinear convergence". The error estimate furnished by the norm of the current step is a very trivial consequence of superlinear convergence [28]. A rather surprising fact is that Broyden's method achieves this fast convergence without being consistent i.e., $\langle A_k \rangle$ does not necessarily converge, much less to $J(x^*)$. The key to the behavior of $\langle A_k \rangle$ is the following theorem from [28] which characterizes superlinear convergence.

Theorem 4.3. Let F and x^* satisfy the hypothesis of Theorem 4.2 except that J need only be continuous and not Lipschitz continu-

ous. Let $\langle B_k \rangle$ be any sequence of nonsingular matrices and suppose that for some x_0 the iteration sequence

$$x_{k+1} = x_k - B_k^{-1} F(x_k), \quad k = 0, 1, \dots$$

remains in Ω and $x_k \neq x^*$, for $k \geq 0$. Then $\langle x_k \rangle$ converges q -superlinearly to x^* if and only if

$$\lim_k \frac{[B_k - J(x^*)](x_{k+1} - x_k)}{\|x_{k+1} - x_k\|} = 0.$$

Recently Gay [36] proved the following completely surprising result about the speed of convergence of $\langle x_k \rangle$.

Theorem 4.4. Let F and x^* satisfy the hypothesis of Theorem 4.2. Then in addition to the conclusion of Theorem 4.2, there exists a positive constant λ such that for every $k \geq 0$, $e_{k+2n} \leq \lambda e_k^2$.

4.2 Hessian Approximations

We will return to a consideration of NLEQ in the next section but let us now consider UCMIN. In this case it would seem silly not to use a sequence of symmetric approximations to the Hessian matrices in carrying out a quasi-Newton iteration. This would allow us to save storage, it would be useful in step (i) and, if done with care, would probably give a better approximate Hessian since

$$\left\| \frac{1}{2}(A_k + A_k^T) - \nabla^2 f(x_k) \right\|_F \leq \|A_k - \nabla^2 f(x_k)\|_F.$$

Thus, averaging any approximation A_k with its transpose gives a symmetric Hessian approximation which is guaranteed to be no worse.

Let us suppose then that $A_k \in S \equiv \{M : M \text{ is a real symmetric } n \times n \text{ matrix}\}$ and that this is part of the information we wish to preserve in A_{k+1} from A_k .

Theorem 4.5. Let $A \in S$ and let $x \neq 0$, y be n -vectors. Then

$A_+ = A + \frac{(y - As)s^T + s(y - As)^T}{s^T s} - \frac{s^T(y - As)ss^T}{(s^T s)^2}$ is the unique solution to $\min ||A - Q||_F$ for $Q \in S \cap Q(y, s)$.

This theorem is easy to prove [29], and since $s_k \neq 0$, it motivates the symmetric Broyden method of Powell [61]

$$A_{k+1} = A_k + \frac{(y_k - A_k s_k)s_k^T + s_k(y_k - A_k s_k)^T}{s_k^T s_k} - \frac{s_k^T(y_k - A_k s_k)s_k s_k^T}{(s_k^T s_k)^2}$$

where $A_0 = A_0^T$. Again we refer elsewhere [23] for a Kantorovich Theorem and state the following local result from [12].

Theorem 4.6. Let the hypothesis of Theorem 4.2 hold and assume in addition that $J(x^*)$ is symmetric. If A_0 is chosen to be symmetric, then the conclusions of Theorem 4.2 hold for the symmetric Broyden method.

Near a solution to the UCMIN problem, it is reasonable to assume that F is strictly convex. In addition, there are technical reasons associated with step (ii) why it is useful to maintain positive definite Hessian approximations. Obviously this would be useful in step (i).

We will derive our next two methods by choosing approximations from $Q(y, s) \cap S^+$, where S^+ is the subset of S whose members are positive definite. This time, we will also use the structure of the problem in measuring the change to be made to the current approximation. Any member of S^+ induces an inner product norm on $\mathbb{R}^n$ and the norm $(x^T \nabla^2 f(x^*) x)^{1/2}$ is sometimes referred to as the natural norm induced by UCMIN since level sets of the local quadratic approximation to f are circles in this norm. It would seem reasonable to try to choose A_{k+1} to make the least possible change to A_k measured by $\| \nabla^2 f(x^*)^{-1/2} [A_k - A_{k+1}] \nabla^2 f(x^*)^{-1/2} \|_F$ but, of course, we don't know the Hessian. Even so, the following theorem from [29] says we can almost do what we wish and in practice, this turns out to be sufficient.

Theorem 4.7. Let $A \in S^+$, s and y be n -vectors. If $s^T y > 0$ then $S^+ \cap Q(y, s) \neq \{ \}$. Let $M \in S^+$, then for $v = M^{-2}s$,

$$A_+ = A + \frac{(y - As)v^T + v(y - As)^T}{v^T s} - \frac{s^T(y - As)vv^T}{(v^T s)^2}$$

is the unique solution to

$$\min \|M(A - Q)M\|_F, \quad Q \in Q(y, s) \cap S^+.$$

Now, since the symmetric positive definite matrix $\nabla^2 f(x^*)^{-1}$, has a square root in S^+ , we see that all we would really need to minimize the approximate Hessian change in the natural metric would be $v_k = \nabla^2 f(x^*) s_k$. The Davidon [18]-Fletcher-Powell [35] method uses $v_k = y_k$ which

can also be looked at as taking M^{-2} to be an arbitrary element of $S^+ \cap Q(y, s)$. This was the first method of the type we are considering and it really opened up the whole field. For definiteness we repeat that the Davidon-Fletcher-Powell or DFP method generates a sequence of symmetric positive definite approximate

$$A_{k+1} = A_k + \frac{(y_k - A_k s_k) y_k^T + y_k (y_k - A_k s_k)^T}{y_k^T s_k} - \frac{s_k^T (y_k - A_k s_k) y_k y_k^T}{(y_k^T s_k)^2}.$$

At the beginning of this section we mentioned that in the real secant method, it was possible to compute the approximation to the reciprocal of the derivative directly. Indeed, it is possible with the same work, to update approximations to the inverse of the Jacobian matrices in the Broyden methods we have discussed. The importance of this is that in either Jacobian or inverse Jacobian form, we are now accomplishing step (iv) with $O(n^2)$ arithmetic operations but if we obtain A_{k+1} then $O(n^3)$ operations are needed for step (i) while only $O(n^2)$ would be needed to obtain s_k^N if we maintain $\langle A_k^{-1} \rangle$.

It was common until a few years ago to maintain $\langle A_k^{-1} \rangle$ for just the reasons above, even though we often had troubles, especially in step (ii), because of numerical singularity in the approximate derivatives. Modern routines maintain $\langle A_k \rangle$ but in factored form, so that step (i) can be carried out in $O(n^2)$ operations, but so that there is greater control over the conditioning of $\langle A_k \rangle$. P. E. Gill and W. Murray of NPL in Britain seem to deserve most of the credit for this practice. See [15], [37], [38], [40], [70]. Moré's program updates a QR factorization of

the approximate Jacobians. A recent thesis by Sorensen [70] opens the way for maintaining a symmetric indefinite factorization in the symmetric Broyden method. Following [38] we could maintain an LDL^T decomposition in the DFP method.

Even if we no longer think in terms of generating the sequence $\langle A_k^{-1} \rangle$ it turns out that the currently preferred secant method for the UCMIN problem can be readily seen from Theorem 4.7 (with y and s interchanged) to solve the following minimization problem:

$$\min ||\nabla^2 f(x^*)^{\frac{1}{2}}[A_k^{-1} - Q^{-1}]\nabla^2 f(x^*)^{\frac{1}{2}}||_F \text{ for, } Q \in S^+ \cap Q(y_k, s_k)$$

for $y_k^T s_k > 0$. This method is due to Broyden [10], Fletcher [33], Goldfarb [39] and Shanno [69] and is often called the BFGS or complementary DFP method. It is best implemented [38] in the LDL^T form but best understood in terms of Theorem 4.7 as :

$A_{k+1}^{-1} = A_k^{-1}$ unless $y_k^T s_k > 0$ in which case

$$A_{k+1}^{-1} = A_k^{-1} + \frac{(s_k - A_k^{-1} y_k) s_k^T + s_k (s_k - A_k^{-1} y_k)^T}{y_k^T s_k} - \frac{y_k^T (s_k - A_k^{-1} y_k) s_k s_k^T}{(y_k^T s_k)^2}.$$

Here is a mystery. Why is the BFGS better in practice than the DFP while the methods due to Broyden [9] and to Greenstadt [43] that solve

$$\min ||A_k^{-1} - Q^{-1}||_F, Q \in Q(y, s)$$

and, for A_k symmetric,

$$\min ||A_k^{-1} - Q^{-1}||_F, Q \in Q(y, s) \cap S$$

are lousy although the Broyden methods are good? Theorems

4.2 and 4.5 hold for these latter two methods by the way [12].

We conclude this section with a theorem from [12].

Theorem 4.8. Let the hypothesis of Theorem 4.2 hold for the UCMIN problem and assume that $J(x^*) = \nabla^2 f(x^*)$ is positive definite as well as symmetric. If A_0 is chosen to be symmetric and positive definite then the conclusions of Theorem 4.2 hold for the DFP and BFGS methods.

5. More Least Change Secant Methods

There are other reasonable choices of M to be made in Theorem 4.7. All the obvious ones yield methods which have been discovered and named. We have only given the ones that seem to be in most common usage. In our opinion, the least change point-of-view makes this group of strange ad-hoc looking methods seem quite reasonable. It certainly is the case that none of them were first discovered in that manner except for those given by Greenstadt [43]. On the other hand, these ideas were at the very heart of the convergence analysis given in [12].

In the next two sections, we would like to take a different tack. The norms used to measure the change from A_k to A_{k+1} will be those we are familiar with, $M = I$ and $M = \nabla^2 f(x^*)^{1/2}$, but we will try to incorporate more structure into $\langle A_k \rangle$ by imposing additional constraints on the set from which A_{k+1} will be chosen.

5.1. Sparsity

It is often the case that $f^i(x)$ in NLEQ will only depend on

a few of the components of x . Thus $J(x)$ will have many entries which will always be zero. If n is large, it may be necessary to use sparse matrix techniques in step (i) in order to make the iteration tractable. If we were to use Broyden's method, then $\langle A_k \rangle$ would be a sequence of full matrices. Schubert [68] and Broyden [11] suggested the following update scheme for which they report good results.

Let $Z = \{A : A \text{ is a real } n \times n \text{ matrix with } a_{ij} = 0 \text{ if } J(x)_{ij} = 0 \text{ for every } x \in \Omega\}$. Let e_j be the j th column of I . Let a^+ be generalized inverse notation in the sense that $0^+ = 0$ and $a^+ = 1/a$ for $a \neq 0$. The last piece of notation we need is that for any n -vector v , $z_i(v)$ will be the n -vector obtained from v by setting $v^j = 0$ for every j such that $a_{ij} = 0$ for every $A \in Z$. Let $A_k \in Z$ and set $A_{k+1} = A_k + \sum_{j=1}^n (||z_j(s_k)||_2^2)^+ e_j e_j^T (y_k - A_k s_k) z_j(s_k)^T$. A_{k+1} is readily shown to be in $Z \cap Q(y_k, s_k)$. Marwil [50] has shown that A_{k+1} is a least change secant method.

Theorem 5.1. Let $s \neq 0$, y be n -vectors and $A \in Z$. Then $A_+ = A + \sum_{j=1}^n (||z_j(s)||_2^2)^+ e_j e_j^T (y - As) z_j(s)^T$ is the unique solution

$$\min ||A - Q||_F \text{ for } Q \in Z \cap Q(y, s).$$

This justifies the name which, with all due respect to Schubert, we suggest: the sparse Broyden method. Marwil [50] proved the following theorem.

Theorem 5.2. Let the hypothesis of Theorem 4.2 hold and let $A_0 \in Z$. Then the conclusions of Theorem 4.2 hold for the sparse Broyden method and $A_k \in Z$ for $k = 1, 2, \dots$.

Neither Schubert nor Broyden used our approach to find the sparse Broyden method. Both Toint [73] and Marwil [50] did find the sparse symmetric Broyden method by solving, for $A_k \in Z$

$$\min \|A_k - Q\|_F \text{ for } Q \in Z \cap S \cap Q(y_k, s_k).$$

The formula is somewhat complicated so we won't give it but Toint [74] reports promising computational results and Marwil [50] has given the local and q-linear convergence result.

An important problem is to find a sparse DFP or BFGS method: Let $M^2 \in Z \cap Q(y, s) \cap S^+$ and $A \in Z \cap S^+$ and solve $\min \|M^{-1}[A - Q]M^{-1}\|_F$, or $\min \|M[A^{-1} - Q^{-1}]M\|_F$ for $Q \in Z \cap S^+ \cap Q(y, x)$.

Can anything be done about maintaining factorizations? This is a very important research question.

5.2 A Partially Available Jacobian

Time and again we have seen NLEQ examples arise in practice in which a significant portion of the Jacobian is available. In other words, there is a natural splitting $J(x) = C(x) + A(x)$ where $C(x)$ can be computed or is amenable to some special approximation rule and $A(x)$ is unavailable. It seems likely that we have the ideas at hand to build special purpose methods to exploit this situation. After all, Section 4 was the special case $C(x) = 0$.

As a broad outline, let us look at the default technique. We think of step (iv) as consisting of two parts. Let

$B_k = C(x_k) + A_k$ be our approximation to $J(x_k)$. First we obtain $C(x_{k+1})$ and then for some $M \in S^+$ (maybe $M = I$) we obtain A_{k+1} as the solution to :

$$\min ||M[A_k - Q]M||_F \text{ for } Q \in Q(y_k - C(x_{k+1}), s_k, s_k)$$

which is the same, for $B_{k+1} = C(x_{k+1}) + A_k$ as obtaining B_{k+1} as the solution to:

$$\min ||M[B_{k+1} - Q]M||_F \text{ for } Q \in Q(y_k, s_k).$$

Of course, one can add further constraints, like symmetry or sparseness.

In order to test these ideas, and to try to solve an important practical problem, we considered the NL2 problem [26]. In that case, $J(x) = \nabla^2 \phi(x)$, $C(x) = J(x)^T J(x)$, and $A(x) = \sum_{i=1}^n r^i(x) \nabla^2 r^i(x)$. We were working in a context where $A(x)$ could usually not be ignored and so it seemed ideal. Others [2], [3] have tried update methods for approximating $A(x)$ but our least change approach seems to perform better. The final algorithm NL2SOL explained in [26] is a DFP method for NL2. We choose

$$A_{k+1} = A_k + \frac{(y_k^\sharp - A_k s_k) y_k^T + y_k (y_k^\sharp - A_k s_k)^T}{y_k^T s_k} - \frac{s_k^T (y_k^\sharp - A_k s_k) y_k y_k^T}{(y_k^T s_k)^2}$$

where $A_k \in S$ and $y_k = \nabla \phi(x_{k+1}) - \nabla \phi(x_k)$ and $y_k^\sharp = J(x_{k+1})^T F(x_{k+1}) - J(x_k)^T F(x_{k+1})$. The matrix A_{k+1} solves, for $M^{-2} \in S^+ \cap Q(y_k, s_k)$,

$$\min ||M[A_k - Q]M||_F \text{ for } Q \in Q(y_k^\sharp, s_k) \cap S.$$

We derived $y_k^\sharp$ from considering $\lambda(x_{k+1}) s_k$ and making the

secant approximations $\nabla^2 r^i(x_{k+1})s_k \approx \nabla r^i(x_{k+1}) - \nabla r^i(x_k)$. This performed better than the default which is to choose

$$A_{k+1} \in Q(y_k - J(x_{k+1})^T J(x_{k+1})s_k, s_k) \cap S.$$

5.3 Not For Every Problem But For Some Others

In order for the techniques of 5.2 to be useful it seems that $C(x)$ must represent a nontrivial portion of $J(x)$. It will probably not be possible to carry along a factored sequence of approximate Jacobians using the partial update approach and so step (i) will cost $O(n^3)$ operations as it would for a full Newton method rather than the $O(n^2)$ needed for the methods of Section 4 in which $C(x)$ is neglected. The research questions here are many and obvious.

Frequently in sparse problems, $J(x)$ is either itself available or more likely, efficiently obtainable by finite differences using the clever technique of [16]. In such cases, the expense in step (i) is in obtaining the matrix factorization. Cline [15] has suggested an LU updating technique and Marwil [50] has analyzed a technique in which L is left fixed for several iterations while a sparse Broyden update of U_k into $Q(L^{-1}y_k, s_k)$ is done to obtain U_{k+1} . The $k+1$ st approximate Jacobian is then thought of as LU_{k+1} . If this idea works out in practice, not only would the algorithm have important applicability in its own right, but a whole range of interesting possibilities open up such as updating only the L_k part of the SVD $U_k^T V^T$ of A_k until ill-conditioning forces a restart.

Finally Davidon [19] presents some new interesting ideas investigated further and isolated by Mei [52]. In [67], Schnabel shows their relationship to least change secant methods. Davidon and Nazareth are apparently enjoying success in implementing and testing these ideas at Argonne Labs.

6. Improving A Poor Initial Guess

In Sections 4 and 5 we saw an area in which there is much activity and in which the directions seem clear. Step (iv) of the general quasi-Newton algorithm is in pretty good shape. On the other hand there is some confusion but a good deal of activity in step (ii). How do we improve a poor x_0 to obtain a point at which $x_{k+1} = x_k + s_k^N$ will suffice? There is no important loss of generality for our purposes in restricting ourselves to UCMIN.

Line searches are by far the most commonly used techniques. These are implementations of the theoretical result that tells us to expect convergence for a convex f if we choose

$x_{k+1} = x_k + t_k^* s_k^N$ where $f(x_{k+1}) = \min_t f(x_k + t s_k^N)$. This is just a case of trading an n -dimensional problem we can't solve for a sequence of 1-dimensional problems we also can't solve. See [30], [31] or [62] for a relevant discussion of the secant methods.

Up until about ten years ago it was considered not unreasonable to make an honest attempt to find a close approximation to a value of t_k^* . Elaborate and expensive techniques such as bisection and Fibonacci search were employed. None of these routines are still competitive. Basically, modern line

search routines settle for any t_k which will make $f(x_{k+1}) < f(x_k)$. They count on finite precision arithmetic to keep t_k from becoming so small that $\langle x_k \rangle$ converges to a non-minimizer. The essence is in the choice of which value to try for t_k if the previous value wasn't acceptable. The first value tried is virtually always $t_k = 1$ for obvious reasons. As is so often the case in numerical analysis, these methods range from ad hoc to elegant with none clearly the most effective.

It would be pointless to discuss specific examples of searches here, but there is a very interesting open question. Powell [64] gives a fine convergence theorem for the BFGS with an implementable line search. What properties does the sequence $\langle A_k \rangle$ have to have for this theorem to go through? In particular does it hold for the DFP method? A negative result would be very interesting. Any example of an implementable line search for which a BFGS but not a DFP convergence theorem held would be well received. See [31]. Some other relevant papers are [14], [71].

The line search routines sometimes fail prematurely because the particular direction they are searching in is not a good one. It seems reasonable to change directions as we change t_k . For example, when a very short step is indicated it might be better to take it along $-\nabla f(x_k)$ rather than the Newton direction. This is the effect of the Levenberg [48]-Marquardt, [49], Goldfeldt-Quandt-Trotter [41] idea. Steps (i) and (ii) are combined and $u \geq 0$ is the search parameter in seeking an acceptable step $s_k(u)$ defined by

$$(uI + A_k)s_k(u) = -\nabla f(x_k).$$

It turns out that for any $u \geq 0$, $s_k(u)$ solves the constrained problem

$$\min [q_k(x_k + s) \equiv f(x_k) + \nabla f(x_k)^T s + \frac{1}{2} s^T A_k s]$$

for $\|s\|_2 \leq \|s_k(u)\|_2$.

A very appealing way to use this fact is to think of the quadratic q_k as a model of the objective function f which is accurate in some neighborhood of x_k . One carries along an estimate of a radius ρ_k of this trust region and attempts to stay within the region by picking $u = 0$ or so that $\|s_k(u)\|_2 \approx \rho_k$. Since each $s_k(u)$ is expensive, the practicality of this method depends on a good scheme for approximately solving the nonlinear equation $g(u) = \|s_k(u)\|_2 - \rho_k = 0$. Moré [54] gives a very clear exposition of an effective implementation based on [44]. Earlier Powell [59], [60], [61] had suggested the dogleg, a segmented approximation to $s_k(u)$, which avoided the problem of solving $g(u) = 0$. The reader will find an analysis of the dogleg in [63] and an exposition in [27].

The weakness of the model-trust region approach is in the setting of the initial trust radius ρ_0 . If $A_0 = I$, as is often the case for the least change secant methods, then there is no rational choice of ρ_0 . Experiments [52] show that the algorithm will recover from a poor ρ_0 but at a cost. Currently there is interest in using a line search to help start the trust region. This calls for experiment and analysis. If A_0 is a good approximation to $J(x_0)$ the $\rho_0 = \|s_0(0)\|_2 = \|s_0^N\|_2$ seems to work pretty well.

As we mentioned previously, the Hessian of f is usually

positive definite near x^* . It has been common practice to maintain $\langle A_k \rangle$ as a positive definite sequence since this at least had the advantage of making s_k^N a descent direction. Some interesting ideas [51], [55], [70] seem to be emerging for dealing with negative curvature. When x_0 is poor, an indefinite Hessian is not uncommon and so this work has a lot of potential.

There is steady activity in methods based on deforming the problem that must be solved into one for which x_0 is a very good initial guess. The solutions of the deformed problems are then tracked back as the deformation is relaxed. See [4], [5], [6], [7], [17] and also [32] and [46]. Boggs in his thesis traces the idea back to the mid 1800's. The current vogue probably dates to [17].

This area needs more work. The computational results are mixed. There seem to be problems when the solution curve of the parameterized problems cross a manifold on which $J(x)$ drops rank.

Acknowledgement: The author wishes to thank Charles Van Loan for carefully reading this manuscript and suggesting several clarifications. Concerning the unclear points that remain, we absolve him by admitting that he suggested more clarifications that we made.

REFERENCES

1. Abbott, J. P. and Brent, R. P. (1976), "Fast local convergence with single and multistep methods for nonlinear equations." To appear in J. Austral. Math. Soc. (Ser. B).
2. Bartholomew-Biggs, M. C. (1977), "The estimation of the Hessian matrix in nonlinear least squares problems with non-zero residuals," Math. Prog. 12, pp. 67-80.
3. Betts, J. T. (1976), "Solving the nonlinear least square problem: Application of a general method," J.O.T.A. 18, pp. 469-484.
4. Boggs, P. T. (1971), "The solution of nonlinear operator equations by A-stable integration techniques," SIAM J. Numer. Anal. 8, pp. 767-785.
5. Boggs, P. T. (1976), "The convergence of the Ben-Israel iteration for nonlinear least squares problems," Math. Comp. (to appear).
6. Boggs, P. T. (1977), "An algorithm, based on singular perturbation theory, for ill-conditioned minimization problems," SIAM J. Numer. Anal. 14, pp. 830-843.
7. Boggs, P. T. and Dennis, J. E., Jr. (1976), "A stability analysis for perturbed nonlinear iterative methods," Math. Comp. 30, pp. 1-17.
8. Brodli, K. W. (1977), "Unconstrained minimization," in The State of the Art in Numerical Analysis edited by D. Jacobs, Academic Press.
9. Broyden, C. G. (1965), "A class of methods for solving nonlinear simultaneous equations," Math. Com. 19, pp. 577-593.
10. Broyden, C. G. (1971), "The convergence of a class of double-rank minimization algorithms," Parts I and II, J. Inst. Math. Appl. 6, pp. 76-90, 222-236.
11. Broyden, C. G. (1971), "The convergence of an algorithm for solving sparse nonlinear systems," Math. Comp. 25, pp. 285-294.
12. Broyden, C. G., Dennis, J. E. and Moré, J. J. (1973), "On the local and superlinear convergence of Quasi-Newton methods," J.I.M.A. 12, pp. 223-246.

13. Bryan, C. A. (1968), "Approximate solutions to nonlinear integral equations," SIAM J. Numer. Anal. 5, pp. 151-155.
14. Burmeister, W. (1973), "Die Konvergenzordnung des Fletcher-Powell-Algorithmus," ZAMM 53, pp. 693-699.
15. Cline, A. K. (1976), "A descent method for the uniform solution of over-determined systems of linear equations," SIAM J. Numer. Anal. 13, pp. 293-309.
16. Curtis, A., Powell, M. and Reid, J. (1974), "On the estimation of sparse Jacobian matrices," J. Inst. Math. Appl. 13, pp. 117-120.
17. Davidenko, D. F. (1953), "On a new method of numerical solution of systems of nonlinear equations," Doklady Akad. Nauk. SSSR(N.S.) 88, pp. 601-602.
18. Davidon, W. C. (1959), "Variable metric method for minimization," Argonne Nat. Labs, report ANL-5990 Rev.
19. Davidon, W. C. (1975), "Optimally conditioned algorithms without line searches," Math. Prog. 9, pp. 1-30.
20. Dennis, J. E., Jr. (1968), "On Newton-like methods," Math. 11, pp. 324-330.
21. Dennis, J. E., Jr. (1971), "On the convergence of Broyden's method for nonlinear systems of equations," Math. Comp. 25, pp. 559-567.
22. Dennis, J. E., Jr. (1971), "Toward a unified convergence theory for Newton-like methods" in Nonlinear Functional Analysis and Applications edited by L. B. Rail, Academic Press, New York.
23. Dennis, J. E., Jr. (1972), "On some methods based on Broyden's secant approximation to the Hessian," in Numerical Methods for Non-linear Optimization edited by F. A. Lootsma, Academic Press, London.
24. Dennis, J. E. (1976), "A brief survey of convergence results for quasi-Newton methods," in Nonlinear Programming edited by Cottle, R. and Lemke, C. SIAM-AMS, Proc. Vol. IX.
25. Dennis, J. E., Jr. (1977), "Nonlinear least squares and equations," in The State of the Art in Numerical Analysis edited by D. Jacobs, Academic Press.
26. Dennis, J. E., Gay, D. M. and Welsch, R. E. (1977), "An adaptive nonlinear least-squares algorithm," Cornell Computer Science TR77-321, (submitted for publication).

27. Dennis, J. E. and Mei, H. W. (1975), "An unconstrained optimization algorithm which uses function and gradient values," Cornell Computer Science TR77-246, (to appear in J.O.T.A.).
28. Dennis, J. E. and Moré, J. J. (1974), "A characterization of superlinear convergence and its application to quasi-Newton methods," Math. Comp. 28, pp. 549-560.
29. Dennis, J. E. and Moré, J. J. (1977), "Quasi-Newton methods, motivation and theory," SIAM Review 19, pp. 46-89.
30. Dixon, L. C. W. (1972a), "Quasi-Newton family generate identical points," Math. Prog. 2 pp. 383-387.
31. Dixon, L. C. W. (1972b), "The choice of step length, a crucial factor in the performance of variable metric algorithms," in Numerical Methods for Non-linear Optimization edited by F. A. Lootsma, Academic Press, London.
32. Eaves, B. C. (1976), "A short course in solving equations with PL homotopies," in Nonlinear Programming edited Cottle, R. and Lemke, C. SIAM-AMS Proc. Vol. IX.
33. Fletcher, R. (1970), "A new approach to variable metric algorithms," Comput. J. 13, pp. 317-322.
34. Fletcher, R. (1977), "Methods for solving non-linearly constrained optimization problems," in The State of the Art in Numerical Analysis edited by D. Jacobs, Academic Press.
35. Fletcher, R. and Powell, M. J. D. (1963), "A rapidly convergent descent method for minimization," Comput. J. 6, pp. 163-168.
36. Gay, D. M. (1977), "Some convergence properties of Broyden's method," NBER Working Paper No. 175.
37. Gill, P. E. Golub, G., Murray, W., Saunders, M. A. (1974), "Methods for modifying matrix factorizations," Math. Comp. 28, pp. 505-536.
38. Gill, P. E. and Murray, W. (1972), "Quasi-Newton methods for unconstrained minimization," J.I.M.A. 9, pp. 91-108.
39. Goldfarb, D. (1970), "A family of variable-metric methods derived by variational means," Math. Comp. 24, pp. 23-26.
40. Goldfarb, D. (1976), "Factorized variable metric methods for unconstrained optimization," Math. Comp. 30, pp. 796-811.
41. Goldfeldt, S. M., Quandt, R. E. and Trotter, H. F. (1966), "Maximization by quadratic hill-climbing," Econometrica 34 pp. 541-551.

42. Golub, G. H. and Pereyra, V. (1973), "The differentiation of pseudoinverses and nonlinear least squares problems whose variables separate," SIAM J. Numer. Anal. 10, pp. 413-432.
43. Greenstadt, J. (1970), "Variations on variable-metric methods," Math. Comp. 24, pp. 1-18.
44. Hebden, M. D. (1973), "An algorithm for minimization using exact second derivatives," A.E.R.E. Harwell Rpt. T.P. 515.
45. Kaufman, L. (1975), "A variable projection method for solving separable nonlinear least squares problems," BIT 15, pp. 49-57.
46. Kellogg, R., Lui, T. and Yorke, J. (1976), "A constructive proof of the Brouwer Fixed Point Theorem and computational results," SIAM J. Numer. Anal. 13, pp. 473-483.
47. Krogh, F. T. (1974), "Efficient implementation of a variable projection algorithm for nonlinear least squares problems," Comm. ACM 17, pp. 167-169.
48. Levenberg, K. (1944), "A method for the solution of certain nonlinear problems in least squares," Quart. Appl. Math. 2, pp. 164-168.
49. Marquardt, D. W. (1963), "An algorithm for least squares estimation of nonlinear parameters," SIAM J. Appl. Math. 11, pp. 431-441.
50. Marwil, E. S. (1978), "Exploiting sparsity in Newton-like methods," Ph.D. thesis, Cornell University.
51. McCormick, G. P. (1977), "A modification of Armijo's step-size rule for negative curvature," Math. Prog. 13, pp. 111-115.
52. Mei, H. H. W. (1977), "An analysis and implementation of Davidson's techniques for unconstrained optimization," Ph.D. thesis, Cornell University.
53. Meyer, Gunter H. (1968), "On solving nonlinear equations with a one-parameter operator imbedding," SIAM J. Numer. Anal. 5, pp. 739-752.
54. Moré, J. J. (1977), "The Levenberg-Marquardt algorithm: implementation and theory," to appear in Proc. 1977 Dundee Conf. on Numer. Anal.
55. Moré, J. J. and Sorensen, D. (1977), "A modified Newton method using directions of negative curvature," Argonne National Labs Rpt.

56. Oren, S. S. (1973), "Self-scaling variable metric algorithm without line search for unconstrained minimization," Math Comp. 27, pp. 873-885.
57. Ortega, J. M. and Rheinboldt, W. C. (1970), Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York.
58. Osborne, M. R. (1975), "Some special nonlinear least squares problems," SIAM J. Numer. Anal. 12, pg. 571.
59. Powell, M. J. D. (1970a), "A hybrid method for nonlinear equations," in Numerical Methods for Nonlinear Algebraic Equations, P. Rabinowitz, ed., Gordon and Breach, London.
60. Powell, M. J. D. (1970b), "A FORTRAN subroutine for solving systems of nonlinear algebraic equations," in Numerical Methods for Nonlinear Algebraic Equations, P. Rabinowitz, ed., Gordon and Breach, London.
61. Powell, M. J. D., (1970), "A new algorithm for unconstrained optimization," in Nonlinear Programming edited by J.B. Rosen, O.L. Maugasarian, K. Ritter, Academic Press, New York.
62. Powell, M. J. D. (1972), "Some properties of the variable metric method," in Numerical Methods for Nonlinear Optimization, edited by F.A. Lootsma, Academic Press, London.
63. Powell, M. J. D. (1974), "Convergence properties of a class of minimization algorithms," A.E.R.E. Harwell tech. rept. C.S.S.8.
64. Powell, J. J. D. (1975), "Some global convergence properties of a variable metric algorithm for minimization without exact line searches," in Nonlinear Programming, edited by Cottle, R. and Lemke, C., SIAM-AMS, Proc. Vol IX.
65. Powell, M. J. D. (1976), "Algorithms for nonlinear constraints that use Lagrangian functions," presented at Ninth International Symposium on Math. Prog., Budapest.
66. Ruhe, A. and Wedin, P. A. (1974), "Algorithms for separable nonlinear least squares problems," Univ. of Umea, Dept. of Info. Processing, rept. UMINF-47.74.
67. Schnabel, R. B. (1977), "Analyzing and improving quasi-Newton methods for unconstrained optimization", PhD thesis, Cornell University.
68. Schubert, L. K. (1970), "Modification of a quasi-Newton method for nonlinear equations with a sparse Jacobian," Math. Comp. 24, pp. 27-30.

69. Shanno, D. F. (1970), "Conditioning of Quasi-Newton methods for function minimization," Math. Comp. 24, pp. 647-656.
70. Sorensen, D. C. (1977), "Updating the symmetric indefinite factorization with application in a modified Newton method," Argonne National Labs. Rpt. 77-49.
71. Stoer, J. (1975), "On the convergence rate of imperfect minimization algorithms in Broyden's β -class," Math. Prog. 9 pp. 313-335.
72. Tapia, R. A. (1977), "Equivalences of quasi-Newton methods for constrained optimization and a new class of algorithms, Rice University, Dept. of Math Sciences.
73. Toint, Ph. L. (1977), "On sparse and symmetric matrix updating subject to a linear equations," to appear Math. Comp.
74. Toint, Ph. L. (1977), "Some numerical results using a sparse matrix updating formula in unconstrained optimization," Cambridge University, Dept. of App. Math. and Th. Phys. Report DAMTP 77/NA4.
75. Wedin, P. A. (1974b), "On surface dependent properties of methods for separable nonlinear least squares problems," Inst. for tellampad matematik, Box 5073, Stockholm 5, ITM Arbetsrapport nr. 23.

Appendix A.Matrix Factorizations

The schedule for the AMS Shortcourse for which the preceding paper was prepared, calls for the material on nonlinear optimization to be presented after Cleve Moler lectures on numerical linear algebra. Thus, the main body of the paper doesn't go into details on the computation of the Newton step by solving the linear system in step (i) of the quasi-Newton algorithm of section 2.

This omission is made partly for brevity and partly to avoid any confusion that might arise from slight differences in notation. The purpose of this appendix is to provide enough of an outline to make the preceding paper somewhat self-contained. More details can be found in Introduction to Matrix Computation by G. W. Stewart III, Academic Press, 1973.

Mathematics students taking their first numerical analysis course are often surprised to learn that computing A^{-1} and then $A^{-1}b$ is a bad way to solve $Ax = b$. Numerical methods usually involve decomposing A into the product of from two to four matrices. The form of the decomposition depends on properties of A and the computing environment but the candidate factors should be so that it is easy to solve a linear system with one of the factors as coefficient matrix. In general, the factorization, say $A = A_1 \cdot A_2 \cdot A_3$ is the first step. One then thinks of successively solving $A_1 x_1 = b$ for x_1 and then $A_2 x_2 = x_1$ for x_2 and finally $A_3 x = x_2$ for x . As a practical matter, some of the intermediate linear systems are often solved as the factorization stage proceeds.

As an illustration, let us consider the traditional form of

A2

Gaussian elimination in which rows (or equations) are interchanged if a zero appears on the main diagonal of the upper triangular matrix being generated. This algorithm corresponds to a factorization $A = P^T LU$ where P is a permutation matrix and L and U are lower and upper triangular respectively. But the solution to $P^T x_1 = b$, $x_1 = Pb$ as well as the solution x_2 to $Lx_2 = x_1 = Pb$ are often generated as the factorization stage proceeds. This leaves the system $Ux = x_2$ as the only obvious "factor" system to be solved. This example really illustrates the use of all the factorizations to solve $Ax = b$ and we complete this appendix by cataloging the factorization we refer to.

The LU factorization: In fact we use this term for the decomposition

$$PA = LU \text{ or } A = P^T LU$$

where P is a permutation matrix, L is unit lower triangular ($l_{ii} = 1$) and U is upper triangular. This is "Gaussian elimination" and it is intended for general well-conditioned square problems.

The Cholesky factorization: If A is symmetric and positive definite, the $A = LL^T$ can be obtained for about half the work and storage of $A = LU$. Generally in practice one uses the related decomposition $A = LDL^T$ where L is unit lower triangular and D is a positive diagonal matrix.

The symmetric indefinite factorization: In this case, A is only assumed symmetric and the decomposition is $A = PLDL^T P^T$ where P .

A3

is a permutation matrix, L is unit lower triangular and D is a block diagonal matrix of 1x1 and 2x2 blocks.

The QR decomposition: In this case, A has no special properties. In fact, it could even be rectangular. We write $A = QRP^T$ where P is a permutation matrix, R is upper triangular and Q is orthogonal, i.e., $Q^T Q = I$.

The SVD or singular value decomposition: Here, A may not only be rectangular but it can even be rank deficient. The decomposition is $A = U \Sigma V^T$ where U and V are orthogonal matrices and Σ is a nonnegative diagonal matrix. The rank of A is the rank of Σ . The SVD is related to the polar decomposition of A:

$$A = (UV^T)(V \Sigma V^T).$$