Modification of a Quasi-Newton Method for Nonlinear Equations with a Sparse Jacobian*

By L. K. Schubert

Abstract. For solving large systems of nonlinear equations by quasi-Newton methods it may often be preferable to store an approximation to the Jacobian rather than an approximation to the inverse Jacobian. The main reason is that when the Jacobian is sparse and the locations of the zeroes are known, the updating procedure can be made more efficient for the approximate Jacobian than for the approximate inverse Jacobian.

I. Introduction. In recent years a class of methods termed quasi-Newton have received considerable attention in the literature [1]-[3]. In one such method [1], [4] iterative approximations to the solution of the system of equations

\[ f(x) = 0 \]

where \( f \) and \( x \) are \( n \)-vectors, are obtained by solving

\[ G^{(k)} p^{(k)} = -f^{(k)} \]

and substituting \( p^{(k)} \) in

\[ x^{(k+1)} = x^{(k)} + t^{(k)} p^{(k)} . \]

The scalar \( t^{(k)} \) is chosen to reduce some norm of \( f \) at each step, thus ensuring stability. The approximation \( G^{(k)} \) to the Jacobian is revised after each step in accordance with

\[ G^{(k+1)} = G^{(k)} + \left[ \frac{f^{(k+1)} - (1 - t^{(k)}) f^{(k)}}{t^{(k)} p^{(k)T} p^{(k)}} \right] p^{(k)T} . \]

This is the result of a primary condition requiring \( G^{(k+1)} \) to predict the same changes in \( f \) in the direction \( p^{(k)} \) that actually occurred at the \( (k + 1) \)th step (supplies \( n \) equations) and a secondary condition requiring \( G^{(k+1)} \) to predict the same changes in \( f \) as \( G^{(k)} \) in all directions orthogonal to \( p^{(k)} \) (supplies the remaining \( n^2 - n \) equations).

Broyden [1] described a class of methods containing the above method as a special case. However, he suggested the use of an approximation \( H^{(k)} \) to the inverse Jacobian instead of \( G^{(k)} \). He supplied an explicit updating algorithm for \( H^{(k)} \) equivalent (in its simplest form) to (4). Other choices of the secondary con-

* The work presented in this note was supported by the National Research Council of Canada under Grant No. A2003 and by the Air Force Office of Scientific Research, United States Air Force, under AFOSR Grant No. 67-0672A.
dition on $G^{(k)}$ or $H^{(k)}$ are possible and this fact gives rise to the class of quasi-
Newton methods.

When $H^{(k)}$ is used, the solution of the linear equations (2) is reduced to the
matrix-vector multiplication

$$p^{(k)} = -H^{(k)}f^{(k)}.$$  

This is certainly an advantage when all elements of the Jacobian are unknown. However, in many large systems of nonlinear equations, particularly the difference
equations arising from nonlinear differential equations, most of the elements of the
Jacobian are known to be zero and other elements may be known nonzero const-
ants. If the known zeros are introduced into $G^{(k)}$, much less storage is required
for $G^{(k)}$ than for the full matrix $H^{(k)}$. Also, if $G^{(k)}$ has a band structure, (2) need not
be a great deal more time-consuming than (5). Moreover, when many of the ele-
ments of the Jacobian are known, the number of secondary conditions on $G^{(k+1)}$
can be greatly reduced, so that $G$ converges more rapidly to the Jacobian. This
requires a simple modification of (4) which will now be described.

II. Modification When the Jacobian is Sparse. The $i$th row $g_i^{(k)}$ of $G^{(k)}$ represents
an approximation to the gradient of the $i$th function component $f_i$. When $n - r_i$
components of $g_i$ are known constants, one first imposes the condition that these
components shall remain unchanged in the Jacobian revision; the remaining choices
have to be made on the basis of the remaining $r_i$ coordinate directions.

Designate by $\hat{p}^{(k)}$ the column vector derived from $p^{(k)}$ by setting $p_{ji}^{(k)}$ to zero
whenever the corresponding element of $g_{ij}$ is a known constant. Note that $\hat{p}^{(k)}$ is
dependent on $i$. Also let $\bar{g}_i$ be the row vector derived from $g_i$ by setting its unknown
elements to zero.

The known components of $g_i$ account for a change $t_i^{(k)}\bar{g}_i\hat{p}^{(k)}$ in $f_i$ at the $(k + 1)$st
step. The remainder of the change, $f_i^{(k+1)} - f_i^{(k)} - t_i^{(k)}\bar{g}_i\hat{p}^{(k)}$, must then be at-
tributed to the unknown components. Thus the primary condition on $g_i^{(k+1)}$
restricted to $r_i$-space, becomes

$$t_i^{(k)}g_i^{(k+1)}\hat{p}^{(k)} = f_i^{(k+1)} - f_i^{(k)} - t_i^{(k)}\bar{g}_i\hat{p}^{(k)}, \quad i = 1, 2, \ldots, n.$$  

This is, in fact, identical to the usual primary condition $t_i^{(k)}G^{(k+1)}p_i^{(k)} = f_i^{(k+1)} - f_i^{(k)}$
because $g_i^{(k+1)}\hat{p}^{(k)} + \bar{g}_i\hat{p}^{(k)} = g_i^{(k+1)}p_i^{(k)}$. The secondary condition is similarly
obtained by restricting the usual secondary condition to the $r_i$-space corresponding
to the unknown elements of $g_i$:

$$g_i^{(k+1)}\hat{q} = g_i^{(k)}\hat{q}, \quad i = 1, 2, \ldots, n$$

where $\hat{q}$ satisfies $\hat{p}^{(k)}\hat{q} = 0$. This does not reduce to the usual condition $G^{(k+1)}q = G^{(k)}q$.

It is easily verified that (6) and (7) are satisfied by the exact row-by-row ana-
logue of (4), i.e.,

$$g_i^{(k+1)} = g_i^{(k)} + \frac{[f_i^{(k+1)} - (1 - t_i^{(k)})f_i^{(k)}]\hat{p}^{(k)}T}{t_i^{(k)}\hat{p}^{(k)}T\hat{p}^{(k)}}, \quad i = 1, 2, \ldots, n.$$
III. Example. A set of equations used by Broyden to test his methods is

\[ f_1 = -(3 + \alpha x_1)x_1 + 2x_2 - \beta , \]

\[ f_i = x_{i-1} - (3 + \alpha x_i)x_i + 2x_{i+1} - \beta , \quad i = 2, 3, \ldots, n - 1 , \]

\[ f_n = x_{n-1} - (3 + \alpha x_n)x_n - \beta . \]

These equations are also suitable for illustrating the present variant, if the zero entries in the Jacobian are regarded as known. The parameter values chosen were \( \alpha = -0.5; \beta = 1; n = 5, 10, 20; x_i^{(0)} = -1 \) for all \( i \). Both the unit matrix and a difference approximation based on a differencing interval of 0.001 were used for \( G^{(0)} \). Broyden's mean convergence rate

\[ R = \frac{1}{m} \ln \frac{N_1}{N_m} , \]

where \( N_1 \) and \( N_m \) are the initial and final Euclidean norms of \( f \), was computed in each case. \( m \) has been redefined as the total number of function component evaluations divided by \( n \). In this way \( m \) reflects the fact that one can take advantage of the Jacobian's sparseness in computing \( G^{(0)} \) by differencing.

Results for the present method ("modified Jacobian revision"), Broyden's 1/fsr method ("basic Jacobian revision"), and the constant matrix method ("no Jacobian revision") are shown in Tables I — III.

### Table I. \( n = 5 \)

<table>
<thead>
<tr>
<th>Nature of ( G^{(0)} )</th>
<th>Method</th>
<th>( N_1 )</th>
<th>( N_m )</th>
<th>( m )</th>
<th>( R )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Difference Approximation to Jacobian</td>
<td>Mod. Jac. rev.</td>
<td>1.803</td>
<td>9.592 × 10^{-7}</td>
<td>8</td>
<td>1.901</td>
</tr>
<tr>
<td></td>
<td>Basic Jac. rev.</td>
<td>1.803</td>
<td>9.657 × 10^{-8}</td>
<td>9</td>
<td>1.947</td>
</tr>
<tr>
<td></td>
<td>No Jac. rev.</td>
<td>1.803</td>
<td>2.149 × 10^{-7}</td>
<td>11</td>
<td>1.504</td>
</tr>
<tr>
<td></td>
<td>Mod. Jac. rev.</td>
<td>1.803</td>
<td>3.272 × 10^{-7}</td>
<td>20</td>
<td>0.776</td>
</tr>
<tr>
<td></td>
<td>Basic Jac. rev.</td>
<td>1.803</td>
<td>7.262 × 10^{-7}</td>
<td>23</td>
<td>0.640</td>
</tr>
<tr>
<td></td>
<td>No Jac. rev.</td>
<td>1.803</td>
<td>5.920 × 10^{-7}</td>
<td>73</td>
<td>0.205</td>
</tr>
</tbody>
</table>

### Table II. \( n = 10 \)

<table>
<thead>
<tr>
<th>Nature of ( G^{(0)} )</th>
<th>Method</th>
<th>( N_1 )</th>
<th>( N_m )</th>
<th>( m )</th>
<th>( R )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Difference Approximation to Jacobian</td>
<td>Mod. Jac. rev.</td>
<td>2.121</td>
<td>1.408 × 10^{-7}</td>
<td>9</td>
<td>1.878</td>
</tr>
<tr>
<td></td>
<td>Basic Jac. rev.</td>
<td>2.121</td>
<td>2.098 × 10^{-7}</td>
<td>11</td>
<td>1.493</td>
</tr>
<tr>
<td></td>
<td>No Jac. rev.</td>
<td>2.121</td>
<td>5.404 × 10^{-7}</td>
<td>15</td>
<td>1.026</td>
</tr>
<tr>
<td></td>
<td>Mod. Jac. rev.</td>
<td>2.121</td>
<td>1.707 × 10^{-7}</td>
<td>26</td>
<td>0.628</td>
</tr>
<tr>
<td></td>
<td>Basic Jac. rev.</td>
<td>2.121</td>
<td>4.391 × 10^{-7}</td>
<td>61</td>
<td>0.252</td>
</tr>
<tr>
<td></td>
<td>No Jac. rev.</td>
<td>2.121</td>
<td>8.363 × 10^{-7}</td>
<td>88</td>
<td>0.168</td>
</tr>
</tbody>
</table>

The results indicate that modified Jacobian revision becomes increasingly desirable as \( n \) is increased, particularly if the initial approximation to the Jacobian is poor.

The modification may also be useful when the Jacobian is full, but most of the...
Table III. \( n = 20 \)

<table>
<thead>
<tr>
<th>Nature of ( G^{(0)} )</th>
<th>Method</th>
<th>( N_1 )</th>
<th>( N_m )</th>
<th>( m )</th>
<th>( R )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Difference</td>
<td>Mod. Jac. rev.</td>
<td>2.646</td>
<td>3.130 ( \times 10^{-7} )</td>
<td>9</td>
<td>1.792</td>
</tr>
<tr>
<td>Approximation to Jacobian</td>
<td>Basic Jac. rev.</td>
<td>2.646</td>
<td>3.846 ( \times 10^{-7} )</td>
<td>12</td>
<td>1.323</td>
</tr>
<tr>
<td></td>
<td>No Jac. rev.</td>
<td>2.646</td>
<td>3.473 ( \times 10^{-7} )</td>
<td>19</td>
<td>0.838</td>
</tr>
<tr>
<td>Unit</td>
<td>Mod. Jac. rev.</td>
<td>2.646</td>
<td>3.402 ( \times 10^{-7} )</td>
<td>25</td>
<td>0.635</td>
</tr>
<tr>
<td>Matrix</td>
<td>Basic Jac. rev.</td>
<td>2.646</td>
<td>9.850 ( \times 10^{-7} )</td>
<td>118</td>
<td>0.125</td>
</tr>
<tr>
<td></td>
<td>No Jac. rev.</td>
<td>2.646</td>
<td>9.222 ( \times 10^{-7} )</td>
<td>97</td>
<td>0.153</td>
</tr>
</tbody>
</table>

Entries are easily computed constants. In this case, however, storage space is not economized and the solution of (2) may be time-consuming.