MATHEMATICS OF COMPUTATION
VOLUME 38, NUMBER 158
APRIL 1982, PAGES 459-474

Direct Secant Updates of Matrix Factorizations*

By J. E. Dennis, Jr. and Earl S. Marwil**

Abstract. This paper presents a new context for using the sparse Broyden update method to solve systems of nonlinear equations. The setting for this work is that a Newton-like algorithm is assumed to be available which incorporates a workable strategy for improving poor initial guesses and providing a satisfactory Jacobian matrix approximation whenever required. The total cost of obtaining each Jacobian matrix, or the cost of factoring it to solve for the Newton step, is assumed to be sufficiently high to make it attractive to keep the same Jacobian approximation for several steps. This paper suggests the extremely convenient and apparently effective technique of applying the sparse Broyden update directly to the matrix factors in the iterations between reevaluations in the hope that fewer fresh factorizations will be required. The strategy is shown to be locally and q-superlinearly convergent, and some encouraging numerical results are presented.

1. Introduction. Consider iterative methods for the solution of a nonlinear system of equations. Given a function F: $\Omega \subset \mathbf{R}^n \to \mathbf{R}^n$, find a solution to $F(x) = 0$. Assume that a solution $x_* \in \Omega$ exists. Let $F'(x) \equiv J_F(x) = J(x)$. Suppose there exist constants $\{\gamma_{ij}: i, j = 1,\ldots,n\}$ such that

$$|e_i^T\left[J(x) - J(x_*)\right]e_j| \leq \gamma_{ij}\|x - x_*\|_2 \tag{1.1}$$

for all $x \in \Omega$, where e_i is the ith column of the identity matrix. Further assume that $J(x_*)$ is nonsingular.

Generalized secant methods are used frequently for this problem. Given an approximation x_k to x_*, obtain a better approximation $x_{k+1} = x_k + s_k^N$, which is the solution to the affine problem

$$M_k(x) \equiv F(x_k) + B_k(x - x_k) = 0, \tag{1.2}$$

a model of $F(x) = 0$ for x near x_k. Locally, this *direct prediction* step is taken. Even when the model is not good the quasi-Newton step s_k^N is used in the determination of x_{k+1}. It is computed by solving

$$B_k s_k^N = -F(x_k), \tag{1.3}$$

which is equivalent to $M_k(x_k + s_k^N) = 0$.

This leaves the method for specification of $\{B_k\}$ to be chosen. Many choices are available; most commonly B_{k+1} is obtained either by a Broyden update of B_k,

$$B_{k+1} \equiv B_k + \frac{(y_k - B_k s_k)s_k^T}{s_k^T s_k}, \tag{1.4}$$

Received July 21, 1980; revised April 4, 1981.
1980 *Mathematics Subject Classification.* Primary 65H10.
* Research sponsored by ARO DAAG-79-C-0124 and NSF MCS 7906671.
** Currently at EG & G Idaho Inc., P. O. Box 1625, Idaho Falls, Idaho 83415.

0025-5718/81/0000-1066/$05.50

with $y_k = F(x_k + s_k) - F(x_k)$ and $s_k = x_{k+1} - x_k$, or from the exact Jacobian or a finite difference approximant, or by taking $B_{k+1} = B_k$ [7]. The Broyden update is the unique solution to the Frobenius norm minimization problem:

$$\min\{\|B - B_k\|_F : B \in Q(y_k, s_k) \equiv \{B \in \mathbf{R}^{n\times n} : Bs_k = y_k\}\}. \tag{1.5}$$

Of particular interest here is the class of problems for which the solution of (1.3) is by matrix factorization techniques, *and* the factorizations, required each time B_k is recomputed, represent an important part of the work necessary to obtain x_{k+1}. In a private communication, Alan Hindmarsh [13] has pointed out an instance of such a problem arising in the numerical solution of ordinary differential equations. In his example, the triangular factorization of $B_k = J(x_k)$ requires 20 relative computational units, while $F(x_k)$ and the analytic form of $J(x_k)$ require only 1 and 3 units, respectively. This is representative of a large and important class of problems in which the Jacobian is computed relatively cheaply because advantage can be taken of its sparsity. See also [4].

Define the subspace $Z_i \subset \mathbf{R}^n$, which identifies the sparsity structure of the ith row of the Jacobian, by

$$Z_i = \{v \in \mathbf{R}^n : e_j^T v = 0 \text{ for all } j \text{ such that } e_j^T J(x) e_i = 0 \text{ for all } x \in \Omega\}. \tag{1.6}$$

Then the subspace $Z \subset \mathbf{R}^n \times \mathbf{R}^n$ that identifies the sparsity structure of the Jacobian is defined by

$$Z \equiv \{A \in \mathbf{R}^{n\times n} : A^T e_i \in Z_i \text{ for } i = 1, 2, \ldots, n\}. \tag{1.7}$$

The sparse Broyden update given by Schubert [17] and Broyden [2] has the property that if $B_k \in Z$, then $B_{k+1} \in Z$ also. To write the sparse Broyden update, we first define the "sparsity" projection operators S_i, $i = 1, 2, \ldots, n$, that project orthogonally in the Frobenius norm onto the spaces Z_i, $i = 1, 2, \ldots, n$. Then B_{k+1} can be written as

$$B_{k+1} = B_k + \sum_{i=1}^{n} \left[(S_i s_k)^T (S_i s_k)\right]^+ e_i^T (y_k - B_k s_k) e_i (S_i s_k)^T, \tag{1.8}$$

where $(\cdot)^+$ is the generalized inverse; for a scalar a, $a^+ = 0$, if $a = 0$, and $a^+ = a^{-1}$ for $a \neq 0$. Note that B_{k+1} is obtained from B_k at a cost proportional to the number of nonzero elements in a member of Z. The sparse Broyden update is the unique solution to the minimization problem:

$$\min\{\|B - B_k\|_F : B \in Q(y_k, s_k) \cap Z\}.$$

It seems clear why (1.8) has not been very widely used for problems like the one pointed out by Hindmarsh. Effectively, B_k is corrected by a rank n matrix, and the availability of a factorization of B_k does not reduce significantly the work necessary to obtain the corresponding factorization of B_{k+1}. In Hindmarsh's example, we see that the work for a Newton step is 24 computational units compared to 21 units for a sparse Broyden step. Even this small saving is possibly an overestimate, since it is quite likely that $B_{k+1} = B_k$ will be successful for several steps at a time if J is computed accurately at the start of a string of stationary steps. This saving is much more likely to accrue in Newton's method than in the sparse Broyden method.

The new method, which should allow a reduction in the number of matrix factorizations needed for convergence, is outlined in the next section. The theoretical

details are given in Section 3, and computational results indicating the utility of the method are presented in Section 4.

2. A Doolittle Updating Method. The algorithm outlined here is representative of a general technique for producing inexpensive updates of matrix factorizations. This is *not* the same as techniques for obtaining matrix updates in factored form as in the case of rank 1 and rank 2 updates [1], [12]. Here, the update is defined implicitly by the updated factors [5].

Consider the case when (1.3) is solved at each step by using a Doolittle decomposition with a partial pivoting strategy,

$$P_k B_k = L_k U_k$$

where P_k is a permutation matrix recording the row interchanges; $L_k \in \mathfrak{L}_k$ an affine subspace of lower triangular $n \times n$ matrices with ones on the diagonal; $U_k \in \mathfrak{U}_k$ a subspace of upper triangular $n \times n$ matrices. Assume that $\mathfrak{L}_k$ and $\mathfrak{U}_k$ reflect the sparsity of $P_k A = LU$ for $A \in Z$. P_k may be chosen to affect the sparsity of $\mathfrak{L}_k$ and $\mathfrak{U}_k$.

Think of P_k and L_k as carriers of the information on row operations which transform B_k into upper triangular form. Thus, if B_k is near $J(x_*)$ then $L_k^{-1} P_k J(x_*)$ ought to be well approximated by upper triangular matrices. The strategy then is to consider U_k to be such an approximation. Obtain U_{k+1} by a sparse Broyden update to U_k in the hope of improving the approximation by incorporating new information gained in making the step s_k; the cost is proportional to the number of nonzeros of an arbitrary element in $\mathfrak{U}_k$. If the update is successful, then take

$$P_{k+1} = P_k, \qquad L_{k+1} = L_k,$$

and $B_{k+1} \in Q(y_k, s_k)$ is implicitly defined by

$$P_{k+1} B_{k+1} = L_{k+1} U_{k+1}. \tag{2.1}$$

The sparse Broyden approximation (1.8) uses $y_k = F(x_k + s_k) - F(x_k)$ since $J(x_*)$ should be near $Q(y_k, s_k)$. In the transformed problem, a v_k is desired so that $U(x_*)$ is near $Q(v_k, s_k)$. From the discussion above then, the row manipulations applied to y_k give $v_k \equiv L_k^{-1} P_k y_k$, an approximation to $L_k^{-1} P_k J(x_*) s_k$, which is the change in the dependent variable expected from an upper triangular approximation of $L_k^{-1} P_k J(x_*)$ applied to the change in the independent variable.

The basic idea is to choose U_{k+1} to solve $\min\{\|U - U_k\|_F : U \in \mathfrak{U}_k \cap Q(v_k, s_k)\}$, where $\mathfrak{U}_k$ defines the sparseness structure and the projectors. If the intersection is nonempty, the update is given by

$$U_{k+1} = U_k + \sum_{i=1}^{n} \left[(S_i s_k)^T (S_i s_k)\right]^+ e_i^T (v_k - U_k s_k) e_i (S_i s_k)^T. \tag{2.2}$$

In fact, U_{k+1} solves

$$\min\{\|U - U_k\|_F : U \in \mathfrak{U}_k \text{ is a nearest point in } \mathfrak{U}_k \text{ to } Q(v_k, s_k)\}.$$

If the intersection is empty, this is a reasonable choice of U_{k+1}.

Unfortunately, neither $U(x_*)s_k$ nor $L(x_*)^{-1} P_k y_k$ is available, so we use $v_k = L_k^{-1} P_k y_k$ which approximates $L(x_*)^{-1} y_k$. This forces us to modify the basic algorithm to include a periodic restart and a test to prevent updating any row of U_k that is not adequately represented in the step s_k. For simplicity, the algorithm is stated under the assumption that a Newton step can be taken at each iteration.

Doolittle Updating Algorithm.

(1) Choose $x_0 \in \mathbf{R}^n$,
m, a fixed positive integer, and
β, a fixed positive number.
Set $k = 0$.

(2) Evaluate $F_0 = F(x_0)$, and
$J_0 = J(x_0)$, or a finite difference approximation to J_0.

(3) Factor $P_0 J_0 = L_0 U_0$ by a Doolittle scheme with partial pivoting.

(4) Solve $L_0 w_k = -P_0 F_k$ and $U_k s_k^N = w_k$.
Set $x_{k+1} = x_k + s_k^N$.

(5) If $k = m - 1$, set $x_0 = x_{k+1}$ and go to (2); else evaluate
$F_{k+1} = F(x_{k+1})$ and set $y_k = F_{k+1} - F_k$.

(6) Solve $L_0 v_k = P_0 y_k$.

(7) Update the jth row of U_k if $\|s_k\| \leqslant \beta \|S_j s_k\|$, i.e., we define
$X_\beta(s_k) = \{j: \|s_k\| \leqslant \beta \|S_j s_k\|\}$ and
$U_{k+1} = U_k + \sum_{j \in X_\beta(s_k)} [(S_j s_k)^T (S_j s_k)]^+ e_j^T (v_k - U_k s_k) e_j (S_j s_k)^T$.

Replace k by $k + 1$, and go to (4).

The role of the constant β is to prevent a correction to a row of U_k when the projected step along that row is too small relative to the full step. For example, if $\beta \geqslant \varepsilon^{-1}$ and $1 + \varepsilon = 1$, then no row of U_k is updated for which

$$\|S_j s_k\| + \|s_k\| = \|s_k\|$$

to working precision. For $j \in X_\beta^c(s_k) = \{1, 2, \ldots, n\} \setminus X_\beta(s_k)$ this has the effect of redefining the jth component of v_k to be the jth component of $U_k s_k$ before applying (2.2) to get U_{k+1}. The need for this precaution and for restarts every m iterations seems to be real. These are certainly useful in the convergence analysis given in the following section, but they appear to be more than just formal conveniences. Intuition suggests that v_k must be chosen so that $Q(v_k, s_k)$ approaches $U(x_*)$ faster than s_k goes to zero.

A value for β has already been suggested; now consider the choice for m. Rather than always restarting after the same fixed number of iterations, it is probably better to allow the need for a change in the pivot sequence, or the size of the proposed change in U_k, or even β_k, to trigger a restart step. The condition number estimates of [3] and [9] applied to U_{k+1} should be extremely useful in this context. Furthermore, a global implementation along the lines of Moré's *MINPACK* implementation of Powell's HYBRID [15] is anticipated. It is difficult to imagine a more reasonable set of restart rules than Moré's. In the preliminary tests presented in Section 4, the tests with m and β were omitted, effectively setting $m = \infty$ and $\beta = 0$, since a good initial guess was used.

Although Z is independent of x, a reevaluation of the Jacobian and subsequent Doolittle factorization possibly introduces a different pivot sequence and changes $\mathfrak{L}$ and $\mathfrak{U}$, and hence the definition of step (7) of the algorithm.

3. Convergence. In this section we give the convergence result for the Doolittle updating algorithm (Theorem 3.9). We begin with a formal description of the Doolittle decomposition and a discussion of pivoting strategies. Once a pivoting

strategy has been selected, we give a continuity result saying that the same pivoting strategy can be used at a "nearby" point. Finally we establish a bounded deterioration estimate relating U_{k+1} to U_k. Some preliminary results are required which make this section rather lengthy. The proofs of Theorems 3.3 and 3.9 have been placed in Appendix 1 for completeness without clutter.

The Doolittle decomposition is a particular implementation of Gaussian elimination that produces triangular factors, L and U, with L unit lower triangular and U upper triangular. The following algorithm is the Doolittle decomposition with partial pivoting.

Algorithm 3.1 [19].

For $r = 1, 2, \ldots, n$ do steps 1 through 4

(1) Compute $u_{rj} = a_{rj} - \sum_{k=1}^{r-1} l_{rk} u_{kj}, j = r, \ldots, n$.

(2) Set int_r equal to the smallest $k \geqslant r$ for which $|a_{kr}| = \max_{r \leqslant j \leqslant n} |a_{jr}|$.

(3) Interchange the rth and int_rth rows of the array. Refer to these rows by their new positions.

(4) Compute $l_{ir} = (a_{ir} - \sum_{k=1}^{r-1} l_{ik} u_{kr})/u_{rr}$, $i = r + 1, \ldots, n$.

Steps (2) and (3) are the partial pivoting strategy. We will refer to the "LU decomposition without pivoting" meaning the algorithm without steps (2) and (3). The following result establishes when the LU decomposition without pivoting may be carried out.

THEOREM 3.2 [19]. *Let $A \in L(\mathbf{R}^n)$ be nonsingular. Then the following are equivalent*:

(a) *The LU decomposition without pivoting can be carried out.*

(b) *There is a unique unit lower triangular matrix L and a nonsingular upper triangular matrix U such that $A = LU$.*

(c) *All the leading principal submatrices of A are nonsingular.*

Now, consider any pivoting strategy P such that the LU decomposition of PA can be carried out, then Theorem 3.2 will apply to PA. The standard partial pivoting strategy chooses the first element of maximum size in a particular column on or below the main diagonal. This is designed for numerical stability, but can destroy the sparsity of the problem. To preserve, as much as possible, the sparseness of the factors, we should use a pivoting strategy with some flexibility. Duff [10] calls "threshold pivoting" a strategy that allows, for some $T > 0$, the selection of nonzero pivots which are at least $1/T$ times the largest element in a row or column. Erisman and Reid [11] give a formula to monitor the growth of the matrix elements in the factorization, using only the sparsity pattern and T. If the growth is too large, a smaller value of T can be chosen. Duff also notes that $T = 10$ yields good retention of sparsity and generally good numerical accuracy. He also recommends iterative refinement to improve the solution, since solutions for systems with sparse triangular factorizations are fairly cheap.

For $T \geqslant 1$, the pivot rule for threshold pivoting can be stated.

(2′, 3′) A row interchange must be made if

$$|l_{kr}| > T|l_{rr}| \quad \text{for some } k = r + 1, \ldots, n.$$

This allows a choice of interchanges to retain as much sparsity as possible in the decomposition, while still pivoting to hold down element growth.

In proving a convergence result of the same nature as for other Newton-like methods, a continuity assumption is required on the Jacobian. Now, using factorizations and an open pivoting strategy, we first must show that factorizations are continuous. The following theorem gives sufficient conditions for the LU decomposition to be continuous in a neighborhood of a point where the LU decomposition without pivoting exists. The proof is in Appendix 1.

THEOREM 3.3. *Let* A: $\mathbf{R}^n \to \mathbf{R}^{n\times n}$, *and suppose that for some* $x_0 \in \mathbf{R}^n$, $A(x_0)$ *is nonsingular, and that there exist* $\varepsilon_0 > 0$ *and* $\gamma_{ij} \geqslant 0$ *such that*

$$|e_i^T[A(x) - A(y)]e_j| \leqslant \gamma_{ij}\|x - y\|_2$$

for $i, j = 1, 2, \ldots, n$ *and for all* $x, y \in N(x_0, \varepsilon_0)$. *If the LU decomposition without pivoting exists at* x_0, $A(x_0) = L(x_0)U(x_0)$, *then there exists* $\varepsilon > 0$ *such that the decomposition without pivoting exists for all* $x \in N(x_0, \varepsilon)$.

Furthermore, there exist constants $c_0, d_0 > 0$ *such that*

$$\|L(x) - L(x_0)\|_F \leqslant c_0\|x - x_0\|_2 \quad \text{and} \quad \|U(x) - U(x_0)\|_F \leqslant d_0\|x - x_0\|_2$$

for all $x \in N(x_0, \varepsilon)$.

For the local convergence of the Doolittle updating algorithm we must have x_0 sufficiently close to x_* such that the pivoting strategy selected at x_0 also works at x_*. The next theorem and corollary establish the relationship.

THEOREM 3.4. *Let* A: $\mathbf{R}^n \to \mathbf{R}^{n\times n}$ *be continuous and nonsingular at* x_*. *If* $PA(x_*)$ *is any row permutation of* $A(x_*)$ *for which* $PA(x_*)$ *does not have an LU decomposition without pivoting, then, for any* $T > 0$, *there exists* $\eta_T > 0$ *such that no threshold pivoting strategy based on* T *would select a pivot sequence corresponding to* P *for any* $x \in N(x_*, \eta_T)$.

Proof. Since $PA(x_*)$ does not have an LU decomposition without pivoting, then for some k, $1 \leqslant k < n$, $l_{kk}(x_*) = 0$ ($k \neq n$ because A is nonsingular). But $PA(x_*)$ is nonsingular, so $l_{k+i,k}(x_*) \neq 0$ for some i, $1 \leqslant i \leqslant n - k$. Thus, by the continuity of the decomposition, there exists $\eta_T > 0$ such that, for $x \in N(x_*, \eta_T)$, $PA(x)$ is nonsingular and $|l_{k+i,k}(x)| > T|l_{kk}(x)|$. Thus P would not be selected by a pivot strategy for $A(x)$.

COROLLARY 3.5. *Let* A *be as in Theorem* 3.4. *For any threshold pivoting strategy, there exists* $\bar{\eta}_T$ *such that if* $x_0 \in N(x_*, \bar{\eta}_T)$ *and if* P_0 *is a pivot sequence for which* $P_0A(x_0)$ *has an LU decomposition without further pivoting,* $P_0A(x_0) = LU$, *then* $P_0A(x_*)$ *can be factored without pivoting.*

Proof. Let $\mathcal{P}_* = \{P\colon PA(x_*)$ does not have an LU decomposition without pivoting$\}$. Note that $\mathcal{P}_*$ is a finite set, so $\mathcal{P}_* = \{P_1, P_2, \ldots, P_m\}$. From Theorem 3.4, for each P_i there exists η_i such that the permutation matrix P_i would not be selected by the pivot strategy for any $A(x)$, $x \in N(x_*, \eta_i)$. Let $\bar{\eta}_T = \min_{1\leqslant i\leqslant m} \eta_i$. Let $x_0 \in N(x_*, \bar{\eta}_T)$, and apply the algorithm to obtain $P_0A(x_0) = LU$. Then $P_0 \notin \mathcal{P}_*$ so $P_0A(x_*)$ can be factored.

Next we give a bounded deterioration result relating the updated triangular factor U_{k+1} to U_k.

LEMMA 3.6. *Let* F: $\mathbf{R}^n \to \mathbf{R}^n$ *and assume that there exists* $x_* \in \mathbf{R}^n$ *such that* $F(x_*) = 0$ *and* $J_* = J(x_*)$ *is nonsingular. Assume that there exists* $\varepsilon > 0$ *such that* (1.1) *holds for all* $x \in N(x_*, \varepsilon)$. *Then, given a pivoting strategy* P_0, *there exists* $\varepsilon_0 \in (0, \varepsilon]$ *such that, if the LU decomposition without pivoting of* $P_0 J(x_0) = L(x_0)U(x_0)$ *exists at* $x_0 \in N(x_*, \varepsilon_0)$, *then* $P_0 J(x_*)$ *can be factored without pivoting.*

Furthermore, $\{U_l\}_{l=0}^{m-1}$ *defined by step* (7) *of the Doolittle updating algorithm satisfies*

$$\|U_{l+1} - U_*\|_F^2 \leq \|U_l - U_*\|_F^2 + \|L_0^{-1}\|_2^2 m\beta^2 \left[\kappa\sigma_l + \|U_*\|_2 c_0 \|x_0 - x_*\|_2\right]^2,$$

where $\beta > 0$ *is set in step* (1) *of the algorithm,* $\kappa = \|\Gamma\|_F$, $\Gamma = (\gamma_{ij})$, *and* $\sigma_l = \max(\|x_{l+1} - x_*\|_2, \|x_l - x_*\|_2)$.

Proof. The first assertion follows from Corollary 3.5, taking $\varepsilon_0 = \min(\varepsilon, \bar{\eta}_T)$; furthermore, it also follows that $P_0 J(x)$ has an LU decomposition without pivoting for all $x \in N(x_*, \varepsilon_0)$. Let $P_0 J(x_*) = L_* U_*$. Since $\varepsilon_0 < \varepsilon$, Theorem 3.3. gives $c_0 > 0$ such that

$$\|L(x) - L_*\|_F \leq c_0 \|x - x_*\|_2$$

for all $x \in N(x_*, \varepsilon_0)$.

To prove the second assertion, evaluate $\|U_{l+1} - U_*\|_F^2$ using the update in step (7). Set $X_\beta(s_l) = \{i: \|s_l\|_2 \leq \beta \|(s_l)_i\|_2$, where $(s_l)_i = S_i s_l\}$. Then,

$$\begin{aligned}
\|U_{l+1} - U_*\|_F^2 &= \left\| U_l - U_* + \sum_{X_\beta(s_l)} \left[(s_l)_i^T (s_l)_i\right]^+ e_i^T \left(L_0^{-1} P_0 y_l - U_l s_l\right) e_i (s_l)_i^T \right\|_F^2 \\
&= \sum_{X_\beta(s_l)} \Big\{ \|e_i^T \left(U_l - U_*\right)\left[I - (s_l)_i^T (s_l)_i\right]^+ (s_l)(s_l)_i^T\|_2^2 \\
&\qquad + \|e_i^T \left(L_0^{-1} P_0 y_l - U_* s_l\right)\left[(s_l)_i^T (s_l)_i\right]^+ (s_l)_i^T\|_2^2 \Big\} + \sum_{X_\beta^c(s_l)} \|e_i^T \left(U_l - U_*\right)\|_2^2 \\
&\leq \|U_l - U_*\|_F^2 + \sum_{X_\beta(s_l)} \|\left[(s_l)_i^T (s_l)_i\right]^+ e_i^T L_0^{-1} P_0 \big(y_l - L_* U_* s_l \\
&\qquad + L_* U_* s_l - L_0 U_* s_l\big)(s_l)_i^T\|_2^2 \\
&\leq \|U_l - U_*\|_F^2 + \sum_{X_\beta(s_l)} \Big\{ \|\left[(s_l)_i^T (s_l)_i\right]^+ e_i^T L_0^{-1} P_0 \left(y_l - J_* s_l\right)(s_l)_i^T\|_2 \\
&\qquad + \|\left[(s_l)_i^T (s_l)_i\right]^+ e_i^T L_0^{-1} P_0 \left(L_* - L_0\right) U_* s_l (s_l)_i^T\|_2 \Big\} \\
&\leq \|U_l - U_*\|_F^2 + \sum_{X_\beta(s_l)} \Big[\|L_0^{-1}\|_2 \big(\|(s_l)_i\|_2^+ \|y_l - J_* s_l\|_2 \\
&\qquad + \|(s_l)_i\|_2^+ \|L_* - L_0\|_2 \|U_*\|_2 \|s_l\|_2\big)\Big]^2 \\
&\leq \|U_l - U_*\|_F^2 + \|L_0^{-1}\|_2^2 \beta^2 \sum_{i=1}^{m} k \left(\kappa\sigma_l + \|U_*\|_2 c_0 \|x_0 - x_*\|_2\right)^2,
\end{aligned}$$

by the update criterion in step (7) of the algorithm, the Lipschitz continuity of J, and Theorem 3.3,

$$\leqslant \|U_l - U_*\|_F^2 + \|L_0^{-1}\|_2 \beta^2 m\left(\kappa\sigma_l + \|U_*\|_2 c_0 \|x_0 - x_*\|_2\right)^2$$

which is the desired result.

The most convenient norm for the error estimates is the weighted norm established at the start of the iteration.

Definition 3.7. Define the left weighted Frobenius norm $\|\cdot\|_{L_0^{-1},F}$: $L(\mathbf{R}^n) \to \mathbf{R}^n$ by

$$\|\cdot\|_{L_0^{-1},F} \equiv \|L_0^{-1}\cdot\|_F$$

for L_0 nonsingular and lower triangular.

The next lemma relates the various norms.

LEMMA 3.8. *Let $L: \mathbf{R}^n \to \mathbf{R}^{n\times n}$ be a continuous operator into the lower triangular matrices. Suppose $L^{-1}(x) = L(x)^{-1}$ exists and is continuous on $N(x_*, \varepsilon)$, for some $x_* \in \mathbf{R}^n$ and $\varepsilon > 0$. Then there exist constants $\bar{\eta}$ and $\hat{\eta} > 0$ such that*

$$\|\cdot\|_{L^{-1}(x),F} \leqslant \bar{\eta}\|\cdot\|_F \quad \text{for all } x \in \bar{N}\left(x_*, \varepsilon\right)$$

and

$$\|\cdot\|_F \leqslant \hat{\eta}\|\cdot\|_{L^{-1}(x),F} \quad \text{for all } x \in \bar{N}\left(x_*, \varepsilon\right).$$

Proof. Let $\bar{\eta} = \sup_{x\in\bar{N}(x_*,\varepsilon)}\|L^{-1}(x)\|_2$ and $\hat{\eta} = \sup_{x\in\bar{N}(x_*,\varepsilon)}\|L(x)\|_2$. Then

$$\|\cdot\|_{L^{-1}(x),F} \leqslant \|L^{-1}(x)\cdot\|_F \leqslant \|L^{-1}(x)\|_2\|\cdot\|_F \leqslant \bar{\eta}\|\cdot\|_F,$$

and

$$\|\cdot\|_F \leqslant \|L(x)L^{-1}(x)\cdot\|_F \leqslant \|L(x)\|_2\|L^{-1}(x)\cdot\|_F \leqslant \hat{\eta}\|\cdot\|_{L^{-1}(x),F}.$$

Let $\eta > 0$ be such that the matrix norms satisfy $\|\cdot\|_F \leqslant \eta\|\cdot\|_2$.

We now state the convergence result for the Doolittle updating algorithm. The proof is in Appendix 1.

THEOREM 3.9. *Let $F: \mathbf{R}^n \to \mathbf{R}^n$ be continuously differentiable in an open convex set D_0. Assume that there exists $x_* \in D_0$ such that $F(x_*) = 0$ and $J(x_*)$ is nonsingular. Assume that there exists $\Gamma = (\gamma_{ij})$ such that for all $x, x' \in D_0$,*

$$|e_i^T[J(x) - J(x')]e_j| \leqslant \gamma_{ij}\|x - x'\|, \qquad 1 \leqslant i, j \leqslant \eta.$$

Then, given a pivoting strategy P_0, there exist $\varepsilon, \delta > 0$ such that if $\|x_0 - x_*\|_2 < \varepsilon$, if $P_0J(x_0)$ has an LU decomposition without pivoting, and if $\|J_0 - J_*\|_F < \delta$, then the Doolittle updating algorithm generates $\{x_k\}$ which converges locally and q-superlinearly to x_*.

4. Numerical Results. An abbreviated version of the new algorithm was tested without using m or β. The test problems are the ones Broyden [2] used for initial testing of the sparse Broyden update. They are all banded systems.

Problems 1–6: $f_i = (3 - k_1x_i)x_i + 1 - x_{i-1} - 2x_{i+1}$.

Problems 7–23: $f_i = (k_1 - k_2x_i^2)x_i + 1 + k_3\sum_{j=r_1;\, j\neq i}^{r_2} x_j + x_j^2$.

For $i < 1$ or $i > n$, $x_i = 0$. The parameters k_1, k_2, and k_3 are varied to increase the nonlinearity of the problem, while r_1 and r_2 determine the bandwidth. The initial guess in each case was $x_i = -1$, $i = 1, 2, \ldots, n$. The convergence criterion used was

$\|F\|_2 < 10^{-6}$. All the algorithms converged from this guess and we felt that this situation provided the best test of our new algorithm since a poor Jacobian approximation method is more likely to inhibit local rather than global convergence in a sophisticated implementation.

Four algorithms were compared, each starting with a finite difference Jacobian generated by the Curtis, Powell and Reid technique [4] to economize function evaluations. The algorithms are: a finite difference Newton, a finite difference Newton with fixed initial Jacobian, the sparse Broyden, and the scheme given here. These are flow-charted concurrently in Figure 1. The numerical results are presented in Table 1.

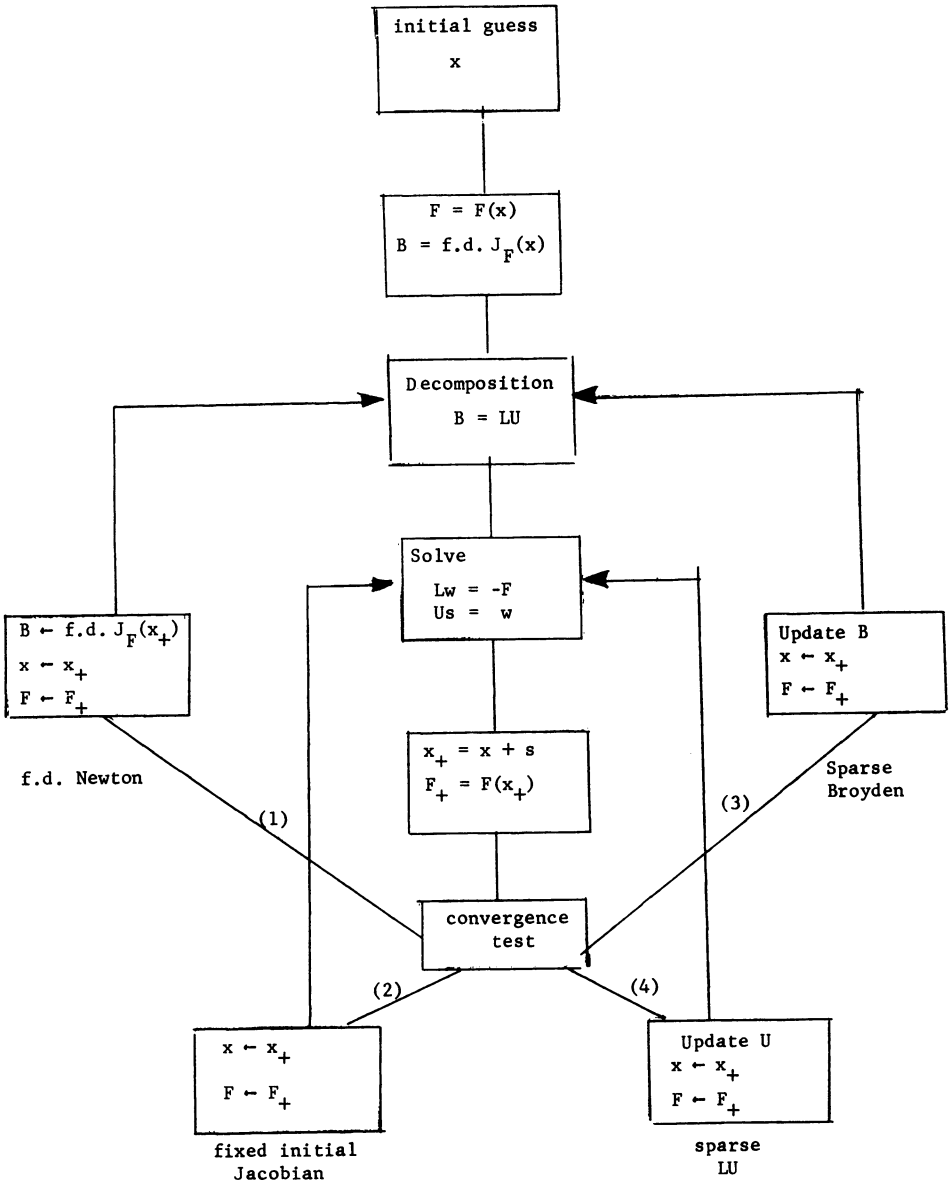

FIGURE 1

The important comparisons are in the number of factorizations, NFAC, and the number of function evaluations, NFEV. For these problems, it is clear that the finite difference Newton method costs more than the fixed initial Jacobian method which costs more than the Doolittle updating method. No pivoting was used in any of the algorithms.

			(1) f.d. Newton		(2) fixed initial Jacobian NFAC = 1	(3) sparse Broyden	(4) sparse LU NFAC = 1
PROB	DIM	BANDW	IT = NFAC	NFEV	IT = NFEV	IT = NFAC = NFEV	IT = NFEV
1	5	3	3	12	8	5	5
2	5	3	3	12	7	4	5
3	10	3	3	12	11	5	6
4	20	3	4	16	15	5	6
5	600	3	4	16	18	5	6
6	600	3	4	16	15	7	8
7	100	7	4	32	11	6	7
8	100	7	4	32	11	6	7
9	100	7	4	32	13	6	7
10	50	11	4	48	10	6	7
11	50	11	4	48	14	7	8
12	50	11	4	48	14	8	8
13	50	11	5	60	23	9	12
14	50	11	5	60	22	11	11
15	50	11	5	60	27	11	13
16	50	11	5	60	18	9	10
17	50	11	4	48	17	8	10
18	50	11	4	48	15	8	9
19	50	11	4	48	17	9	10
20	50	11	5	60	26	12	13
21	50	11	5	60	30	13	14
22	50	11	5	60	31	13	14
23	50	11	5	60	34	13	16

TABLE 1

The comparison with the sparse Broyden depends on the relative costs of NFAC and NFEV. Compared with the finite difference Newton method, the sparse Broyden costs less if 3 to 21 function evaluations cost more than one factorization. Compared with the fixed initial Jacobian, the sparse Broyden costs less if 1 to 3 function evaluations cost more than one factorization. But compared with the Doolittle updating scheme, it costs more whenever 1 function evaluation costs *less* than 3 factorizations. In other words, the function evaluations would have to be very expensive, compared to the factorizations for the sparse Broyden to be a better choice than our scheme for these problems. We should point out that since the Doolittle updating scheme has done so well on banded problems for which factorizations and finite difference Jacobians are so cheap, and from such good initial guesses that an inaccurate Jacobian can impede convergence, we expect even better results on general sparse problems.

5. Conclusion. We have delayed formal publication of this research for several years because we were unhappy with the need to refresh the Jacobian at intervals and not to update any row if the part of s is small that interacts with that row.

Recent work, [8], [14], has convinced us that these requirements are necessary for convergence and numerical stability even though we cannot give a rigorous proof of this conjecture. We have no doubt that the situation will become clarified in time, but we feel that it is silly to delay publication any longer of a potentially useful numerical method just because the analysis of the method is not provably sharp.

The ideas here are certainly applicable to other factorizations. In the current context, we really do not mean to make too much of the idea; we just suggest that it could be used in an implementation of Newton's method in place of any iteration when the Jacobian would otherwise have been left fixed.

Finally we remark that the Doolittle updating scheme is reported [20] to be five times faster than Newton's method for a particular application called the black oil model in reservoir engineering.

6. Acknowledgements. The reader will see that we have several references to private communications. This is indicative of the helpful conversations we have had with several colleagues about these ideas. We would also like to thank the referees for their careful reading and helpful suggestions.

Appendix 1. This appendix contains the proofs for Theorem 3.3 and Theorem 3.9.

Proof of Theorem 3.3. Since $A(x_0)$ is nonsingular and the LU decomposition without pivoting exists at x_0, it follows from Theorem 3.2 that the leading principal submatrices of $A(x_0) \equiv {}^kA(x_0)$, $k = 1,2,\ldots,n$, must be nonsingular. $A(x)$ and all of its submatrices are Lipschitz continuous in $N(x_0, \varepsilon_0)$. By the Inverse Function Theorem [16] there exist $\varepsilon_1, \varepsilon_2,\ldots,\varepsilon_n$ such that ${}^kA(x)$ is nonsingular in $N(x_0, \varepsilon_k)$. Let $\varepsilon = \min_{1\leq k\leq n} \varepsilon_k$. Then all the leading principal submatrices of $A(x)$ are nonsingular in $N(x_0, \varepsilon)$. Another application of Theorem 3.2 establishes the existence of the LU decomposition without pivoting for all $x \in N(x_0, \varepsilon)$.

Further restrict ε so that the preceding factorization holds on $\bar{N}(x_0, \varepsilon)$. The proof that the factors are Lipschitz continuous is by induction on the dimension [18]. Let $x \in N(x_0, \varepsilon)$ and $A(x) = L(x)U(x)$.

For $l = 1$, ${}^lA(x) = {}^1A(x) = a_{11}(x)$ and

$$a_{11}(x) = 1 \cdot a_{11}(x) = l_{11}(x) \cdot u_{11}(x) = {}^1L(x)\,{}^1U(x).$$

Thus

$$\|{}^1L(x) - {}^1L(x_0)\|_F = |\,1 - 1\,| = 0 \cdot \|x - x_0\|_2$$

and $\|{}^1U(x) - {}^1U(x_0)\|_F = |\,a_{11}(x) - a_{11}(x_0)\,| \leq \gamma_{11}\|x - x_0\|_2 \equiv c_1\|x - x_0\|_2$. By the Inverse Function Theorem, ${}^1L^{-1}$ and ${}^1U^{-1}$ are Lipschitz continuous in $\bar{N}(x_0, \varepsilon)$. Let $\hat{c}_1$ and $\hat{d}_1$ be the respective Lipschitz constants.

Assume, for $l = 2, 3,\ldots,k$, $k \leq n - 1$, that

$$\|{}^lL(x) - {}^lL(x_0)\|_F \leq c_l\|x - x_0\|_2 \quad \text{and} \quad \|{}^lU(x) - {}^lU(x_0)\|_F \leq d_l\|x - x_0\|_2,$$

for all $x \in N(x_0, \varepsilon)$; further assume that the inverses ${}^lL^{-1}(x)$ and ${}^lU^{-1}(x)$ are Lipschitz continuous on $N(x_0, \varepsilon)$ with Lipschitz constants $\hat{c}_l$ and $\hat{d}_l$ respectively.

For $l = k + 1$, let ${}^{k+1}A(x)$ be partitioned as

$$ {}^{k+1}A(x) = \left[\begin{array}{c|c} {}^{k}A(x) & {}^{k}v(x) \\ \hline {}^{k}w^T(x) & a_{k+1,k+1}(x) \end{array}\right]. $$

Since ${}^{k}A(x) = {}^{k}L(x){}^{k}U(x)$, ${}^{k+1}A(x)$ can be factored

$$ \begin{aligned} {}^{k+1}A(x) &= {}^{k+1}L(x){}^{k+1}U(x) \\ &= \left[\begin{array}{c|c} {}^{k}L(x) & 0 \\ \hline {}^{k}w^T(x){}^{k}U^{-1}(x) & 1 \end{array}\right]\left[\begin{array}{c|c} {}^{k}U(x) & {}^{k}L^{-1}(x){}^{k}v(x) \\ \hline 0 & \alpha_{k+1}(x) \end{array}\right] \end{aligned} $$

where $\alpha_{k+1}(x) = a_{k+1,k+1}(x) - {}^{k}w^T(x){}^{k}U^{-1}(x){}^{k}L^{-1}(x){}^{k}v(x)$.

Now,

$$ \begin{aligned} &\|{}^{k+1}L(x) - {}^{k+1}L(x_0)\|_F \\ &\quad = \left\| \left[\begin{array}{c|c} {}^{k}L(x) & 0 \\ \hline {}^{k}w^T(x){}^{k}U^{-1}(x) & 1 \end{array}\right] - \left[\begin{array}{c|c} {}^{k}L(x_0) & 0 \\ \hline {}^{k}w^T(x_0){}^{k}U^{-1}(x_0) & 1 \end{array}\right] \right\|_F \\ &\quad \leq \|{}^{k}L(x) - {}^{k}L(x_0)\|_F + \|{}^{k}w^T(x){}^{k}U^{-1}(x) - {}^{k}w^T(x_0){}^{k}U^{-1}(x_0)\|_2 \\ &\quad \leq c_k\|x - x_0\|_2 + \|{}^{k}w^T(x) - {}^{k}w^T(x_0)\|_2\|{}^{k}U^{-1}(x)\|_2 \\ &\qquad + \|{}^{k}w^T(x_0)\|_2\|{}^{k}U^{-1}(x) - {}^{k}U^{-1}(x_0)\|_2 \\ &\quad \leq \left\{ c_k + \left(\sum_{i=1}^{k} \gamma_{k+1,i}^2 \right)^{1/2} \cdot \sup_{x \in N(x_0, \varepsilon)} \|{}^{k}U^{-1}(x)\|_2 + \|{}^{k}w^T(x_0)\|_2 \hat{d}_k \right\} \cdot \|x - x_0\|_2 \\ &\quad \equiv c_{k+1}\|x - x_0\|_2. \end{aligned} $$

And,

$$ \begin{aligned} &\|{}^{k+1}U(x) - {}^{k+1}U(x_0)\|_F \\ &\qquad = \left\| \left[\begin{array}{c|c} {}^{k}U(x) & {}^{k}L^{-1}(x){}^{k}v(x) \\ \hline 0 & \alpha_{k+1}(x) \end{array}\right] - \left[\begin{array}{c|c} {}^{k}U(x_0) & {}^{k}L^{-1}(x_0){}^{k}v(x_0) \\ \hline 0 & \alpha_{k+1}(x_0) \end{array}\right] \right\|_F \\ &\qquad \leq \|{}^{k}U(x) - {}^{k}U(x_0)\|_F + \|{}^{k}L^{-1}(x){}^{k}v(x) - {}^{k}L^{-1}(x_0){}^{k}v(x_0)\|_2 \\ &\qquad\quad + |\alpha_{k+1}(x) - \alpha_{k+1}(x_0)|. \end{aligned} $$

Applying the triangle inequality to the second term yields

$$ \begin{aligned} &\|{}^{k}L^{-1}(x){}^{k}v(x) - {}^{k}L^{-1}(x_0){}^{k}v(x_0)\|_2 \\ &\qquad \leq \|{}^{k}L^{-1}(x)[{}^{k}v(x) - {}^{k}v(x_0)]\|_2 + \|[{}^{k}L^{-1}(x) - {}^{k}L^{-1}(x_0)]{}^{k}v(x_0)\|_2 \\ &\qquad \leq \left[\sup_{x \in N(x_0, \varepsilon)} \|{}^{k}L^{-1}(x)\|_2 \left(\sum_{i=1}^{k} \gamma_{i,k+1}^2 \right)^{1/2} + \hat{c}_k\|{}^{k}v(x_0)\|_2 \right] \|x - x_0\|_2 \\ &\qquad \equiv \beta_k\|x - x_0\|_2. \end{aligned} $$

Similarly, the third term on the right-hand side can be estimated:

$$
\begin{aligned}
&|\alpha_{k+1}(x) - \alpha_{k+1}(x_0)| \\
&\quad \leqslant |a_{k+1,k+1}(x) - a_{k+1,k+1}(x_0)| \\
&\qquad + |{}^k w^T(x)^k U^{-1}(x)^k L^{-1}(x)^k v(x) - {}^k w^T(x_0)^k U^{-1}(x_0)^k L^{-1}(x_0)^k v(x_0)| \\
&\quad \leqslant \gamma_{k+1,k+1}\|x - x_0\|_2 + |\left({}^k w^T(x) - {}^k w^T(x_0)\right)^k U^{-1}(x)^k L^{-1}(x)^k v(x)| \\
&\qquad + |{}^k w^T(x_0)\left[{}^k U^{-1}(x) - {}^k U^{-1}(x_0)\right]^k L^{-1}(x)^k v(x)| \\
&\qquad + |{}^k w^T(x_0)^k U^{-1}(x_0)\left[{}^k L^{-1}(x) - {}^k L^{-1}(x_0)\right]^k v(x)| \\
&\qquad + |{}^k w^T(x_0)^k U^{-1}(x_0)^k L^{-1}(x_0)\left({}^k v(x) - {}^k v(x_0)\right)| \\
&\quad \leqslant \Bigg\{ \gamma_{k+1,k+1} + \left(\sum_{i=1}^{n} \gamma_{k+1,i}^2\right)^{1/2} \sup_{x \in N(x_0,\varepsilon)} \|{}^k U^{-1}(x)\|_2 \|{}^k L^{-1}(x)\|_2 \|{}^k v(x)\|_2 \\
&\qquad\quad + \|{}^k w(x_0)\|_2 \hat{d}_k \sup_{x \in N(x_0,\varepsilon)} \|{}^k L^{-1}(x)\|_2 \|{}^k v(x)\|_2 \\
&\qquad\quad + \|{}^k w(x_0)\|_2 \|{}^k U^{-1}(x_0)\|_2 \hat{c}_k \sup_{x \in N(x_0,\varepsilon)} \|{}^k v(x)\|_2 \\
&\qquad\quad + \|{}^k w(x_0)\|_2 \|{}^k U^{-1}(x_0)\|_2 \|{}^k L^{-1}(x_0)\|_2 \left(\sum_{i=1}^{k} \gamma_{i,k+1}^2\right)^{1/2} \Bigg\} \\
&\qquad \cdot \|x - x_0\|_2 \equiv \delta_k \|x - x_0\|_2 .
\end{aligned}
$$

Set $d_{k+1} = d_k + \beta_k + \delta_k$.

By the Inverse Function Theorem, ${}^{k+1}L^{-1}(x)$ and ${}^{k+1}U^{-1}(x)$ are Lipschitz continuous on $\bar{N}(x_0, \varepsilon)$ with Lipschitz constants $\hat{c}_{k+1}$ and $\hat{d}_{k+1}$ respectively. The induction is complete. Define $c_0 = c_n$ and $d_0 = d_n$.

The proof of Theorem 3.9 is rather long and detailed. It consists of three main portions: first establishing that $P_0 J(x_*)$ can be factored without pivoting, proving local linear convergence by a Kantorovich analysis, and finally establishing the (m-step) Q-superlinear convergence. The second portion is subdivided further in the text.

Proof of Theorem 3.9. Part 1: $P_0 J(x_*)$ can be factored without pivoting.

Let $\varepsilon_1 > 0$ such that $N(x_*, \varepsilon_1) \subset D_0$. Set $\gamma \geqslant \|J_*^{-1}\|_2$ and define $\kappa = \|\Gamma\|_F$. By Lemma 3.6, given a pivoting strategy P_0, there exists an $\varepsilon_0 \in (0, \varepsilon_1]$ such that: if the LU decomposition without pivoting of $P_0 J(x_0)$ exists at $x_0 \in N(x_*, \varepsilon_0)$, then $P_0 J(x_*)$ can be factored without pivoting, and hence $P_0 J(x)$ can be factored without pivoting, for all $x \in N(x_*, \varepsilon_0)$. Let $P_0 J(x) = L(x)U(x)$. Furthermore,

$$\|L(x) - L(x_*)\|_F \leqslant c_0 \|x - x_*\|_2$$

for all $x \in N(x_*, \varepsilon)$.

Part 2: Local linear convergence.

(a) Select constants. Now choose ε in $(0, \varepsilon_0)$, $\delta > 0$, and $r \in (0, 1)$ such that

$$\text{(A.1)} \qquad \alpha_2 \varepsilon \left(\frac{1 - r^{m-1}}{1 - r}\right) < \delta \quad \text{and} \quad \gamma(1 + r)[\kappa\varepsilon + 2\eta\bar{\eta}\hat{\eta}\delta] \leqslant r(1 - r),$$

where $\alpha_2 = \alpha_1 + (m-1)\alpha_0$, and $\alpha_0 = \bar{\eta}\|U_*\|_2 c_0(2+\sqrt{n}\,\beta)$, $\alpha_1 = \sqrt{n}\,\bar{\eta}\beta\kappa$, and

$$\hat{\eta} = \sup_{x\in\bar{N}(x_*,\varepsilon_0)} \|L(x)\|_2 \quad \text{and} \quad \bar{\eta} = \max\left\{1, \sup_{x\in\bar{N}(x_*,\varepsilon_0)} \|L^{-1}(x)\|_2\right\}.$$

(b) Establish upper bound on $\|J_0^{-1}\|_2$. Further restrict ε so that $\|J(x) - J_*\|_F < \delta$ whenever $\|x - x_*\|_2 < \varepsilon$. Let $x_0 \in N(x_*, \varepsilon)$, then

$$\|L(x_0) - L(x_*)\|_F \leq c_0\|x_0 - x_*\|_2.$$

If $\|J_0 - J_*\|_F < \delta$, then $\|J_0 - J_*\|_2 < \eta\delta < 2\eta\delta$; by definition of $\hat{\eta}$ and $\bar{\eta}$, we have $\bar{\eta}\hat{\eta} \geq 1$ and thus from (A.1), $\gamma(1+r)2\eta\delta \leq r$. Therefore the Banach Lemma [16] gives

$$\|J_0^{-1}\|_2 \leq \frac{(1+r)}{(1-r)}\gamma.$$

(c) A double induction. The algorithm requires a double induction because of the restart criterion. Therefore we index the Doolittle updating algorithm as follows. Let $p = i \cdot m$, $i = 0, 1, \ldots$, and $k \in \{p, p+1, \ldots, p+(m-1)\}$, and

$$x_{k+1} = x_k - U_k^{-1}L_p^{-1}F_k$$

with $\{L_p\}$ and $\{U_p\}$ determined in step (5) and $\{U_k\}_{k=p+1}^{p+(m-1)}$ determined in step (7). Then

$$\begin{aligned} x_{k+1} - x_* = x_k - x_* &- U_k^{-1}L_p^{-1}(F_k - F_*) \\ &+ U_k^{-1}L_p^{-1}J_*(x_k - x_*) - U_k^{-1}L_p^{-1}J_*(x_k - x_*) \end{aligned}$$

and

$$\begin{aligned} \|x_{k+1} - x_*\|_2 \leq \|U_k^{-1}L_p^{-1}\|_2 \big[&\|F_k - F_* - J_*(x_k - x_*)\|_2 \\ &+ \|L_pU_k - J_*\|_2\|x_k - x_*\|_2\big]. \end{aligned} \tag{A.2}$$

The induction will show that

$$\|x_{n+1} - x_*\|_2 \leq r\|x_n - x_*\|_2$$

for $0 \leq r < 1$ and for any n, and thus establish the local convergence result.

(1) Step 1 of induction. For $p = 0$, $k = 0$, (A.2) becomes

$$\begin{aligned} \|x_1 - x^*\|_2 &\leq \|J_0^{-1}\|_2\big[(\kappa\varepsilon + 2\eta\delta)\|x_0 - x_*\|_2\big] \\ &\leq (1+r)\gamma(\kappa\varepsilon + 2\eta\delta)\|x_0 - x_*\|_2 \leq r\|x_0 - x_*\|_2. \end{aligned}$$

Assume, for $p = im$, $i = 0, 1, \ldots, j$, that

$$\|x_{jm} - x_*\| \leq r\|x_{jm-1} - x_*\|_2.$$

Then $x_{jm} \in N(x_*, \varepsilon)$, and it follows that $\|J_{jm} - J_*\|_F < \delta$ and $\|J_{jm} - J_*\|_2 < 2\eta\delta$. From (A.1), $2\gamma(1+r)\eta\delta \leq r$, and the Banach lemma gives

$$\|J_{jm}^{-1}\|_2 \leq \frac{(1+r)}{(1-r)}\gamma.$$

(2) Induction on p. For $p = jm$, let $k = p$. Then (A.2) is

$$\|x_{jm+1} - x_*\|_2 \leq \|J_{jm}^{-1}\|_2\left[(\kappa\varepsilon + 2\eta\delta)\|x_{jm} - x_*\|_2\right]$$
$$\leq \frac{(1+r)}{(1-r)}\gamma(\kappa\varepsilon + 2\eta\delta)\|x_{jm} - x_*\|_2 \leq r\|x_{jm} - x_*\|_2.$$

(3) Induction on k. Assume, for $l = p, p+1, \ldots, k-1$, $k \leq p + (m-1)$, that

$$\|x_{l+1} - x_*\|_2 \leq r\|x_l - x_*\|_2.$$

Now,

$$\|L_pU_{l+1} - J_*\|_{L_p^{-1},F} = \|L_pU_{l+1} - L_pU_* + L_pU_* - L_*U_*\|_{L_p^{-1},F}$$
$$\leq \|U_{l+1} - U_*\|_F + \|L_p^{-1}\|_2\|L_p - L_*\|_F\|U_*\|_2$$
$$\leq \|U_{l+1} - U_*\|_F + \|L_p^{-1}\|_2\sqrt{n}\,\beta\left[\kappa\sigma_l + \|U_*\|_2 c_0\|x_p - x_*\|_2\right]$$
$$+ \|L_p^{-1}\|_2\|U_*\|_2\|L_p - L_*\|_F$$

by Lemma 3.6 and the fact that: for $a, b, c > 0$, $c^2 \leq a^2 + b^2$ implies $c \leq a + b$. Since $\|L_p^{-1}\|_2 \leq \bar{\eta}$,

$$\|L_pU_{l+1} - J_*\|_{L_p^{-1},F} \leq \|L_pU_l - L_*U_*\|_{L_p^{-1},F} + 2\bar{\eta}\|U_*\|_2\|L_p - L_*\|_F$$
$$+ \alpha_1\sigma_l + \bar{\eta}\sqrt{n}\,\beta\|U_*\|_2 c_0\|x_l - x_*\|_2$$
$$\leq \|L_pU_l - L_*U_*\|_{L_p^{-1},F} + \alpha_1\sigma_l + \alpha_0\|x_l - x_*\|_2.$$

Therefore,

$$\|L_pU_{l+1} - L_*U_*\|_{L_p^{-1},F} \leq \|L_pU_p - L_*U_*\|_{L_p^{-1},F} + \alpha_1\sum_{i=p}^{l}\sigma_i + \alpha_0(l+1)\|x_p - x_*\|_2$$
$$\leq \alpha_2\sum_{i=p}^{l}\sigma_i + \|L_pU_p - L_*U_*\|_{L_p^{-1},F},$$

and, for $l = k - 1$,

$$\|L_pU_k - L_*U_*\|_{L_p^{-1},F} \leq \|L_pU_p - L_*U_*\|_{L_p^{-1},F} + \alpha_2\sum_{i=p}^{k-1} r^i\varepsilon$$
$$\leq \|L_pU_p - L_*U_*\|_{L_p^{-1},F} + \alpha_2\left(\frac{1 - r^{m-1}}{1-r}\right)r^p \leq \bar{\eta}\delta + \delta \leq 2\bar{\eta}\delta.$$

Thus,

$$\|L_pU_{l+1} - J_*\|_2 \leq \eta\|L_pU_{l+1} - J_*\|_F \leq \eta\hat{\eta}\|L_pU_{l+1} - J_*\|_{L_p^{-1},F},$$

and this gives

$$\|U_k^{-1}L_p^{-1}\|_2 \leq \frac{(1+r)}{(1-r)}\gamma,$$

which with (A.2) yields

$$\|x_{k+1} - x_*\|_2 \leq \frac{(1+r)}{(1-r)}\gamma[\kappa\varepsilon + 2\eta\bar{\eta}\hat{\eta}\delta]\|x_k - x_*\|_2 \leq r\|x_k - x_*\|_2 \quad \text{by (A.1).}$$

(4) Conclusion of local convergence. For $l = p + (m - 1)$, (A.2) is

$$\|x_{p+m} - x_*\|_2 \leq r\|x_{p+(m-1)} - x_*\|_2.$$

So $x_{p+m} \in N(x_*, \varepsilon)$ and $\|J_{p+m} - J_*\|_F < \delta$. This establishes the local and linear convergence of the Doolittle updating algorithm.

Part 3: m-step Q-superlinear convergence. Q-superlinear convergence then follows from Theorem 3.1 of [6]. Since $p = im$, $i = 0, 1, \ldots$,

$$\frac{\|(J_p - J_*)(x_{p+1} - x_p)_2\|}{\|x_{p+1} - x_p\|_2} \leq \|J_p - J_*\|_2 \underset{p\to\infty}{\to} 0,$$

it follows that

$$\frac{\|x_{p+1} - x_*\|}{\|x_p - x_*\|} \underset{p\to\infty}{\to} 0.$$

Mathematical Sciences Department
Rice University
Houston, Texas 77001

Center for Applied Mathematics
Cornell University
Ithaca, New York 14853

1. K. W. BRODLIE, A. R. GOURLAY & J. GREENSTADT, "Rank-one and rank-two corrections to positive definite matrices expressed in product form," *J. Inst. Math. Appl.*, v. 11, 1973, pp. 73–82.

2. C. G. BROYDEN, "The convergence of an algorithm for solving sparse nonlinear systems," *Math. Comp.*, v. 25, 1971, pp. 285–294.

3. A. K. CLINE, C. B. MOLER, G. W. STEWART & J. H. WILKINSON, "An estimate for the condition number of a matrix," *SIAM J. Numer. Anal.*, v. 16, 1979, pp. 368–375.

4. A. R. CURTIS, M. J. D. POWELL & J. K. REID, "On the estimation of sparse Jacobian matrices," *J. Inst. Math. Appl.*, v. 13, 1974, pp 117–119.

5. J. E. DENNIS, JR., "A brief introduction to quasi-Newton methods" in *Numerical Analysis* (Edited by G. Golub and J. Oliger), Proc. Sympos. Appl. Math., vol. 22, Amer. Math. Soc., Providence, R. I., 1978.

6. J. E. DENNIS, JR. & J. J. MORÉ, "A characterization of superlinear convergence and its application to quasi-Newton methods," *Math. Comp.*, v. 28, 1974, pp. 549–560.

7. J. E. DENNIS, JR & J. J. MORÉ, "Quasi-Newton methods, motivation and theory," *SIAM Rev.*, v. 19, 1977, pp. 46–89.

8. J. E. DENNIS, JR. & H. F. WALKER, "Convergence theorems for least-change secant update methods," *SIAM J. Numer. Anal.*, v. 18, 1981, pp 949–987.

9. J. J. DONGARRA, J. R. BUNCH, C. B. MOLER & G. W. STEWART, *LINPACK User's Guide*, *SIAM*, Philadelphia, Pa., 1979.

10. I. S. DUFF, "A survey of sparse matrix research," *Proc. IEEE*, v. 65, 1977, pp. 500–535.

11. A. M. ERISMAN & J. K. REID, "Monitoring the stability of the triangular factorization of a sparse matrix," *Numer. Math.*, v. 22, 1974, pp. 183–186.

12. P. E. GILL, G. GOLUB, W. MURRAY & M. A. SAUNDERS, "Methods for modifying matrix factorizations," *Math. Comp.*, v. 28, 1974, pp. 505–535.

13. A. HINDMARSH, private communication, 1978.

14. A. LUCIA, Thesis, Dept. of Chem. Engr., University of Connecticut, Storrs, 1980.

15. J. J. MORÉ, B. GARBOW & K. HILLSTROM, *User Guide for MINPACK-1*, Report ANL-80-74, Argonne National Laboratories, 1980.

16. J. M. ORTEGA & W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

17. L. K. SCHUBERT, "Modification of a quasi-Newton method for nonlinear equations with a sparse Jacobian," *Math. Comp.*, v. 24, 1970, pp. 27–30.

18. C. VAN LOAN, private communication, 1978.

19. J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.

20. M. WHEELER & T. POTEMPA, private communication, 1980.