A REDUCED HESSIAN METHOD FOR LARGE-SCALE CONSTRAINED OPTIMIZATION

by

Lorenz T. Biegler,[1] *Jorge Nocedal,*[2] *and Claudia Schmid*[1]

[1] Chemical Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213. Partial support for these authors is acknowledged from the Engineering Design Research Center, an NSF-sponsored Engineering Research Center at Carnegie Mellon University.

[2] Department of Electrical Engineering and Computer Science, Northwestern University, Evanston IL 60208, nocedal@eecs.nwu.edu; also, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439. This author was supported by National Science Foundation grants CCR-9101359 and ASC-9213149, by U.S. Department of Energy grant DE-FG02-87ER25047-A004, and by the Office of Scientific Computing, U.S. Department of Energy, under Contract W-31-109-Eng-38.

A REDUCED HESSIAN METHOD FOR LARGE-SCALE CONSTRAINED OPTIMIZATION

by

Lorenz T. Biegler, Jorge Nocedal, and Claudia Schmid

ABSTRACT

We propose a quasi-Newton algorithm for solving large optimization problems with nonlinear equality constraints. It is designed for problems with few degrees of freedom and is motivated by the need to use sparse matrix factorizations. The algorithm incorporates a correction vector that approximates the cross term $Z^TWYp_{\rm Y}$ in order to estimate the curvature in both the range and null spaces of the constraints. The algorithm can be considered to be, in some sense, a practical implementation of an algorithm of Coleman and Conn. We give conditions under which local and superlinear convergence is obtained.

Key words: successive quadratic programming, reduced Hessian methods, constrained optimization, quasi-Newton method, large-scale optimization.

Abbreviated title: A Reduced Hessian Method

1. Introduction

We consider the nonlinear optimization problem

$$\min_{x\in\mathbf{R}^n} f(x) \tag{1.1}$$

$$\text{subject to } \ c(x)=0, \tag{1.2}$$

where $f:\mathbf{R}^n\to\mathbf{R}$ and $c:\mathbf{R}^n\to\mathbf{R}^m$ are smooth functions. We are particularly interested in the case when the number of variables n is large, and the algorithm we propose, which is a variation of the successive quadratic programming method, is designed to be efficient in this case. We assume that the first derivatives of f and c are available, but our algorithm does not require second derivatives.

The successive quadratic programming (SQP) method for solving (1.1)–(1.2) generates, at an iterate x_k, a search direction d_k by solving

$$\min_{d\in\mathbf{R}^n} g(x_k)^Td+\frac{1}{2}d^TW(x_k)d \tag{1.3}$$

$$\text{subject to } \ c(x_k)+A(x_k)^Td=0, \tag{1.4}$$

where g denotes the gradient of f, W denotes the Hessian of the Lagrangian function $L(x,\lambda) = f(x) + \lambda^T c(x)$, and A denotes the $n \times m$ matrix of constraint gradients

$$A(x) = [\nabla c_1(x), ..., \nabla c_m(x)]. \tag{1.5}$$

A new iterate is then computed as

$$x_{k+1} = x_k + \alpha_k d_k, \tag{1.6}$$

where α_k is a steplength parameter chosen so as to reduce the value of the merit function. In this study we will use the ℓ_1 merit function

$$\phi_\mu(x) = f(x) + \mu \|c(x)\|_1, \tag{1.7}$$

where μ is a penalty parameter; see, for example, Conn (1973), Han (1977) or Fletcher (1987). We could have used other merit functions, but the essential points we wish to convey in this article are not dependent upon the particular choice of the merit function.

The solution of the quadratic program (1.3)–(1.4) can be written in a simple form if we choose a suitable basis of $\mathbf{R}^n$ to represent the search direction d_k. For this purpose, we introduce a nonsingular matrix of dimension n, which we write as

$$[Y_k \; Z_k], \tag{1.8}$$

where $Y_k \in \mathbf{R}^{n \times m}$ and $Z_k \in \mathbf{R}^{n \times (n-m)}$, and we assume that

$$A_k^T Z_k = 0. \tag{1.9}$$

(From now on we abbreviate $A(x_k)$ as A_k, $g(x_k)$ as g_k, etc.) Thus Z_k is a basis for the tangent space of the constraints. We can now express d_k, the solution to (1.3)–(1.4), as

$$d_k = Y_k p_{\mathrm{Y}} + Z_k p_{\mathrm{Z}}, \tag{1.10}$$

for some vectors $p_{\mathrm{Y}} \in \mathbf{R}^m$ and $p_{\mathrm{Z}} \in \mathbf{R}^{n-m}$. Due to (1.9) the linear constraints (1.4) become

$$c_k + A_k^T Y_k p_{\mathrm{Y}} = 0. \tag{1.11}$$

If we assume that A_k has full column rank, then the nonsingularity of $[Y_k \; Z_k]$ and equation (1.9) imply that the matrix $A_k^T Y_k$ is nonsingular, so that p_{Y} is determined by (1.11):

$$p_{\mathrm{Y}} = -[A_k^T Y_k]^{-1} c_k. \tag{1.12}$$

Substituting this in (1.10), we have

$$d_k = -Y_k [A_k^T Y_k]^{-1} c_k + Z_k p_{\mathrm{Z}}. \tag{1.13}$$

Note that

$$Y_k [A_k^T Y_k]^{-1} \tag{1.14}$$

is a right inverse of A_k^T and that the first term in (1.13) represents a particular solution of the linear equations (1.4).

We have thus reduced the size of the SQP subproblem, which can now be expressed exclusively in terms of the variables p_{Z}. Indeed, substituting (1.10) into (1.3), considering $Y_k p_{\mathrm{Y}}$ as constant, and ignoring constant terms, we obtain the unconstrained quadratic problem

$$\min_{p_{\mathrm{Z}} \in \mathbf{R}^{n-m}} \; (Z_k^T g_k + Z_k^T W_k Y_k p_{\mathrm{Y}})^T p_{\mathrm{Z}} + \frac{1}{2} {p_{\mathrm{Z}}}^T (Z_k^T W_k Z_k) p_{\mathrm{Z}}. \tag{1.15}$$

If we assume that $Z_k^T W_k Z_k$ is positive definite, the solution of (1.15) is

$$p_{\rm Z} = -(Z_k^T W_k Z_k)^{-1}[Z_k^T g_k + Z_k^T W_k Y_k p_{\rm Y}]. \tag{1.16}$$

This determines the search direction of the SQP method.

We are particularly interested in the class of problems in which the number of variables n is large, but $n-m$ is small. In this case it is practical to approximate $Z_k^T W_k Z_k$ using a variable metric formula such as BFGS. On the other hand, the matrix $Z_k^T W_k Y_k$, of dimension $(n-m)\times m$ may be too expensive to compute directly when m is large. For this reason several authors simply ignore the "cross term" $Z_k^T W_k Y_k p_{\rm Y}$ in (1.16) and compute only an approximation to the reduced Hessian $Z_k^T W_k Z_k$; see Coleman and Conn (1984), Nocedal and Overton (1985), and Xie (1991). This approach is quite adequate when the basis matrices Y_k and Z_k in (1.8) are chosen to be orthonormal (Gurwitz and Overton (1989)).

For large problems, however, computing orthogonal bases can be expensive, and it is more efficient to obtain Y_k and Z_k by simple elimination of variables (cf. Fletcher (1987)). Unfortunately, in this case ignoring the cross term $Z_k^T W_k Y_k p_{\rm Y}$ can make the algorithm inefficient, as is illustrated by an example given in a companion paper (Biegler, Nocedal, and Schmid (1993)). The central point is that the range space component $Y_k p_{\rm Y}$ may be very large, and ignoring the contribution from the cross term in (1.16) can result in a poor step.

Therefore, in this paper we suggest ways of approximating the cross term $Z_k^T W_k Y_k p_{\rm Y}$ by a vector w_k,

$$[Z_k^T W_k Y_k]p_{\rm Y} \approx w_k, \tag{1.17}$$

without computing the matrix $Z_k^T W_k Y_k$. We consider two approaches for calculating w_k; the first involves an approximation to the matrix $[Z_k^T W_k Y_k]$ using Broyden's update, and the second generates w_k using finite differences. We will show that the rate of convergence of the new algorithm is 1-step Q-superlinear, as opposed to the 2-step superlinear rate for methods that ignore the cross term (Byrd (1985) and Yuan (1985)). The null space step (1.16) of our algorithm will be given by

$$p_{\rm Z} = -(Z_k^T W_k Z_k)^{-1}[Z_k^T g_k + \zeta_k w_k], \tag{1.18}$$

where $0 < \zeta_k \le 1$ is a damping factor to be discussed later on.

To describe our first strategy for computing the vector w_k, we consider a quasi-Newton method in which the rectangular matrix $Z_k^T W_k$ is approximated by a matrix S_k, using Broyden's method. We then obtain w_k by multiplying this matrix by $Y_k p_{\rm Y}$, that is,

$$w_k = S_k Y_k p_{\rm Y}.$$

How should S_k be updated? Since $W_{k+1} = \nabla_{xx}^2 L(x_{k+1}, \lambda_{k+1})$, we have that

$$Z_k^T W_{k+1}(x_{k+1} - x_k) \approx Z_k^T[\nabla_x L(x_{k+1}, \lambda_{k+1}) - \nabla_x L(x_k, \lambda_{k+1})], \tag{1.19}$$

when x_{k+1} is close to x_k. We use this relation to establish the following secant equation: we demand that S_{k+1} satisfy

$$S_{k+1}(x_{k+1} - x_k) = Z_k^T[\nabla_x L(x_{k+1}, \lambda_{k+1}) - \nabla_x L(x_k, \lambda_{k+1})]. \tag{1.20}$$

One point in this derivation requires clarification. In the left-hand side of (1.19) we have $Z_k^T W_{k+1}$, and not $Z_{k+1}^T W_{k+1}$. We could have used Z_{k+1} in (1.19), avoiding an inconsistency of indices, but this is not necessary since we will show that using Z_k instead of Z_{k+1} in (1.20) results in algorithms with all the desirable properties. This fact will not be surprising to readers familiar with the analysis of SQP methods; see, for example, Coleman and Conn (1984) or Nocedal and Overton (1985). In addition, using Z_k allows updating of S_{k+1} and B_{k+1} prior to creating Z_{k+1} at the new point.

Let us now consider how to approximate the reduced Hessian matrix $Z_k^T W_k Z_k$. Using (1.6) and (1.10) in (1.20), we obtain

$$[S_{k+1} Z_k]\alpha_k p_{\mathrm{Z}} = -\alpha_k S_{k+1}(Y_k p_{\mathrm{Y}}) + Z_k^T[\nabla_x L(x_{k+1}, \lambda_{k+1}) - \nabla_x L(x_k, \lambda_{k+1})].$$

Since S_{k+1} approximates $Z_k^T W_k$, this suggests the following secant equation for B_{k+1}, the quasi-Newton approximation to the reduced Hessian $Z_k^T W_k Z_k$:

$$B_{k+1} s_k = y_k, \tag{1.21}$$

where s_k is defined by

$$s_k = \alpha_k p_{\mathrm{Z}},$$

and y_k by

$$y_k = Z_k^T[\nabla_x L(x_{k+1}, \lambda_{k+1}) - \nabla_x L(x_k, \lambda_{k+1})] - \overline{w}_k, \tag{1.22}$$

with

$$\overline{w}_k = \alpha_k S_{k+1}(Y_k p_{\mathrm{Y}}). \tag{1.23}$$

We will update B_k by the BFGS formula (cf. Fletcher (1987))

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}, \tag{1.24}$$

provided $s_k^T y_k$ is sufficiently positive.

We highlight a subtle but important point. We have defined two correction terms, w_k and $\overline{w}_k$. Both are approximations to the cross term $(Z^T W Y)p_{\mathrm{Y}}$. The first term, w_k, which is needed to define the null-space step (1.18) — and thus the new iterate x_{k+1} — makes use of the matrix S_k. The second term, $\overline{w}_k$, which is used in (1.22) to define the BFGS update of B_k, is computed by using the new Broyden matrix S_{k+1} and takes into account the steplength α_k. We will see below that it is useful to incorporate the most recent information in $\overline{w}_k$. Note that this requires the Broyden update to be applied before the vector y_k for the BFGS update can be calculated from (1.22).

The Lagrange multiplier estimates λ_k needed in the definition (1.22) of y_k are defined by

$$\lambda_k = -[Y_k^T A_k]^{-1} Y_k^T g_k. \tag{1.25}$$

This formula is motivated by the fact that, at a solution x_* of (1.1)–(1.2), we have $-g_* = A_* \lambda_*$, and since $Y_*[A_*^T Y_*]^{-1}$ is a right inverse of A_*^T,

$$\lambda_* = -[Y_*^T A_*]^{-1} Y_*^T g_*.$$

Using the same right inverse (1.14) in the definitions of p_{Y} and λ_k will allow us a convenient simplification in the formulae presented in the following sections. We stress, however, that other Lagrange multiplier estimates can be used and that the best choice in practice might be the one that involves the least computation or storage.

We can now outline the sequential quadratic programming method analyzed in this paper.

Algorithm I

1. Choose constants $\eta \in (0, 1/2)$ and τ, τ' with $0 < \tau < \tau' < 1$. Set $k := 1$, and choose a starting point x_1 and an $(n-m) \times (n-m)$ symmetric and positive definite starting matrix B_1.
2. Evaluate f_k, g_k, c_k, and A_k, and compute Y_k and Z_k.
3. Compute $p_{\rm Y}$ by solving the system
$$(A_k^T Y_k) p_{\rm Y} = -c_k. \qquad \text{(range space step)} \tag{1.26}$$
4. Compute an approximation w_k to $(Z_k^T W_k Y_k) p_{\rm Y}$.
5. Choose the damping parameter $\zeta_k \in (0, 1]$ and compute $p_{\rm Z}$ from
$$B_k p_{\rm Z} = -[Z_k^T g_k + \zeta_k w_k]. \qquad \text{(null space step)} \tag{1.27}$$
Define the search direction by
$$d_k = Y_k p_{\rm Y} + Z_k p_{\rm Z}. \tag{1.28}$$
6. Set $\alpha_k = 1$, and choose the weight μ_k of the merit function (1.7).
7. Test the line search condition
$$\phi_{\mu_k}(x_k + \alpha_k d_k) \le \phi_{\mu_k}(x_k) + \eta \alpha_k D\phi_{\mu_k}(x_k; d_k), \tag{1.29}$$
where $D\phi_{\mu_k}(x_k; d_k)$ is the directional derivative of the merit function ϕ in the direction d_k.
8. If (1.29) is not satisfied, choose a new $\alpha_k \in [\tau\alpha_k, \tau'\alpha_k]$ and go to (7); otherwise set
$$x_{k+1} = x_k + \alpha_k d_k. \tag{1.30}$$
9. Evaluate $f_{k+1}, g_{k+1}, c_{k+1}$, and A_{k+1}, and compute Y_{k+1} and Z_{k+1}.
10. Compute the Lagrange multiplier estimate
$$\lambda_{k+1} = -[Y_{k+1}^T A_{k+1}]^{-1} Y_{k+1}^T g_{k+1}. \tag{1.31}$$
Define $\overline{w}_k$ (as will be discussed in §3), and compute
$$s_k = \alpha_k p_{\rm Z} \tag{1.32}$$
and
$$y_k = Z_k^T [\nabla_x L(x_{k+1}, \lambda_{k+1}) - \nabla_x L(x_k, \lambda_{k+1})] - \overline{w}_k. \tag{1.33}$$
If the update criterion (to be discussed in §3.3) is satisfied, compute B_{k+1} by the BFGS formula (1.24); else set $B_{k+1} = B_k$.
11. Set $k := k + 1$, and go to (3).

The algorithm has been left in a very general form, but in the next sections we discuss all its aspects in detail. In §2 we consider the choice of the basis matrices Y_k and Z_k. In §3 we describe the calculation of the correction terms w_k and $\overline{w}_k$, the conditions under which BFGS updating takes place, the choice of the damping parameter ζ_k, and the procedure for updating the weight μ_k in the merit function. In §4 and §5 we analyze of the local behavior of the algorithm and show that the rate of convergence is at least R-linear. In §6 we present a superlinear convergence result, and some final remarks in §7 conclude the paper.

We now make a few comments about our notation. Throughout the paper, the vectors $p_{\rm Y}$ and $p_{\rm Z}$ are computed at x_k and could be denoted by ${p_{\rm Y}}^{(k)}$ and ${p_{\rm Z}}^{(k)}$, but we will normally omit the superscript for simplicity. The symbol $\|\cdot\|$ denotes the l_2 vector norm or the corresponding induced matrix norm. When using the l_1 or l_∞ norms we will indicate it explicitly by writing $\|\cdot\|_1$ or $\|\cdot\|_\infty$. A solution of problem (1.1) is denoted by x_*, and we define

$$e_k = x_k - x_* \quad \text{and} \quad \sigma_k = \max\{\|e_k\|, \|e_{k+1}\|\}. \tag{1.34}$$

Here, and for the rest of the paper, $\nabla L(x, \lambda)$ indicates the gradient of the Lagrangian with respect to x only.

2. The Basis Matrices

As long as Z_k spans the null space of A_k^T, and $[Y_k \; Z_k]$ is nonsingular, the choice of Y_k and Z_k is arbitrary. However, from the viewpoint of numerical stability and robustness of the algorithm it is desirable to define Y_k and Z_k to be orthonormal, that is,

$$\begin{aligned} Z(x)^T Z(x) &= I_{n-m} \\ Y(x)^T Y(x) &= I_m \\ Y(x)^T Z(x) &= 0. \end{aligned}$$

One way of obtaining these matrices is by forming the QR factorization of A. For large problems, however, computing this QR factorization is often too expensive. Therefore many researchers, including Gabay (1982), Gilbert (1991), Fletcher (1987), Murray and Prieto (1992), and Xie (1991), consider other, nonorthogonal choices of Y and Z. For example, if we partition x into m basic or dependent variables (which without loss of generality are assumed to be the first m variables) and $n-m$ nonbasic or control variables, we induce the partition

$$A(x)^T = [C(x) \; N(x)], \tag{2.1}$$

where the $m \times m$ basis matrix $C(x)$ is assumed to be nonsingular. We now define $Z(x)$ and $Y(x)$ to be

$$Z(x) = \begin{bmatrix} -C(x)^{-1} N(x) \\ I \end{bmatrix} \quad Y(x) = \begin{bmatrix} I \\ 0 \end{bmatrix}. \tag{2.2}$$

When $A(x)$ is large and sparse, a sparse LU decomposition of $C(x)$ can often be computed efficiently, and this approach will be considerably less expensive than the QR factorization of A. Note that from the assumed nonsingularity of $C(x)$ both $Y(x)$ and $Z(x)$ vary smoothly with x, provided the same partition of the variables is maintained. In our implementation of the new algorithm (Biegler, Nocedal, and Schmid (1993)) we choose Y_k and Z_k by (2.2).

There is a price to pay for using nonorthogonal bases. If the matrix C is ill conditioned (and this can be difficult to detect), the step computation may be inaccurate. Moreover, even if the basis is well conditioned, the range space step $Y_k p_{\rm Y}$ can be large, and ignoring the cross term can cause serious difficulties. This phenomenon is illustrated in a two-dimensional example given by Biegler, Nocedal, and Schmid (1993). It is shown in that example that if the cross term $Z_k^T W_k Y_k p_{\rm Y}$ is ignored, the ratio $\|x_k + d_k\| / \|x_k\|$ can be arbitrarily large, even close to the solution. It is also shown that these inefficiencies disappear if the cross term is approximated as suggested in the following sections.

In the rest of the paper we allow much freedom in the choice of the basis matrices. They can be given by (2.2), can be orthonormal, or can be chosen in other ways. The only restrictions we impose are that $A_k^T Z_k = 0$ is satisfied, that the $n \times n$ matrix $[Y_k \; Z_k]$ is nonsingular and well conditioned, and that this matrix varies smoothly in a neighborhood of the solution.

3. Further Details of the Algorithm

In this section we consider how to calculate approximations w_k and $\overline{w}_k$ to $(Z_k^T W_k Y_k) p_{\mathrm{Y}}$ to be used in the determination of the search direction p_{Z} and in updating B_k, respectively. We also discuss when to skip the BFGS update of the reduced Hessian approximation, as well as the selection of the damping factor ζ_k and the penalty parameter μ_k.

To calculate approximations to $(Z^T W Y) p_{\mathrm{Y}}$, we propose two approaches. First, we consider a finite difference approximation to $Z_k^T W_k$ along the direction $Y_k p_{\mathrm{Y}}$. While this approach requires additional evaluations of reduced gradients at each iteration, it gives rise to a very good step. The second, more economical approach defines w_k and $\overline{w}_k$ in terms of a Broyden approximation to $Z_k^T W_k$, as discussed in §1, and requires no additional function or gradient evaluations. Our algorithm will normally use this second approach, but as we will later see, it is sometimes necessary to use finite differences.

3.1. Calculating w_k and $\overline{w}_k$ through Finite Differences

We first calculate the range space step p_{Y} at x_k through Equation (1.26). Next we compute the reduced gradient of the Lagrangian at $x_k + Y_k p_{\mathrm{Y}}$ and define

$$w_k = Z_k^T [\nabla L(x_k + Y_k p_{\mathrm{Y}}, \lambda_k) - \nabla L(x_k, \lambda_k)]. \tag{3.1}$$

After the step to the new iterate x_{k+1} has been taken, we define

$$\overline{w}_k = Z_k^T [\nabla L(x_k + \alpha_k Y_k p_{\mathrm{Y}}, \lambda_{k+1}) - \nabla L(x_k, \lambda_{k+1})], \tag{3.2}$$

which requires a new evaluation of gradients if $\alpha_k \neq 1$. Thus, up to three evaluations of the objective function gradient may be required at each iteration.

We note that this finite-difference approach is very similar to the algorithm of Coleman and Conn (1982, 1984). Starting at a point z_k, the Coleman-Conn algorithm (with steplength $\alpha_k = 1$) is given by

$$Z_k p_{\mathrm{Z}} = -Z(z_k) B_k^{-1} Z(z_k)^T g(z_k) \tag{3.3}$$

$$Y_k p_{\mathrm{Y}} = -Y(z_k)[A(z_k)^T Y(z_k)]^{-1} c(z_k + Z_k p_{\mathrm{Z}}) \tag{3.4}$$

$$z_{k+1} = z_k + Z_k p_{\mathrm{Z}} + Y_k p_{\mathrm{Y}}. \tag{3.5}$$

Let us now consider Algorithm I, and to better illustrate its similarity with the Coleman and Conn method, let us assume that instead of (3.1), w_k is defined by

$$w_k = Z(x_k + Y_k p_{\mathrm{Y}})^T g(x_k + Y_k p_{\mathrm{Y}}) - Z(x_k)^T g(x_k),$$

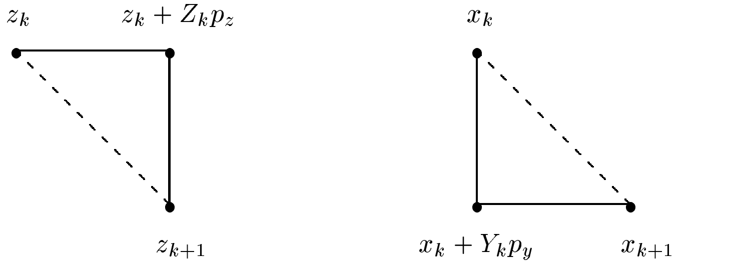
which differs from (3.1) by terms of order $O(\|p_{\mathrm{Y}}\|)$. Then Algorithm I with $\alpha_k = 1$ and $\zeta_k = 1$ is given by

$$Y_k p_{\mathrm{Y}} = -Y(x_k)[A(x_k)^T Y(x_k)]^{-1} c(x_k). \tag{3.6}$$

$$\begin{aligned} Z_k p_{\rm Z} &= -Z(x_k)B_k^{-1}[Z(x_k)^T g(x_k) + w_k] \\ &= -Z(x_k)B_k^{-1}[Z(x_k + Y_k p_{\rm Y})^T g(x_k + Y_k p_{\rm Y})]. \end{aligned} \tag{3.7}$$

$$x_{k+1} = x_k + Y_k p_{\rm Y} + Z_k p_{\rm Z}. \tag{3.8}$$

The similarity between the two approaches is apparent in Figure 1, especially if we consider the intermediate points in the Coleman-Conn iteration to be the starting and final points, respectively.



Coleman-Conn step

The step of Algorithm I

Figure 1

Comparison of Coleman-Conn method and Algorithm I

In the Coleman-Conn algorithm, the approximation B_k to the reduced Hessian $Z_k^T W_k Z_k$ is obtained by moving along the null space direction $Z_k p_{\rm Z}$, and making a new evaluation of the function and constraint gradients. To be more precise, Coleman and Conn define

$$y_k = Z_k^T[\nabla L(x_k + Z_k p_{\rm Z}, \lambda_k) - \nabla L(x_k, \lambda_k)]$$

and $s_k = Z_k^T[x_{k+1} - x_k]$ and apply a quasi-Newton formula to update B_k. Algorithm I, using finite differences, amounts essentially to the same thing. To see this, note that if Formula (3.2) is used in (1.33), then

$$y_k = Z_k^T[\nabla L(x_{k+1}, \lambda_{k+1}) - \nabla L(x_k + \alpha_k Y_k p_{\rm Y}, \lambda_{k+1})],$$

which represents a difference in reduced gradients of the Lagrangian along the null space direction $Z_k p_{\rm Z}$.

Byrd (1990) and Gilbert (1989) showed that the sequence $\{z_k + Z_k p_{\rm Z}\}$ (but not the sequence $\{z_k\}$) generated by the Coleman-Conn method converges one-step Q-superlinearly. If Algorithm I always computed the correction terms w_k and $\overline{w}_k$ by finite differences, its cost and convergence behavior would be similar to those of the Coleman-Conn method (except when $\alpha_k \neq 1$, which requires one extra gradient evaluation for Algorithm I). However, we will often be able to avoid the use of finite differences and instead use the more economical approach discussed next.

3.2. Using Broyden's Method to Compute w_k and $\overline{w}_k$

We can approximate the rectangular matrix $Z_k^T W_k$ by a matrix S_k updated by Broyden's method, and then compute w_k and $\overline{w}_k$ by post-multiplying this matrix by $Y_k p_{\rm Y}$ or by a multiple of this vector. As discussed in §1, it is reasonable to impose the secant equation (1.20) on this Broyden approximation, which can therefore be updated by the formula (cf. Fletcher (1987))

$$S_{k+1} = S_k + \frac{(\bar{y}_k - S_k \bar{s}_k)\bar{s}_k^T}{\bar{s}_k^T \bar{s}_k}, \tag{3.9}$$

where

$$\bar{y}_k = Z_k^T[\nabla L(x_{k+1}, \lambda_{k+1}) - \nabla L(x_k, \lambda_{k+1})] \tag{3.10}$$

and

$$\bar{s}_k = x_{k+1} - x_k. \tag{3.11}$$

We now define

$$w_k = S_k Y_k p_{\mathrm{Y}} \quad \text{ and } \quad \overline{w}_k = \alpha_k S_{k+1} Y_k p_{\mathrm{Y}}. \tag{3.12}$$

It should be noted that this approach requires the storage of the $(n-m) \times n$ matrix S_k, in addition to the reduced Hessian approximation, B_k. For problems where $n-m$ is small, this expense is far less than the storage of a full Hessian approximation to W_k. On the other hand, if $n-m$ is not very small, it may be preferable to use a limited-memory implementation of Broyden's method. Here the matrices S_k are represented implicitly, using, for example, the compact representation described in Byrd, Nocedal, and Schnabel (1992). The advantage of the limited memory implementation is that it requires the storage of only a few $n-$vectors to represent S.

Since is no guarantee that the Broyden approximations S_k will remain bounded, we need to safeguard them. At the beginning of the algorithm we choose a positive constant Γ and define

$$w_k := \begin{cases} w_k & \text{if } \|w_k\| \leq \frac{\Gamma}{\|p_{\mathrm{Y}}\|^{1/2}}\|p_{\mathrm{Y}}\| \\ w_k \frac{\Gamma\|p_{\mathrm{Y}}\|^{1/2}}{\|w_k\|} & \text{otherwise.} \end{cases} \tag{3.13}$$

The correction $\overline{w}_k$ will be safeguarded in a different way. We choose a sequence of positive numbers $\{\gamma_k\}$ such that $\Sigma_{k=1}^{\infty}\gamma_k < \infty$, and we set

$$\overline{w}_k := \begin{cases} \overline{w}_k & \text{if } \|\overline{w}_k\| \leq \alpha_k\|p_{\mathrm{Y}}\|/\gamma_k \\ \overline{w}_k \frac{\alpha_k\|p_{\mathrm{Y}}\|}{\gamma_k\|\overline{w}_k\|} & \text{otherwise.} \end{cases} \tag{3.14}$$

As the iterates converge to the solution, $p_{\mathrm{Y}} \to 0$, so that from (3.12) and from the boundedness of Y_k we see that these safeguards allow the Broyden updates S_k to become unbounded, but in a controlled manner. We will show in §4 and §5 that with the safeguards (3.13) and (3.14) Algorithm I is locally and R-linearly convergent and that this implies that the Broyden updates S_k do, in fact, remain bounded, so that the safeguards become inactive asymptotically.

Our Broyden approximation to the correction terms w_k and $\overline{w}_k$ was motivated by recent work of Gurwitz (1993). She approximates $Z_k^T W_k Z_k$ by the BFGS formula with

$$s_k = Z_k^T[x_{k+1} - x_k]$$

and

$$y_k = Z_k^T[\nabla L(x_{k+1}, \lambda_{k+1}) - \nabla L(x_k, \lambda_{k+1})]$$

and approximates $Z_k^T W_k Y_k$ by a matrix D_k using Broyden's formula (3.9) with

$$\bar{s}_k = Y_k^T[x_{k+1} - x_k]$$

$$\bar{y}_k = Z_k^T[\nabla L(x_{k+1}, \lambda_{k+1}) - \nabla L(x_k, \lambda_{k+1})] - B_k p_Z.$$

Since the updates may not always be defined, Gurwitz proposes to sometimes skip the update of B_k or D_k. She shows 1-step Q-superlinear convergence *if and only if* one of the updates is taken at each iteration, but this cannot be guaranteed. The analysis of this paper will show that it is preferable to update an approximation to $Z_k^T W_k$, as in Algorithm I, instead of an approximation to $Z_k^T W_k Y_k$, as proposed by Gurwitz, since our approach leads to 1-step superlinear convergence in all cases.

A related method was derived by Coleman and Fenyes (1992). Their lower partition BFGS formula (LPB) simultaneously updates approximations to $Z_k^T W_k Z_k$ and $Z_k^T W_k Y_k$, by means of a new variational problem. The resulting updating formula requires the solution of a cubic equation, and its roots can correspond to cases where updates should be avoided (e.g., $s_k^T y_k \leq 0$). The drawback of this approach is that choosing the correct root is not always easy.

An earlier proposal by Tagliaferro (1989) consists of approximating the matrices $Z_k^T W_k Z_k$ and $Z_k^T W_k Y_k$ using the PSB update formula and Broyden's method, respectively. One disadvantage of this approach is that the matrices generated by this updating procedure may become very ill conditioned.

3.3. Update Criterion

It is well known that the BFGS update (1.24) is well defined only if the curvature condition $s_k^T y_k > 0$ is satisfied. This condition can always be enforced in the unconstrained case by performing an appropriate line search; see, for example, Fletcher (1987). When constraints are present, however, the curvature condition $s_k^T y_k > 0$ can be difficult to obtain, even near the solution.

To show this, we first note from (1.33), (1.28), and (1.32) and from the Mean Value theorem that

$$\begin{aligned} y_k &= Z_k^T \left[\int_0^1 \nabla_{xx}^2 L(x_k + \tau\alpha_k d_k, \lambda_{k+1}) d\tau \right] \alpha_k d_k - \overline{w}_k \\ &\equiv Z_k^T \tilde{W}_k \alpha_k d_k - \overline{w}_k \\ &= Z_k^T \tilde{W}_k Z_k s_k + \alpha_k Z_k^T \tilde{W}_k Y_k p_Y - \overline{w}_k, \end{aligned} \tag{3.15}$$

where we have defined

$$\tilde{W}_k = \int_0^1 \nabla_{xx}^2 L(x_k + \tau\alpha_k d_k, \lambda_{k+1}) d\tau. \tag{3.16}$$

Thus

$$s_k^T y_k = s_k^T \left(Z_k^T \tilde{W}_k Z_k \right) s_k + \alpha_k s_k^T \left(Z_k^T \tilde{W}_k Y_k \right) p_Y - s_k^T \overline{w}_k. \tag{3.17}$$

Near the solution, the first term on the right-hand side will be positive, since $Z_k^T \tilde{W}_k Z_k$ can be assumed positive definite. Nevertheless, the last two terms are of uncertain sign and can make $s_k^T y_k$ negative. Several reduced Hessian methods in the literature set $\overline{w}_k$ equal to zero for all k, and update B_k only if p_Y is small enough compared with s_k that the first term in the right-hand side of (3.17) dominates the second term (see Nocedal and Overton (1985), Gurwitz and Overton (1989), and Xie (1991)).

Skipping the BFGS update may appear to be a crude heuristic, but we argue that it gives rise to a sound algorithm. First of all, the last two terms in (3.17) normally converge to zero faster than the first term, so that the right-hand side of (3.17) will often be positive near the solution and BFGS updating will take place frequently. Furthermore, if the right-hand side of (3.17) is negative, the range space step $Y_k p_{\mathrm{Y}}$ is relatively large, resulting in sufficient progress towards the solution. These arguments will be made more precise in §5.

We therefore opt for skipping the BFGS update, when necessary, and we now present a strategy for deciding when to do so. Recall that σ_k, defined by (1.34), converges to zero if the iterates converge to x_*.

Update Criterion I
Choose a constant $\gamma_{\mathrm{fd}} > 0$ and a sequence of positive numbers $\{\gamma_k\}$ such that $\Sigma_{k=1}^{\infty}\gamma_k < \infty$ (this is the same sequence $\{\gamma_k\}$ that was used in (3.14)).

- *If $\overline{w}_k$ is computed by Broyden's method, and if both $s_k^T y_k > 0$ and*

$$\|p_{\mathrm{Y}}\| \leq \gamma_k^2 \|p_{\mathrm{Z}}\| \tag{3.18}$$

hold at iteration k, then update the matrix B_k by means of the BFGS formula (1.24) with s_k and y_k given by (1.32) and (1.33). Otherwise, set $B_{k+1} = B_k$.

- *If $\overline{w}_k$ is computed by finite differences, and if both $s_k^T y_k > 0$ and*

$$\|p_{\mathrm{Y}}\| \leq \gamma_{\mathrm{fd}} \|p_{\mathrm{Z}}\| / \sigma_k^{1/2} \tag{3.19}$$

hold at iteration k, then update the matrix B_k by means of the BFGS formula (1.24) with s_k and y_k given by (1.32) and (1.33). Otherwise, set $B_{k+1} = B_k$.

Note that σ_k requires knowledge of the solution vector x_* and is therefore not computable. However, we will later see that σ_k can be replaced by any quantity that is of the same order as the error e_k, for example, the optimality conditions $(\|Z_k^T g_k\| + \|c_k\|)$. Nevertheless, for convenience we will leave σ_k in (3.19).

We now closely consider the properties of the BFGS matrices B_k when Update Criterion I is used. Let us define

$$\cos\theta_k = \frac{s_k^T B_k s_k}{\|s_k\|\,\|B_k s_k\|}, \tag{3.20}$$

which, as we will see, is a measure of the goodness of the null space step $Z_k p_{\mathrm{Z}}$. We begin by restating a theorem from Byrd and Nocedal (1989) regarding the behavior of $\cos\theta_k$ when the matrix B_k is updated by the BFGS formula.

Theorem 3.1 *Let $\{B_k\}$ be generated by the BFGS formula (1.24) where, for all $k \geq 1$, $s_k \neq 0$ and*

$$\frac{y_k^T s_k}{s_k^T s_k} \geq m > 0 \tag{3.21}$$

$$\frac{\|y_k\|^2}{y_k^T s_k} \leq M. \tag{3.22}$$

Then, there exist constants $\beta_1, \beta_2, \beta_3 > 0$ such that, for any $k \geq 1$, the relations

$$\cos\theta_j \geq \beta_1 \tag{3.23}$$

$$\beta_2 \leq \frac{\|B_j s_j\|}{\|s_j\|} \leq \beta_3 \tag{3.24}$$

hold for at least $\lceil \frac{1}{2}k \rceil$ values of $j \in [1, k]$.

This theorem refers to the iterates for which BFGS updating takes place; but since, for the other iterates, $B_{k+1} = B_k$, the theorem characterizes the whole sequence of matrices $\{B_k\}$. Theorem 3.1 states that, if $s_k^T y_k$ is always sufficiently positive, in the sense that Conditions (3.21) and (3.22) are satisfied, then at least half of the iterates at which updating takes place are such that $\cos\theta_j$ is bounded away from zero and $B_j s_j = O(\|s_j\|)$. Since it will be useful to refer easily to these iterates, we make the following definition.

Definition 3.1 *We define J to be the set of iterates for which (3.23) and (3.24) hold. We call J the set of "good iterates" and define $J_k = J \cap \{1, 2, ..., k\}$.*

Note that if the matrices B_k are updated only a finite number of times, their condition number is bounded, and (3.23)–(3.24) are satisfied for all k. Thus in this case all iterates are good iterates.

We now study the case when BFGS updating takes place an infinite number of times. Let us assume that all functions under consideration are smooth and bounded. If at a solution point x_* the reduced Hessian $Z_*^T W_* Z_*$ is positive definite, then for all x_k in a neighborhood of x_* the smallest eigenvalue of $Z_k^T \tilde{W}_k Z_k$ is bounded away from zero ($\tilde{W}_k$ is defined in (3.16)). We now show that in such a neighborhood Update Criterion I implies (3.21)–(3.22).

Let us first consider the case when $\overline{w}_k$ is computed by Broyden's method. Using (3.17), (3.18), and (3.14), and since γ_k converges to zero, we have

$$\begin{aligned} s_k^T y_k &\geq C\|s_k\|^2 - O(\gamma_k^2\|s_k\|^2) - O(\gamma_k\|s_k\|^2) \\ &\geq m\|s_k\|^2, \end{aligned} \tag{3.25}$$

for some positive constants C, m. Also, from (3.15), (3.18), and (3.14) we have that

$$\begin{aligned} \|y_k\| &\leq O(\|s_k\|) + O(\gamma_k^2\|s_k\|) + O(\gamma_k\|s_k\|) \\ &\leq O(\|s_k\|). \end{aligned} \tag{3.26}$$

We thus see from (3.25)–(3.26) that there is a constant M such that for all k for which updating takes place,

$$\frac{\|y_k\|^2}{y_k^T s_k} \leq M,$$

which together with (3.25) shows that (3.21)–(3.22) hold when Broyden's method is used.

If $\overline{w}_k$ is computed by the finite-difference formula (3.2), we see from (1.33) and the Mean Value theorem that there is a matrix $\hat{W}_k$ such that

$$\begin{aligned} y_k &= Z_k^T[\nabla L(x_{k+1}, \lambda_{k+1}) - \nabla L(x_k + \alpha_k Y_k p_{\mathrm{Y}}, \lambda_{k+1})] \\ &\equiv Z_k^T \hat{W}_k Z_k s_k. \end{aligned}$$

Reasoning as before we see that (3.25) and (3.26) also hold in this case, and that (3.21)–(3.22) are satisfied in the case when finite differences are used. We have therefore established the following result.

Lemma 3.1 *In a neighborhood of a solution point x_*, and whenever BFGS updating takes place as stipulated by Update Criterion I, $s_k^T y_k$ is sufficiently positive in the sense that (3.21)–(3.22) hold.*

3.4. Choosing μ_k and ζ_k

We will now see that by appropriately choosing the penalty parameter μ_k and the damping parameter ζ_k for w_k, the search direction generated by Algorithm I is always a descent direction for the merit function. Moreover, for the good iterates J, it is a direction of strong descent.

Since d_k satisfies the linearized constraint (1.11), it is easy to show (see Eq. (2.24) of Byrd and Nocedal (1991)) that the directional derivative of the ℓ_1 merit function in the direction d_k is given by

$$D\phi_{\mu_k}(x_k; d_k) = g_k^T d_k - \mu_k \|c_k\|_1. \tag{3.27}$$

The fact that the same right inverse of A_k^T is used in (1.26) and (1.31) implies that

$$g_k^T Y_k p_{\rm Y} = \lambda_k^T c_k. \tag{3.28}$$

Recalling the decomposition (1.28) and using (3.28), we obtain

$$\begin{aligned} D\phi_{\mu_k}(x_k; d_k) &= g_k^T Z_k p_{\rm Z} - \mu_k \|c_k\|_1 + \lambda_k^T c_k \\ &= (Z_k^T g_k + \zeta_k w_k)^T p_{\rm Z} - \zeta_k w_k^T p_{\rm Z} - \mu_k \|c_k\|_1 + \lambda_k^T c_k. \end{aligned} \tag{3.29}$$

Now from (1.32) and (1.27) we have that

$$B_k s_k = -\alpha_k (Z_k^T g_k + \zeta_k w_k). \tag{3.30}$$

Substituting this in (3.20), we obtain

$$\cos\theta_k = \frac{-(Z_k^T g_k + \zeta_k w_k)^T p_{\rm Z}}{\|Z_k^T g_k + \zeta_k w_k\| \, \|p_{\rm Z}\|}. \tag{3.31}$$

Recalling the inequality $\lambda_k^T c_k \le \|\lambda_k\|_\infty \|c_k\|_1$, and using (3.31) in (3.29), we obtain, for all k,

$$D\phi_{\mu_k}(x_k; d_k) \le -\|Z_k^T g_k + \zeta_k w_k\| \, \|p_{\rm Z}\| \cos\theta_k - \zeta_k w_k^T p_{\rm Z} - (\mu_k - \|\lambda_k\|_\infty)\|c_k\|_1. \tag{3.32}$$

Note also from (3.30) and (1.32) that

$$\frac{\|s_k\|}{\|B_k s_k\|} = \frac{\|p_{\rm Z}\|}{\|Z_k^T g_k + \zeta_k w_k\|}. \tag{3.33}$$

We now concentrate on the good iterates J, as given in Definition 3.1. If $j \in J$, we have from (3.33) and (3.24) that

$$\frac{1}{\beta_3}\|Z_j^T g_j + \zeta_j w_j\| \le \|p_{\rm Z}^{(j)}\| \le \frac{1}{\beta_2}\|Z_j^T g_j + \zeta_j w_j\|. \tag{3.34}$$

Using this and (3.23) in (3.32), we obtain, for $j \in J$,

$$\begin{aligned} D\phi_{\mu_j}(x_j; d_j) &\le -\frac{1}{\beta_3}\|Z_j^T g_j + \zeta_j w_j\|^2 \cos\theta_j - \zeta_k w_j^T p_{\rm Z}^{(j)} - (\mu_j - \|\lambda_j\|_\infty)\|c_j\|_1 \\ &\le -\frac{\beta_1}{\beta_3}\|Z_j^T g_j\|^2 - \frac{2\zeta_j \cos\theta_j}{\beta_3}(g_j^T Z_j w_j) - \zeta_j w_j^T p_{\rm Z}^{(j)} - (\mu_j - \|\lambda_j\|_\infty)\|c_j\|_1, \end{aligned}$$

where we have dropped the nonpositive term $-\zeta_j^2 \cos\theta_j \|w_j\|^2/\beta_3$. Since we can assume that $\beta_3 > 1$ (it is defined as an upper bound in (3.24)), we have

$$D\phi_{\mu_j}(x_j; d_j) \le -\frac{\beta_1}{\beta_3}\|Z_j^T g_j\|^2 + \left[2\zeta_j \cos\theta_j |g_j^T Z_j w_j| - \zeta_j w_j^T p_{\rm Z}^{(j)}\right] - (\mu_j - \|\lambda_j\|_\infty)\|c_j\|_1.$$

It is now clear that if

$$2\zeta_j \cos\theta_j |g_j^T Z_j w_j| - \zeta_j w_j^T p_{\mathrm{Z}}^{(j)} \leq \rho \|c_j\|_1, \tag{3.35}$$

for some constant ρ, and if

$$\mu_j \geq \|\lambda_j\|_\infty + 2\rho, \tag{3.36}$$

then for all $j \in J$,

$$D\phi_{\mu_j}(x_j; d_j) \leq -\frac{\beta_1}{\beta_3} \|Z_j^T g_j\|^2 - \rho \|c_j\|_1. \tag{3.37}$$

This means that if (3.35) and (3.36) hold, then for the good iterates $j \in J$, the search direction d_j is a strong direction of descent for the ℓ_1 merit function in the sense that the first-order reduction is proportional to the KKT error.

We will choose ζ_k so that (3.35) holds for *all* iterations. To show how to do this, we note from (1.27) that

$$p_{\mathrm{Z}} = -B_k^{-1} Z_k^T g_k - \zeta_k B_k^{-1} w_k,$$

so that, for $j = k$, (3.35) can be written as

$$\zeta_k [2 \cos\theta_k |g_k^T Z_k w_k| + w_k^T B_k^{-1} Z_k^T g_k + \zeta_k w_k^T B_k^{-1} w_k] \leq \rho \|c_k\|_1. \tag{3.38}$$

Clearly this condition is satisfied for a sufficiently small and positive value of ζ_k. Specifically, at the beginning of the algorithm we choose a constant $\rho > 0$ and, at every iteration k, define

$$\zeta_k = \min\{1, \hat{\zeta}_k\}, \tag{3.39}$$

where $\hat{\zeta}_k$ is the largest value that satisfies (3.38) as an equality.

The penalty parameter μ_k must satisfy (3.36), so we define it at every iteration of the algorithm by

$$\mu_k = \begin{cases} \mu_{k-1} & \text{if } \mu_{k-1} \geq \|\lambda_k\|_\infty + 2\rho \\ \|\lambda_k\|_\infty + 3\rho & \text{otherwise.} \end{cases} \tag{3.40}$$

The damping factor ζ_k and the updating formula for the penalty parameter μ_k have been defined so as to give strong descent for the good iterates J. We now show that they ensure that the search direction is also a direction of descent (but not necessarily of strong descent) for the other iterates, $k \notin J$. Since (3.35) holds for all iterations by our choice of ζ_k, we have in particular

$$-\zeta_k w_k^T p_{\mathrm{Z}} \leq \rho \|c_k\|_1.$$

Using this and (3.40) in (3.32), we have

$$D\phi_{\mu_k}(x_k; d_k) \leq -\|Z_k^T g_k + \zeta_k w_k\| \, \|p_{\mathrm{Z}}\| \cos\theta_k - \rho_k \|c_k\|_1. \tag{3.41}$$

The directional derivative is thus nonpositive. Furthermore, since $w_k = 0$ whenever $c_k = 0$ (regardless of whether w_k is obtained by finite differences or through Broyden's method), it is easy to show that this directional derivative can be zero only at a stationary point of problem (1.1)–(1.2).

3.5. The Algorithm

We can now give a complete description of the algorithm that incorporates all the ideas discussed so far and that specifies the only remaining question, namely, when to apply finite differences and when to use Broyden's method to approximate the cross term. The idea is to consider the relative sizes of p_Y and p_Z. Update Criterion I generates the three regions R_1, R_2, and R_3 illustrated in Figure 2. The algorithm starts by computing w_k through Broyden's method and by calculating p_Y and p_Z. If the search direction is in R_1 or R_3, we proceed. Otherwise we recompute w_k by finite differences, use this value to recompute p_Z, and proceed. The reason for applying finite differences in this fashion is that in the middle region R_2 Broyden's method is not good enough, nor is the convergence sufficiently tangential, to give a superlinear step. Therefore we must resort to finite differences to obtain a good estimate of w_k. The motivation behind this strategy will become clearer when we study the rate of convergence of the algorithm in §6.

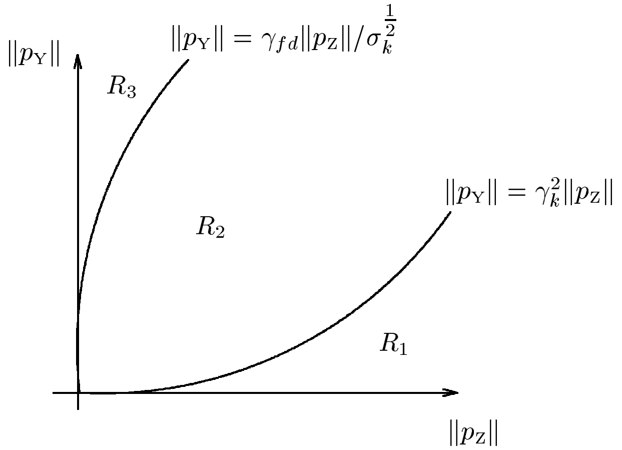

$||p_\mathrm{Y}||$
$||p_\mathrm{Y}|| = \gamma_{fd}||p_\mathrm{Z}||/\sigma_k^{\frac{1}{2}}$
R_3
R_2
$||p_\mathrm{Y}|| = \gamma_k^2||p_\mathrm{Z}||$
R_1
$||p_\mathrm{Z}||$

Figure 2

Three regions generated by Update Criterion I

Note from Updating Criterion I that the BFGS update of B_k is skipped if the search direction is in R_3. A precise description of the algorithm follows.

Algorithm II

1. Choose constants $\eta \in (0, 1/2)$, $\rho > 0$ and τ, τ' with $0 < \tau < \tau' < 1$, and positive constants Γ and γ_fd for Conditions (3.13) and (3.19), respectively. For Conditions (3.14) and (3.18), select a summable sequence of positive numbers $\{\gamma_k\}$. Set $k := 1$, and choose a starting point x_1, an initial value μ_1 for the penalty parameter, an $(n-m) \times (n-m)$ symmetric and positive definite starting matrix B_1 and an $(n-m) \times n$ starting matrix S_1.
2. Evaluate f_k, g_k, c_k, and A_k, and compute Y_k and Z_k.
3. Set *findiff* = *false* and compute p_Y by solving the system
$$(A_k^T Y_k) p_\mathrm{Y} = -c_k. \qquad \text{(range space step)} \qquad (3.42)$$
4. Calculate w_k using Broyden's method, from Equations (3.12) and (3.13).

5. Choose the damping parameter ζ_k from Equations (3.38) and (3.39), and compute p_Z from

$$B_k p_Z = -[Z_k^T g_k + \zeta_k w_k]. \qquad \text{(null space step)} \tag{3.43}$$

6. If (3.19) is satisfied and (3.18) is *not* satisfied, set *findiff* = *true* and recompute w_k from Equation (3.1). (In practice we replace σ_k by $\|Z_k^T g_k\| + \|c_k\|$ in (3.19).)

7. If *findiff* = *true*, use this new value of w_k to choose the damping parameter ζ_k from Equations (3.38) and (3.39), and recompute p_Z from Equation (3.43).

8. Define the search direction by

$$d_k = Y_k p_Y + Z_k p_Z, \tag{3.44}$$

and set $\alpha_k = 1$.

9. Test the line search condition

$$\phi_{\mu_k}(x_k + \alpha_k d_k) \le \phi_{\mu_k}(x_k) + \eta \alpha_k D\phi_{\mu_k}(x_k; d_k). \tag{3.45}$$

10. If (3.45) is not satisfied, choose a new $\alpha_k \in [\tau\alpha_k, \tau'\alpha_k]$ and go to 9; otherwise set

$$x_{k+1} = x_k + \alpha_k d_k. \tag{3.46}$$

11. Evaluate $f_{k+1}, g_{k+1}, c_{k+1}, A_{k+1}$, and compute Y_{k+1} and Z_{k+1}.

12. Compute the Lagrange multiplier estimate

$$\lambda_{k+1} = -[Y_{k+1}^T A_{k+1}]^{-1} Y_{k+1}^T g_{k+1}, \tag{3.47}$$

and update μ_k so as to satisfy (3.40).

13. Update S_{k+1} using Equations (3.9) to (3.11). If *findiff* = *false*, calculate $\overline{w}_k$ by Broyden's method through Equations (3.12) and (3.14); otherwise calculate $\overline{w}_k$ by (3.2).

14. If ($s_k^T y_k \le 0$) or if ($findiff = true$ and (3.19) is not satisfied) or if ($findiff = false$ and (3.18) is not satisfied), set $B_{k+1} = B_k$. Else, compute

$$s_k = \alpha_k p_Z, \tag{3.48}$$

$$y_k = Z_k^T[\nabla L(x_{k+1}, \lambda_{k+1}) - \nabla L(x_k, \lambda_{k+1})] - \overline{w}_k, \tag{3.49}$$

and compute B_{k+1} by the BFGS formula (1.24).

15. Set $k := k + 1$, and go to 3.

We mentioned in §3.1 that, when using finite differences, there are various ways of defining w_k and $\overline{w}_k$, but for concreteness we now assume in steps 6 and 13 that they are computed by (3.1) and (3.2), respectively. We should also point out that the curves in Figure 2 may intersect, creating a fourth region, and in practice we should stipulate a new set of conditions in this region. We discuss these conditions in another paper that considers the implementation of the algorithm (Biegler, Nocedal, and Schmid (1993)).

In the next sections we present several convergence results for Algorithm II. The analysis, which does not assume that the BFGS matrices B_k or the Broyden matrices S_k are bounded, is based on the results of Byrd and Nocedal (1991), who have studied the convergence of the Coleman-Conn updating algorithm. We also make use of some results of Xie (1991), who has analyzed the algorithm proposed by Nocedal and Overton (1985) using nonorthogonal bases Y and Z. The main difference between this paper and that of Xie stems from our use of the correction terms w_k and $\overline{w}_k$, which are not employed in his method.

4. Semi-Local Behavior of the Algorithm

We first show that the merit function ϕ decreases significantly at the good iterates J and that this gives the algorithm a weak convergence property. To establish the results of this section, we make the following assumptions.

Assumptions 4.1 The sequence $\{x_k\}$ generated by Algorithm II is contained in a convex set D with the following properties:

(I) The functions $f : \mathbf{R}^n \to \mathbf{R}$ and $c : \mathbf{R}^n \to \mathbf{R}^m$ and their first and second derivatives are uniformly bounded in norm over D.

(II) The matrix $A(x)$ has full column rank for all $x \in D$, and there exist constants γ_0 and β_0 such that

$$\|Y(x)[A(x)^T Y(x)]^{-1}\| \le \gamma_0, \quad \|Z(x)\| \le \beta_0, \tag{4.1}$$

for all $x \in D$.

(III) For all $k \ge 1$ for which B_k is updated, (3.21) and (3.22) hold.

(IV) The correction term w_k is chosen so that there is a constant $\kappa > 0$ such that for all k,

$$\|w_k\| \le \kappa \|c_k\|^{1/2}. \tag{4.2}$$

Note that Condition (I) is rather strong, since it would often be satisfied only if D is bounded, and it is far from certain that the iterates will remain in a bounded set. Nevertheless, the convergence result of this section can be combined with the local analysis of §5 to give a satisfactory semi-global result. Condition (II) requires that the basis matrices Y and Z be chosen carefully, and is important to obtain good behavior in practice. Note that (4.1) and (3.42) imply that

$$\|Y_k p_{\mathrm{Y}}\| \le \gamma_0 \|c_k\|. \tag{4.3}$$

Condition (III) is justified by Lemma 3.1. Condition (III) and Theorem 3.1 ensure that at least half of the iterates at which BFGS updating takes place are good iterates.

We have left some freedom in the choice of w_k since (4.2) suffices for the analysis of this section. Relation (4.2) holds for the finite-difference approach, since (3.1) implies that $w_k = O(Y_k p_{\mathrm{Y}})$ and since (I) ensures that $\{\|c_k\|\}$ is uniformly bounded (see (5.21)). Furthermore, the safeguard (3.13) and (4.3) immediately imply that (4.2) is satisfied when the Broyden approximation is used.

The following result concerns the good iterates J, as given in Definition 3.1.

Lemma 4.1 *If Assumptions 4.1 hold and if $\mu_j = \mu$ is constant for all sufficiently large j, then there is a positive constant γ_μ such that for all large $j \in J$,*

$$\phi_\mu(x_j) - \phi_\mu(x_{j+1}) \ge \gamma_\mu \left[\|Z_j^T g_j\|^2 + \|c_j\|_1 \right]. \tag{4.4}$$

Proof. Using (3.37), we have for all $j \in J$

$$D\phi_{\mu_j}(x_j; d_j) \leq -b_2 \left[\|Z_j^T g_j\|^2 + \|c_j\|_1\right], \tag{4.5}$$

where $b_2 = \min(\beta_1/\beta_3, \rho)$. Note that the line search enforces the Armijo condition (3.45),

$$\phi_{\mu_j}(x_j) - \phi_{\mu_j}(x_{j+1}) \geq -\eta\alpha_j D\phi_{\mu_j}(x_j; d_j). \tag{4.6}$$

It is then clear from (4.5) that (4.4) holds, provided the α_j, $j \in J$, can be bounded from below. Suppose that $\alpha_j < 1$, which means that (4.6) failed for a steplength $\tilde{\alpha}$:

$$\phi_{\mu_j}(x_j + \tilde{\alpha} d_j) - \phi_{\mu_j}(x_j) > \eta\tilde{\alpha} D\phi_{\mu_j}(x_j; d_j), \tag{4.7}$$

where

$$\tau\tilde{\alpha} \leq \alpha_j \tag{4.8}$$

(see Step 10 of Algorithm II). On the other hand, expanding to second order, we have

$$\phi_{\mu_j}(x_j + \tilde{\alpha} d_j) - \phi_{\mu_j}(x_j) \leq \tilde{\alpha} D\phi_{\mu_j}(x_j; d_j) + \tilde{\alpha}^2 b_1 \|d_j\|^2, \tag{4.9}$$

where b_1 depends on μ_j. Combining (4.7) and (4.9), we have

$$(\eta - 1)\tilde{\alpha} D\phi_{\mu_j}(x_j; d_j) < \tilde{\alpha}^2 b_1 \|d_j\|^2. \tag{4.10}$$

Next we show that, for $j \in J$,

$$\|d_j\|^2 \leq b_3[\|Z_j^T g_j\|^2 + \|c_j\|_1], \tag{4.11}$$

for some constant b_3. To do this, we make repeated use of the following elementary result:

$$a, b \geq 0 \quad \Rightarrow \quad a^2 + 2ab + b^2 \leq 3a^2 + 3b^2. \tag{4.12}$$

Using (3.44), (4.12), (4.1), and (4.3), we have

$$\begin{aligned} \|d_j\|^2 &\leq \|Z_j p_Z^{(j)}\|^2 + 2\|Z_j p_Z^{(j)}\| \, \|Y_j p_Y^{(j)}\| + \|Y_j p_Y^{(j)}\|^2 \\ &\leq 3\left[\|Z_j p_Z^{(j)}\|^2 + \|Y_j p_Y^{(j)}\|^2\right] \\ &\leq 3\left[\beta_0^2 \|p_Z^{(j)}\|^2 + \gamma_0^2 \|c_j\|^2\right]. \end{aligned} \tag{4.13}$$

Also by (3.34), (4.12), and (4.2) and noting that $\|\cdot\| \leq \|\cdot\|_1$, we have that for $j \in J$

$$\begin{aligned} \|p_Z^{(j)}\|^2 &\leq \frac{1}{\beta_2^2}\left[\|Z_j^T g_j\|^2 + 2\zeta_j \|Z_j^T g_j\| \, \|w_j\| + \zeta_j^2 \|w_j\|^2\right] \\ &\leq \frac{3}{\beta_2^2}\left[\|Z_j^T g_j\|^2 + \zeta_j^2 \|w_j\|^2\right] \\ &\leq \frac{3}{\beta_2^2}\left[\|Z_j^T g_j\|^2 + \kappa^2 \|c_j\|_1\right], \end{aligned}$$

since $\zeta_j \leq 1$. Since $\|c_j\|_1$ is uniformly bounded on D, we see from this relation and (4.13) that (4.11) holds, where

$$b_3 = \max\{9\beta_0^2/\beta_2^2, 3(3\kappa^2\beta_0^2/\beta_2^2 + \gamma_0^2 \sup_{x \in D} \|c(x)\|)\}.$$

Combining (4.10), (4.5), and (4.11), and recalling that $\eta < 1$, we obtain

$$\tilde{\alpha} > \frac{(1-\eta)b_2}{b_1 b_3}. \tag{4.14}$$

This relation and (4.8) imply that the steplengths α_j are bounded away from zero for all $j \in J$. Since by assumption $\mu_j = \mu$ for all large j, we conclude that (4.4) holds with $\gamma_\mu = \eta b_2 \min\{1, (1-\eta)\tau b_2/(b_1 b_3)\}$.

□

It is now easy to show that the penalty parameter settles down and that the set of iterates is not bounded away from stationary points of the problem.

Theorem 4.2 *If Assumptions 4.1 hold, then the weights* $\{\mu_k\}$ *are constant for all sufficiently large* k *and*

$$\liminf_{k\to\infty}(\|Z_k^T g_k\| + \|c_k\|) = 0.$$

Proof. First note that by Assumptions 4.1 (I)–(II) and (3.47) that $\{\|\lambda_k\|\}$ is bounded. Therefore, since the procedure (3.40) increases μ_k by at least ρ whenever it changes the penalty parameter, it follows that there are an index k_0 and a value μ such that for all $k > k_0$, $\mu_k = \mu \geq \|\lambda_k\| + 2\rho$.

If BFGS updating is performed an infinite number of times, by Assumptions 4.1-(III) and Theorem 3.1 there is an infinite set J of good iterates, and by Lemma 4.1 and the fact that the Armijo condition (3.45) forces $\phi_\mu(x_k)$ to decrease at each iterate, we have that for $k > k_0$,

$$\begin{aligned}\phi_\mu(x_{k_0}) - \phi_\mu(x_{k+1}) &= \sum_{j=k_0}^{k}(\phi_\mu(x_j) - \phi_\mu(x_{j+1})) \\ &\geq \sum_{j\in J\cap[k_0,k]}(\phi_\mu(x_j) - \phi_\mu(x_{j+1})) \\ &\geq \gamma_\mu \sum_{j\in J\cap[k_0,k]}[\|Z_j^T g_j\|^2 + \|c_j\|_1].\end{aligned}$$

By Assumption 4.1(I) $\phi_\mu(x)$ is bounded below for all $x \in D$, so the last sum is finite, and thus the term inside the square brackets converges to zero. Therefore

$$\lim_{\substack{j\in J\\ j\to\infty}}(\|Z_j^T g_j\| + \|c_j\|_1) = 0. \tag{4.15}$$

If BFGS updating is performed a finite number of times, then, as discussed after Definition 3.1, all iterates are good iterates, and in this case we obtain the stronger result

$$\lim_{k\to\infty}(\|Z_k^T g_k\| + \|c_k\|_1) = 0.$$

□

5. Local Convergence

In this section we show that if x_* is a local minimizer that satisfies the second-order optimality conditions, and if the penalty parameter μ_k is chosen large enough, then x_* is a point of attraction for the sequence of iterates $\{x_k\}$ generated by Algorithm II. To prove this result, we will make the following assumptions. In what follows, G denotes the reduced Hessian of the Lagrangian function, namely,

$$G_k = Z_k^T \nabla_{xx}^2 L(x_k, \lambda_k) Z_k. \tag{5.1}$$

Assumptions 5.1 The point x_* is a local minimizer for problem (1.1)–(1.2), at which the following conditions hold.

(1) The functions $f : \mathbf{R}^n \to \mathbf{R}$ and $c : \mathbf{R}^n \to \mathbf{R}^m$ are twice continuously differentiable in a neighborhood of x_*, and their Hessians are Lipschitz continuous in a neighborhood of x_*.

(2) The matrix $A(x_*)$ has full column rank. This implies that there exists a vector $\lambda_* \in \mathbf{R}^m$ such that

$$\nabla L(x_*, \lambda_*) = g(x_*) + A(x_*)\lambda_* = 0.$$

(3) For all $q \in \mathbf{R}^{n-m}$, $q \neq 0$, we have $q^T G_* q > 0$.

(4) There exist constants γ_0, β_0, and γ_c such that, for all x in a neighborhood of x_*,

$$\|Y(x)[A(x)^T Y(x)]^{-1}\| \leq \gamma_0, \quad \|Z(x)\| \leq \beta_0, \tag{5.2}$$

and

$$\|[Y(x)\; Z(x)]^{-1}\| \leq \gamma_c. \tag{5.3}$$

(5) $Z(x)$ and $\lambda(x)$ are Lipschitz continuous in a neighborhood of x_*. That is, there exist constants γ_Z and γ_λ such that

$$\begin{aligned} \|\lambda(x) - \lambda(z)\| &\leq \gamma_\lambda \|x - z\|, && (5.4) \\ \|Z(x) - Z(z)\| &\leq \gamma_Z \|x - z\|, && (5.5) \end{aligned}$$

for all x, z near x_*.

Note that (1), (3), and (5) imply that for all (x, λ) sufficiently near (x_*, λ_*), and for all $q \in \mathbf{R}^{n-m}$,

$$m\|q\|^2 \leq q^T G(x, \lambda) q \leq M\|q\|^2, \tag{5.6}$$

for some positive constants m, M. We also note that Assumptions 5.1 ensure that the conditions (3.21)–(3.22) required by Theorem 3.1 hold whenever BFGS updating takes place in a neighborhood of x_*, as shown in Lemma 3.1. Therefore Theorem 3.1 can be applied in the convergence analysis.

The following two lemmas are proved by Xie (1991) for very general choices of Y and Z. His result generalizes Lemmas 4.1 and 4.2 of Byrd and Nocedal (1991); see also Powell (1978).

Lemma 5.1 *If Assumptions 5.1 hold, then for all x sufficiently near x_**

$$\gamma_1 \|x - x_*\| \leq \|c(x)\| + \|Z(x)^T g(x)\| \leq \gamma_2 \|x - x_*\|, \tag{5.7}$$

for some positive constants γ_1, γ_2.

This result states that, near x_*, the quantities $c(x)$ and $Z(x)^T g(x)$ may be regarded as a measure of the error at x. The next lemma states that, for a large enough weight, the merit function may also be regarded as a measure of the error.

Lemma 5.2 *Suppose that Assumptions 5.1 hold at x_*. Then for any $\mu > \|\lambda_*\|_\infty$ there exist constants $\gamma_3 > 0$ and $\gamma_4 > 0$, such that for all x sufficiently near x_**

$$\gamma_3 \|x - x_*\|^2 \leq \phi_\mu(x) - \phi_\mu(x_*) \leq \gamma_4 \left[\|Z(x)^T g(x)\|^2 + \|c(x)\|_1 \right]. \tag{5.8}$$

Note that the left inequality in (5.8) implies that, for a sufficiently large value of the penalty parameter, the merit function will have a strong local minimizer at x_*. We will now use the descent property of Algorithm II to show convergence of the algorithm. However, because of the nonconvexity of the problem, the line search could generate a step that decreases the merit function but that takes us away from the neighborhood of x_*. To rule this out, we make the following assumption.

Assumption 5.2 The line search has the property that, for all large k, $\phi_\mu((1-\theta)x_k + \theta x_{k+1}) \leq \phi_\mu(x_k)$ for all $\theta \in [0,1]$. In other words, x_{k+1} is in the connected component of the level set $\{x : \phi_\mu(x) \leq \phi_\mu(x_k)\}$ that contains x_k.

There is no practical line search algorithm that can guarantee this condition, but it is likely to hold close to x_*. Assumption 5.2 is made by Byrd, Nocedal, and Yuan (1987) when analyzing the convergence of variable metric methods for unconstrained problems, as well as by Byrd and Nocedal (1991) in the analysis of Coleman-Conn updates for equality constrained optimization.

Lemma 5.3 *Suppose that the iterates generated by Algorithm II (with a line search satisfying Assumptions 5.2) are contained in a convex region D satisfying Assumptions 4.1. If an iterate x_{k_0} is sufficiently close to a solution point x_* that satisfies Assumptions 5.1, and if the weight μ_{k_0} is large enough, then the sequence of iterates converges to x_*.*

Proof. By Assumptions 4.1 (I)–(II) and (3.47) we know that $\{\|\lambda_k\|\}$ is bounded. Therefore the procedure (3.40) ensures that the weights μ_k are constant, say $\mu_k = \mu$ for all large k. Moreover, if an iterate gets sufficiently close to x_*, we know by (3.40) and by the continuity of λ that $\mu > \|\lambda_*\|$. For such value of μ, Lemma 5.2 implies that the merit function has a strict local minimizer at x_*. Now suppose that once the penalty parameter has settled, and for a given $\epsilon > 0$, there is an iterate x_{k_0} such that

$$\|x_{k_0} - x_*\| \leq \frac{\gamma_3}{\gamma_2 \gamma_4 \hat{\gamma}_0} \epsilon^2,$$

where $\hat{\gamma}_0$ is such that $\|\cdot\|_1 \leq \hat{\gamma}_0 \|\cdot\|$. Assumption 5.2 shows that for any $k \geq k_0$, x_k is in the connected component of the level set of x_{k_0} that contains x_{k_0}, and we can assume that ϵ is small enough that Lemmas 5.1 and 5.2 hold in this level set. Thus since $\phi_\mu(x_k) \leq \phi_\mu(x_{k_0})$ for $k \geq k_0$, and since we can assume that $\|Z_{k_0}^T g_{k_0}\| \leq 1$, we have from Lemmas 5.1 and 5.2, for any $k \geq k_0$

$$\begin{aligned}
\|x_k - x_*\| &\leq \gamma_3^{-\frac{1}{2}} (\phi_\mu(x_k) - \phi_\mu(x_*))^{\frac{1}{2}} \\
&\leq \gamma_3^{-\frac{1}{2}} (\phi_\mu(x_{k_0}) - \phi_\mu(x_*))^{\frac{1}{2}} \\
&\leq \left(\frac{\gamma_4}{\gamma_3}\right)^{\frac{1}{2}} \left[\|Z_{k_0}^T g_{k_0}\|^2 + \|c_{k_0}\|_1 \right]^{\frac{1}{2}} \\
&\leq \left(\frac{\gamma_4}{\gamma_3}\right)^{\frac{1}{2}} \left[\|Z_{k_0}^T g_{k_0}\|^2 + \hat{\gamma}_0 \|c_{k_0}\| \right]^{\frac{1}{2}}
\end{aligned}$$

$$\begin{aligned} &\leq \left(\frac{\gamma_2\gamma_4\hat{\gamma}_0}{\gamma_3}\|x_{k_0}-x_*\|\right)^{\frac{1}{2}} \\ &\leq \epsilon. \end{aligned}$$

This implies that the whole sequence of iterates remains in a neighborhood of radius ϵ of x_*. If ϵ is small enough, we conclude by (5.8), by the monotonicity of $\{\phi_\mu(x_k)\}$, and by Theorem 4.2 that the iterates converge to x_*.

□

The assumptions of this lemma, which is modeled after a result in Xie (1991), are restrictive — especially the assumption on the penalty parameter. One can relax these assumptions and obtain a stronger result, such as Theorem 4.3 in Byrd and Nocedal (1991), but the proof would be more complex and is not particularly relevant to Algorithm II since it is based only on the properties of the merit function. Therefore, instead of further analyzing the local convergence properties of the new algorithm, we will study its rate of convergence.

5.1. R-Linear Convergence

For the rest of the paper we assume that the line search strategy satisfies Assumption 5.2. We also assume that the iterates generated by Algorithm II converge to a point x_* at which Assumptions 5.1 hold, which implies that for all large k, $\mu_k = \mu > \|\lambda_*\|$. The analysis that follows depends on how often BFGS updating is applied. To make this concept precise, we define U to be the set of iterates at which BFGS updating takes place,

$$U = \{k : B_{k+1} = BFGS(B_k, s_k, y_k)\}, \tag{5.9}$$

and let

$$U_k = U \cap \{1, 2, ..., k\}. \tag{5.10}$$

The number of elements in U_k will be denoted by $|U_k|$.

Theorem 5.4 *Suppose that the iterates $\{x_k\}$ generated by Algorithm II converge to a point x_* that satisfies Assumptions 5.1. Then for any $k \in U$ and any $j \geq k$*

$$\|x_j - x_*\| \leq Cr^{|U_k|}, \tag{5.11}$$

for some constants $C > 0$ and $0 \leq r < 1$.

Proof. Using (4.4) and (5.8), we have for $i \in J$,

$$\phi_\mu(x_i) - \phi_\mu(x_{i+1}) \geq \frac{\gamma_\mu}{\gamma_4}\left[\phi_\mu(x_i) - \phi_\mu(x_*)\right]. \tag{5.12}$$

Let us define $r = (1 - \gamma_\mu/\gamma_4)^{\frac{1}{4}}$. Then for $i \in J$

$$\phi_\mu(x_{i+1}) - \phi_\mu(x_*) \leq r^4\left[\phi_\mu(x_i) - \phi_\mu(x_*)\right]. \tag{5.13}$$

We know that the merit function decreases at each step, and by (5.8) we have, for $j \geq k$ and $k \in U$,

$$\begin{aligned} \|x_j - x_*\| &\leq \gamma_3^{-\frac{1}{2}}(\phi_\mu(x_j) - \phi_\mu(x_*))^{\frac{1}{2}} \\ &\leq \gamma_3^{-\frac{1}{2}}(\phi_\mu(x_k) - \phi_\mu(x_*))^{\frac{1}{2}}. \end{aligned}$$

We continue in this fashion, bounding the right-hand side by terms involving earlier iterates, but using now (5.13) for all good iterates. Since by Theorem 3.1 at least half of the iterates at which updating takes place are good iterates (i.e., $|J_k| \geq \frac{1}{2}|U_k|$), we have

$$\begin{aligned}
\|x_j - x_*\| &\leq \gamma_3^{-\frac{1}{2}} \left[r^{4|J_k|}(\phi_\mu(x_1) - \phi_\mu(x_*)) \right]^{\frac{1}{2}} \\
&\leq \gamma_3^{-\frac{1}{2}} \left[r^{2|U_k|}(\phi_\mu(x_1) - \phi_\mu(x_*)) \right]^{\frac{1}{2}} \\
&\leq [\gamma_3^{-\frac{1}{2}}(\phi_\mu(x_1) - \phi_\mu(x_*))^{\frac{1}{2}}] r^{|U_k|} \\
&\equiv C r^{|U_k|}.
\end{aligned}$$

□

This result implies that if $\{|U_k|/k\}$ is bounded away from zero, then Algorithm II is R-linearly convergent. However, BFGS updating could take place only a finite number of times, in which case this ratio would converge to zero. It is also possible for BFGS updating to take place an infinite number of times, but every time less often, in such a way that $|U_k|/k \rightarrow 0$. We therefore need to examine the iteration more closely.

We make use of the matrix function ψ defined by

$$\psi(B) = tr(B) - \ln(det(B)), \tag{5.14}$$

where *tr* denotes the trace, and *det* the determinant. It can be shown that

$$\ln \operatorname{cond}(B) < \psi(B), \tag{5.15}$$

for any positive definite matrix B (Byrd and Nocedal (1989)). We also make use of the weighted quantities

$$\tilde{y}_k = G_*^{-1/2} y_k, \qquad \tilde{s}_k = G_*^{1/2} s_k, \tag{5.16}$$

$$\tilde{B}_k = G_*^{-1/2} B_k G_*^{-1/2}, \tag{5.17}$$

$$\cos \tilde{\theta}_k = \frac{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k}{\|\tilde{B}_k \tilde{s}_k\| \|\tilde{s}_k\|}, \tag{5.18}$$

and

$$\tilde{q}_k = \frac{\tilde{s}_k^T \tilde{B}_k \tilde{s}_k}{\tilde{s}_k^T \tilde{s}_k}. \tag{5.19}$$

One can show (see Eq. (3.22) of Byrd and Nocedal (1989)) that if B_k is updated by the BFGS formula, then

$$\begin{aligned}
\psi(\tilde{B}_{k+1}) &= \psi(\tilde{B}_k) + \frac{\|\tilde{y}_k\|^2}{\tilde{y}_k^T \tilde{s}_k} - 1 - \ln \frac{\tilde{y}_k^T \tilde{s}_k}{\tilde{s}_k^T \tilde{s}_k} + \ln \cos^2 \tilde{\theta}_k \\
&\quad + \left[1 - \frac{\tilde{q}_k}{\cos^2 \tilde{\theta}_k} + \ln \frac{\tilde{q}_k}{\cos^2 \tilde{\theta}_k} \right].
\end{aligned} \tag{5.20}$$

This expression characterizes the behavior of the BFGS matrices B_k and will be crucial to the analysis of this section. Before we can make use of this relation, however, we need to consider the accuracy of the correction terms. We begin by showing that when finite differences are used to estimate w_k and $\overline{w}_k$, these are accurate to second order.

Lemma 5.5 *If at the iterate x_k, the corrections w_k and $\overline{w}_k$ are computed by the finite-difference formulae (3.1)–(3.2), and if x_k is sufficiently close to a solution point x_* that satisfies Assumptions 5.1, then*

$$w_k = O(\|p_{\rm Y}\|), \tag{5.21}$$

$$\|w_k - Z_*^T W_* Y_k p_{\rm Y}\| = O(\sigma_k \|p_{\rm Y}\|) \tag{5.22}$$

and

$$\|\overline{w}_k - \alpha_k Z_*^T W_* Y_k p_{\rm Y}\| = O(\sigma_k \|p_{\rm Y}\|). \tag{5.23}$$

Proof. Recalling that $\nabla L(x,\lambda) = g(x) + A(x)\lambda$, we have from (3.1) that

$$\begin{aligned} w_k &= Z_k^T[\nabla L(x_k + Y_k p_{\rm Y}, \lambda_k) - \nabla L(x_k, \lambda_k)] \\ &= Z_k^T[\nabla L(x_k + Y_k p_{\rm Y}, \lambda_*) - \nabla L(x_k, \lambda_*)] + Z_k^T[(A(x_k + Y_k p_{\rm Y}) - A_k)(\lambda_k - \lambda_*)] \\ &= Z_k^T \left[\int_0^1 \nabla_{xx}^2 L(x_k + \tau Y_k p_{\rm Y}, \lambda_*) d\tau \right] Y_k p_{\rm Y} + Z_k^T[(A(x_k + Y_k p_{\rm Y}) - A_k)(\lambda_k - \lambda_*)] \\ &\equiv Z_k^T \overline{W}_k Y_k p_{\rm Y} + Z_k^T[(A(x_k + Y_k p_{\rm Y}) - A_k)(\lambda_k - \lambda_*)]. \end{aligned} \tag{5.24}$$

Let us assume that x_k is in the neighborhood of x_* where (5.2)–(5.5) hold. Then $\|\lambda_k - \lambda_*\| = O(\|e_k\|) = O(\sigma_k)$, where σ_k is defined by (1.34). Therefore the last term in (5.24) is $O(\|p_{\rm Y}\|\sigma_k)$, which proves (5.21). Also, a simple computation shows that

$$[Z_k^T \overline{W}_k - Z_*^T W_*] Y_k p_{\rm Y} = O(\sigma_k \|p_{\rm Y}\|). \tag{5.25}$$

Using these facts in (5.24) yields the desired result (5.22). To prove (5.23), we note only that $\alpha_k \le 1$ and reason in the same manner.

□

Next we show that the condition number of the matrices B_k is bounded and that, in the limit, at the iterates U at which BFGS updating takes place, the matrices B_k are accurate approximations of the reduced Hessian of the Lagrangian.

Theorem 5.6 *Suppose that the iterates $\{x_k\}$ generated by Algorithm II converge to a solution point x_* that satisfies Assumptions 5.1. Then $\{\|B_k\|\}$ and $\{\|B_k^{-1}\|\}$ are bounded, and for all $k \in U$*

$$\|(B_k - Z_*^T W_* Z_*) p_{\rm Z}\| = o(\|d_k\|). \tag{5.26}$$

Proof. We will only consider iterates k for which BFGS updating of B_k takes place. We have from (3.49), (3.46), (3.44), (3.16), and (3.48)

$$\begin{aligned} y_k &= Z_k^T[\nabla L(x_{k+1}, \lambda_{k+1}) - \nabla L(x_k, \lambda_{k+1})] - \overline{w}_k \\ &= Z_k^T \left[\int_0^1 \nabla_{xx}^2 L(x_k + \tau \alpha_k d_k, \lambda_{k+1}) d\tau \right] \alpha_k d_k - \overline{w}_k \\ &= \alpha_k Z_k^T \tilde{W}_k (Z_k p_{\rm Z} + Y_k p_{\rm Y}) - \overline{w}_k \\ &= Z_k^T \tilde{W}_k Z_k s_k + \alpha_k (Z_k^T \tilde{W}_k - Z_*^T W_*) Y_k p_{\rm Y} + (\alpha_k Z_*^T W_* Y_k p_{\rm Y} - \overline{w}_k). \end{aligned} \tag{5.27}$$

Since $\overline{w}_k$ can be computed by Broyden's method or by finite differences, we need to consider these two cases separately.

Part I. Let us first assume that $\overline{w}_k$ is determined by Broyden's method. A simple computation shows that $\|Z_k^T\tilde{W}_k - Z_*^T W_*\| = O(\sigma_k)$, and from (3.14) we have that $\overline{w}_k = O(\|p_{\rm Y}\|/\gamma_k)$. Using this and Assumptions 5.1 in (5.27), we have

$$\begin{aligned} y_k &= Z_k^T\tilde{W}_k Z_k s_k + (\sigma_k + 1 + 1/\gamma_k)O(\alpha_k\|p_{\rm Y}\|) \\ &= (Z_k^T\tilde{W}_k Z_k - G_*)s_k + G_* s_k + (\sigma_k + 1 + 1/\gamma_k)O(\alpha_k\|p_{\rm Y}\|). \end{aligned} \tag{5.28}$$

Recalling (5.16) and noting that $\tilde{y}_k^T\tilde{s}_k = y_k^T s_k$, we have

$$\tilde{y}_k^T\tilde{s}_k = s_k^T(Z_k^T\tilde{W}_k Z_k - G_*)s_k + \|\tilde{s}_k\|^2 + (\sigma_k + 1 + 1/\gamma_k)O(\alpha_k\|p_{\rm Y}\|)\|\tilde{s}_k\|,$$

since $\|\tilde{s_k}\|$ and $\|s_k\|$ are of the same order. Therefore

$$\begin{aligned} \frac{\tilde{y}_k^T\tilde{s}_k}{\|\tilde{s}_k\|^2} &= 1 + \frac{s_k^T(Z_k^T\tilde{W}_k Z_k - G_*)s_k}{\|\tilde{s}_k\|^2} + (\sigma_k + 1 + 1/\gamma_k)O\left(\frac{\|\alpha_k p_{\rm Y}\|}{\|\tilde{s}_k\|}\right) \\ &= 1 + O(\sigma_k) + (\sigma_k + 1 + 1/\gamma_k)O\left(\frac{\|\alpha_k p_{\rm Y}\|}{\|\tilde{s}_k\|}\right). \end{aligned} \tag{5.29}$$

Similarly from (5.28) and (5.16) we have

$$\begin{aligned} \tilde{y}_k^T\tilde{y}_k \;\leq\; & \|(Z_k^T\tilde{W}_k Z_k - G_*)s_k\|^2\|G_*^{-1}\| + 2\|(Z_k^T\tilde{W}_k Z_k - G_*)s_k\|\,\|G_*^{-1/2}\|\,\|\tilde{s}_k\| + \|\tilde{s}_k\|^2 \\ & + 2(\sigma_k + 1 + 1/\gamma_k)O(\|\alpha_k p_{\rm Y}\|)\|G_*^{-\frac{1}{2}}\|\left(\|\tilde{s}_k\| + \|(Z_k^T\tilde{W}_k Z_k - G_*)s_k\|\|G_*^{-1/2}\|\right) \\ & + (\sigma_k + 1 + 1/\gamma_k)^2 O(\|\alpha_k p_{\rm Y}\|)^2, \end{aligned}$$

and thus

$$\begin{aligned} \frac{\|\tilde{y}_k\|^2}{\|\tilde{s}_k\|^2} \;\leq\; & 1 + O(\sigma_k) + (\sigma_k + 1 + 1/\gamma_k)(1 + \sigma_k)O\left(\frac{\|\alpha_k p_{\rm Y}\|}{\|\tilde{s}_k\|}\right) \\ & + (\sigma_k + 1 + 1/\gamma_k)^2 O\left(\frac{\|\alpha_k p_{\rm Y}\|^2}{\|\tilde{s}_k\|^2}\right). \end{aligned} \tag{5.30}$$

At this point we invoke the update criterion and note from (3.18) that, if BFGS updating of B_k takes place at iteration k, then $\|\alpha_k p_{\rm Y}\| \leq \gamma_k^2\|s_k\|$, where $\{\gamma_k\}$ is summable. Using this, the assumption that σ_k converges to zero, and (5.29), we see that for large k

$$\frac{\tilde{y}_k^T\tilde{s}_k}{\|\tilde{s}_k\|^2} = 1 + O(\sigma_k + \gamma_k), \tag{5.31}$$

and using (5.30)

$$\frac{\|\tilde{y}_k\|^2}{\|\tilde{s}_k\|^2} = 1 + O(\sigma_k + \gamma_k).$$

Therefore

$$\frac{\|\tilde{y}_k\|^2}{\tilde{y}_k^T\tilde{s}_k} = \frac{\|\tilde{y}_k\|^2}{\|\tilde{s}_k\|^2}\frac{\|\tilde{s}_k\|^2}{\tilde{y}_k^T\tilde{s}_k} = 1 + O(\sigma_k + \gamma_k). \tag{5.32}$$

We now consider $\psi(\tilde{B}_{k+1})$ given by (5.20). A simple expansion shows that for large k, $\ln(1 + O(\sigma_k + \gamma_k)) = O(\sigma_k + \gamma_k)$. Using this, (5.31), and (5.32), we have

$$\psi(\tilde{B}_{k+1}) = \psi(\tilde{B}_k) + O(\sigma_k + \gamma_k) + \ln\cos^2\tilde{\theta}_k + \left[1 - \frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k} + \ln\frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k}\right]. \tag{5.33}$$

Note that for $x \geq 0$ the function $1 - x + \ln x$ is nonpositive, implying that the term in square brackets is nonpositive and that $\ln \cos^2 \tilde{\theta}_k$ is also non-positive. We can therefore delete these terms to obtain

$$\psi(\tilde{B}_{k+1}) \leq \psi(\tilde{B}_k) + O(\sigma_k + \gamma_k). \tag{5.34}$$

Before proceeding further we show that a similar expression holds when finite differences are used.

Part II. Let us now consider the iterates k for which updating takes place and for which $\overline{w}_k$ is computed by finite differences. In this case (3.19) holds. Again we begin by considering (5.27),

$$y_k = Z_k^T \tilde{W}_k Z_k s_k + \alpha_k (Z_k^T \tilde{W}_k - Z_*^T W_*) Y_k p_{\mathrm{Y}} + (\alpha_k Z_*^T W_* Y_k p_{\mathrm{Y}} - \overline{w}_k).$$

Using (5.23) the last term is of order $\sigma_k(\alpha_k \|p_{\mathrm{Y}}\|)$, and so is the second term. Thus

$$\begin{aligned} y_k &= Z_k^T \tilde{W}_k Z_k s_k + O(\sigma_k \alpha_k \|p_{\mathrm{Y}}\|) \\ &= (Z_k^T \tilde{W}_k Z_k - G_*) s_k + G_* s_k + O(\sigma_k \alpha_k \|p_{\mathrm{Y}}\|). \end{aligned} \tag{5.35}$$

Noting once more that $\tilde{y}_k^T \tilde{s}_k = y_k^T s_k$ and recalling the definition (5.16), we have

$$\tilde{y}_k^T \tilde{s}_k = s_k^T (Z_k^T \tilde{W}_k Z_k - G_*) s_k + \|\tilde{s}_k\|^2 + O(\sigma_k \alpha_k \|p_{\mathrm{Y}}\| \|\tilde{s}_k\|),$$

since $\|\tilde{s_k}\|$ and $\|s_k\|$ are of the same order. Therefore

$$\begin{aligned} \frac{\tilde{y}_k^T \tilde{s}_k}{\|\tilde{s}_k\|^2} &= 1 + \frac{s_k^T (Z_k^T \tilde{W}_k Z_k - G_*) s_k}{\|\tilde{s}_k\|^2} + O\left(\sigma_k \frac{\|\alpha_k p_{\mathrm{Y}}\|}{\|\tilde{s}_k\|}\right) \\ &= 1 + O(\sigma_k) + O\left(\sigma_k \frac{\|\alpha_k p_{\mathrm{Y}}\|}{\|\tilde{s}_k\|}\right). \end{aligned} \tag{5.36}$$

Similarly from (5.35) and (5.16) we have

$$\begin{aligned} \tilde{y}_k^T \tilde{y}_k \leq\ & \|(Z_k^T \tilde{W}_k Z_k - G_*) s_k\|^2 \|G_*^{-1}\| + 2\|(Z_k^T \tilde{W}_k Z_k - G_*) s_k\| \, \|G_*^{-1/2}\| \, \|\tilde{s}_k\| + \|\tilde{s}_k\|^2 \\ & + \sigma_k O\left(\|\alpha_k p_{\mathrm{Y}}\| \|G_*^{-\frac{1}{2}}\| \left[\|\tilde{s}_k\| + \|(Z_k^T \tilde{W}_k Z_k - G_*) s_k\| \|G_*^{-1/2}\| \right] \right) \\ & + \sigma_k^2 O(\|\alpha_k p_{\mathrm{Y}}\|)^2, \end{aligned}$$

and thus

$$\frac{\|\tilde{y}_k\|^2}{\|\tilde{s}_k\|^2} \leq 1 + O(\sigma_k) + \sigma_k O\left(\frac{\|\alpha_k p_{\mathrm{Y}}\|}{\|\tilde{s}_k\|}\right) + \sigma_k^2 O\left(\frac{\|\alpha_k p_{\mathrm{Y}}\|^2}{\|\tilde{s}_k\|^2}\right). \tag{5.37}$$

We now invoke Update Criterion I and note from (3.19) that, if BFGS updating of B_k takes place at iteration k, then $\|p_{\mathrm{Y}}\| \leq \gamma_{\mathrm{fd}} \|p_{\mathrm{Z}}\| / \sigma_k^{1/2}$. Using this, (5.36), and the fact that σ_k converges to zero, we see that for large k

$$\frac{\tilde{y}_k^T \tilde{s}_k}{\|\tilde{s}_k\|^2} = 1 + O(\sigma_k^{1/2}),$$

and using (5.37)

$$\frac{\|\tilde{y}_k\|^2}{\|\tilde{s}_k\|^2} = 1 + O(\sigma_k^{1/2}).$$

Therefore

$$\frac{\|\tilde{y}_k\|^2}{\tilde{y}_k^T \tilde{s}_k} = \frac{\|\tilde{y}_k\|^2}{\|\tilde{s}_k\|^2} \frac{\|\tilde{s}_k\|^2}{\tilde{y}_k^T \tilde{s}_k} = 1 + O(\sigma_k^{1/2}). \tag{5.38}$$

We now consider $\psi(\tilde{B}_{k+1})$ given by (5.20). Noting that $\ln(1+O(\sigma_k^{1/2})) = O(\sigma_k^{1/2})$ for all large k, we see that if updating takes place at iteration k

$$\psi(\tilde{B}_{k+1}) = \psi(\tilde{B}_k) + O(\sigma_k^{1/2}) + \ln\cos^2\tilde{\theta}_k + \left[1 - \frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k} + \ln\frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k}\right]. \tag{5.39}$$

Since both $\ln\cos^2\tilde{\theta}_k$ and the term inside the square brackets are nonpositive, we can delete them to obtain

$$\psi(\tilde{B}_{k+1}) \le \psi(\tilde{B}_k) + O(\sigma_k^{1/2}). \tag{5.40}$$

We now combine the results of Parts I and II of this proof. Let us subdivide the set of iterates U for which BFGS updating takes place into two subsets: U' corresponds to the iterates in which $\overline{w}_k$ is computed by Broyden's method, and U'' to the iterates in which finite differences are used. We also define $U'_k = U' \cap \{1,2,...,k\}$ and $U''_k = U'' \cap \{1,2,...,k\}$.

Summing over the set of iterates in U_k, using (5.34) and (5.40), and noting that $B_{j+1} = B_j$ for $j \notin U_k$, we have

$$\psi(\tilde{B}_{k+1}) \le \psi(\tilde{B}_1) + C_1 \sum_{j\in U''_k} \sigma_j^{1/2} + C_2 \sum_{j\in U'_k} \sigma_j + C_3 \sum_{j\in U'_k} \gamma_j, \tag{5.41}$$

for some constants C_1, C_2, C_3. Since $0 \le r \le 1$ and $|U''_j| \le |U_j|$ we have, from (5.11)

$$\begin{aligned} \sum_{j\in U''} \sigma_j^{1/2} &\le \sum_{j\in U''} C^{1/2} r^{|U_j|/2} \\ &\le \sum_{j\in U''} C^{1/2} r^{|U''_j|/2} \\ &= \sum_{i=1}^{|U''|} C^{1/2} r^{i/2} \\ &< \infty. \end{aligned}$$

Similarly

$$\sum_{j\in U'} \sigma_j < \infty,$$

and since $\{\gamma_k\}$ is summable, we conclude from (5.41) that $\{\psi(\tilde{B}_k)\}$ is bounded above. By (5.14) $\psi(\tilde{B}_k) = \sum_{i=1}^n (l_i - \ln l_i)$, where l_i are the eigenvalues of $\tilde{B}_k$, and it is easy to see that this implies that both $\|B_k\|$ and $\|B_k^{-1}\|$ are bounded.

To prove (5.26), we sum relations (5.33) and (5.39), recalling that σ_k, γ_k and $\sigma_k^{1/2}$ are summable, to obtain

$$\psi(\tilde{B}_{k+1}) \le C + \sum_{j\in U_k} \left(\ln\cos^2\tilde{\theta}_k + \left[1 - \frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k} + \ln\frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k}\right]\right),$$

for some constant C. Since $\psi(\tilde{B}_{k+1}) > 0$, and since both $\ln\cos^2\theta_k$ and the term inside the square brackets are nonpositive, we see that

$$\lim_{\substack{k\to\infty \\ k\in U}} \ln\cos^2\tilde{\theta}_k = 0$$

and

$$\lim_{\substack{k\to\infty\\ k\in U}} \left[1 - \frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k} + \ln\frac{\tilde{q}_k}{\cos^2\tilde{\theta}_k}\right] \to 0.$$

Now, for $x \geq 0$ the function $1 - x + \ln x$ is concave and has its unique maximizer at $x = 1$. Therefore the relations above imply that

$$\lim_{\substack{k\to\infty\\ k\in U}} \cos\tilde{\theta}_k = \lim_{\substack{k\to\infty\\ k\in U}} \tilde{q}_k = 1. \tag{5.42}$$

Now from (5.18)–(5.19)

$$\begin{aligned}\frac{\|G_*^{-1/2}(B_k - G_*)p_{\rm Z}\|^2}{\|G_*^{1/2}p_{\rm Z}\|^2} &= \frac{\|(\tilde{B}_k - I)\tilde{s}_k\|^2}{\|\tilde{s}_k\|^2}\\ &= \frac{\|\tilde{B}_k\tilde{s}_k\|^2 - 2\tilde{s}_k^T\tilde{B}_k\tilde{s}_k + \tilde{s}_k^T\tilde{s}_k}{\tilde{s}_k^T\tilde{s}_k}\\ &= \frac{\tilde{q}_k^2}{\cos\tilde{\theta}_k^2} - 2\tilde{q}_k + 1.\end{aligned}$$

It is clear from (5.42) that the last term converges to 0 for $k \in U$, which implies that (5.26) holds. □

This result immediately implies that the iterates are R-linearly convergent, regardless of how often updating takes place.

Theorem 5.7 *Suppose that the iterates $\{x_k\}$ generated by Algorithm II converge to a solution point x_* that satisfies Assumptions 5.1 and the fact that $|U| \to \infty$. Then the rate of convergence is at least R-linear.*

Proof. Theorem 5.6 implies that the condition number of the matrices $\{B_k\}$ is bounded. Therefore, *all* the iterates are good iterates. Reasoning as in the proof of Theorem 5.4, we conclude that for all j

$$\|x_j - x_*\| \leq Cr^j,$$

for some constants $C > 0$ and $0 \leq r < 1$.

□

Prior to considering the convergence rate, we show that the Broyden matrices S_k are bounded.

Lemma 5.8 *Suppose that the iterates $\{x_k\}$ generated by Algorithm II converge R-linearly to a solution point x_* that satisfies Assumptions 5.1. Then the Broyden matrices S_k are bounded and the safeguards (3.13) and (3.14) become inactive for all large k.*

Proof. We make use of the well-known bounded deterioration property for Broyden's method (cf. Lemma 8.2.1 in Dennis and Schnabel (1983)), which states that under Assumptions 5.1

$$\|S_{k+1} - Z_*^TW_*\| \leq \|S_k - Z_*^TW_*\| + C\sigma_k,$$

for some constant $C > 0$. As a result of the R-linear convergence of $\{x_k\}$, we obtain

$$\begin{aligned}\|S_{k+1} - Z_*^TW_*\| &\leq \|S_1 - Z_*^TW_*\| + C\sum_{i=1}^{k}\sigma_k\\ &< \infty,\end{aligned}$$

which shows that the matrices S_k remain bounded. We then see from (3.12) that the Broyden corrections w_k and $\overline{w}_k$ satisfy

$$w_k = O(\|p_{\mathrm{Y}}\|) \quad \overline{w}_k = O(\|p_{\mathrm{Y}}\|), \tag{5.43}$$

and it is clear that the safeguards (3.13) and (3.14) become inactive for all large k. □

Therefore, the algorithm will not modify the information supplied by Broyden's method, asymptotically. This is an important point in establishing superlinear convergence.

6. Superlinear Convergence

Without the correction terms w_k and $\overline{w}_k$, and with appropriate update criteria, Algorithm II is 2-step Q-superlinearly convergent. This was proved by Nocedal and Overton (1985) assuming that Y_k and Z_k are orthogonal bases and assuming that a good starting matrix B_1 is used. This result has been extended by Xie (1991) for more general bases and for any starting matrix $B_1 > 0$. In this section we will show that if the correction terms are used in Algorithm II, the rate of convergence is 1-step Q-superlinear. This result is possible by Update Criterion I and by the selected application of finite-difference approximations, which allow BFGS updating to occur more frequently.

To establish superlinear convergence, we need to ensure that the steplengths α_k have the value 1 for all large k. When a smooth merit function, such as Fletcher's differentiable function (Fletcher (1973)) is used, it is not difficult to show that, near the solution, unit steplengths give a sufficient reduction in the merit function and will be accepted. However, the nondifferentiable ℓ_1 merit function (1.7) used in this paper may reject steplengths of one, even very close to the solution. This so-called Maratos effect requires that the algorithm be modified to allow unit steplengths and to achieve a fast rate of convergence. We will not consider this modification here, so as not to complicate our already lengthy analysis and since it does not affect the main structure of the algorithm or its essential properties. In the companion paper (Biegler, Nocedal, and Schmid (1993)), which is devoted to a numerical investigation of Algorithm II, we describe how to incorporate the nonmonotone line search (or watchdog technique) of Chamberlain et al. (1982) that allows unit steplengths to be accepted for all large k. The analysis of the modified algorithm would be similar to that presented in §5.5 of Byrd and Nocedal (1991).

In the remainder of this section we assume that the iterates generated by Algorithm II converge R-linearly to a solution and that unit steplengths are taken for all large k. In the presentation of the results that follow we do not restate the assumptions under which R-linear convergence was proved in §5, but simply assume that R-linear convergence occurs. We begin by showing that the damping parameter ζ_k, used in (3.43) to ensure that descent directions are always generated, has the value of 1 for all large k.

We have shown in Theorem 5.6 that $\|B_k^{-1}\|$ is bounded above. Also, (5.21), (5.2), and (3.42) show that, when finite differences are used, $w_k = O(\|p_{\mathrm{Y}}\|) = O(\|c_k\|)$, and by (5.43) we see that this is also the case when Broyden's method is used. Using these facts, and noting that $\|\cdot\| \leq \|\cdot\|_1$, we see that there is a constant C such that the left-hand side of (3.38) can be bounded by

$$\zeta_k[2\cos\theta_k|g_k^T Z_k w_k| + w_k^T B_k^{-1} Z_k^T g_k + \zeta_k w_k^T B_k^{-1} w_k] \leq [\zeta_k C(\|e_k\| + \zeta_k\|c_k\|)]\|c_k\|_1,$$

since $g_k^T Z_k = O(\|e_k\|)$. As the iterates converge to the solution, and since $\zeta_k \leq 1$, the term inside the square brackets is less than the constant ρ given in (3.38), showing that $\zeta_k = 1$ for all large k. This and the remarks made at the end of §5 show that all the safeguards included in Algorithm II become inactive asymptotically.

We can now show that the Broyden matrices satisfy the condition of Dennis and Moré (1974) for superlinear convergence. Note from Algorithm II that a Broyden update of S_k is always performed, regardless of whether a BFGS update of B_k takes place or not. The following result is a straightforward modification of a well-known property for Broyden's method.

Lemma 6.1 *Suppose that the iterates generated by Algorithm II converge R-linearly to a point x_* that satisfies Assumptions 5.1. Then*

$$lim_{k\to\infty}\frac{\|(S_k - Z_*^T W_*)d_k\|}{\|d_k\|} = 0. \tag{6.1}$$

Proof. The proof is essentially given in Griewank (1986) and is also very similar to the analysis in Dennis and Schnabel (1983, pp. 183–4), but we will give it here for the sake of completeness. Using the Broyden formula (3.9), we have

$$\begin{aligned} S_{k+1} - Z_*^T W_* &= S_k - Z_*^T W_* + \frac{(\bar{y}_k - S_k \bar{s}_k)\bar{s}_k^T}{\bar{s}_k^T \bar{s}_k} \\ &= S_k - Z_*^T W_* + \frac{(\bar{y}_k - Z_*^T W_* \bar{s}_k)\bar{s}_k^T}{\bar{s}_k^T \bar{s}_k} + \frac{(Z_*^T W_* - S_k)\bar{s}_k \bar{s}_k^T}{\bar{s}_k^T \bar{s}_k} \\ &= (S_k - Z_*^T W_*)(I - \bar{s}_k \bar{s}_k^T / \bar{s}_k^T \bar{s}_k) + (\bar{y}_k - Z_*^T W_* \bar{s}_k)\bar{s}_k^T / \bar{s}_k^T \bar{s}_k. \end{aligned}$$

Defining $E_k = S_k - Z_*^T W_*$, applying Lemma 8.2.5 of Dennis and Schnabel (1983), recalling (3.10)–(3.11), and using the Mean Value theorem, we obtain

$$\begin{aligned} \|E_{k+1}\|_F &\le \|E_k(I - \bar{s}_k \bar{s}_k^T / \bar{s}_k^T \bar{s}_k)\|_F + O(\sigma_k) \\ &\le \|E_k\|_F - \frac{\|E_k \bar{s}_k\|^2}{2\|E_k\|_F \|\bar{s}_k\|^2} + O(\sigma_k). \end{aligned}$$

Rearranging this expression yields

$$\frac{\|E_k \bar{s}_k\|^2}{\|\bar{s}_k\|^2} \le 2\|E_k\|_F \left[\|E_k\|_F - \|E_{k+1}\|_F + O(\sigma_k)\right]. \tag{6.2}$$

By Lemma 5.8, we know that the matrices S_k remain bounded, therefore there exists some Δ such that for all $k \ge \bar{k}$, $\|E_k\| \le \Delta/2$ and

$$\sum_{k=\bar{k}}^{\infty} \frac{\|E_k \bar{s}_k\|^2}{\|\bar{s}_k\|^2} \le \Delta[\|E_{\bar{k}}\|_F + \sum_{k=\bar{k}}^{\infty} O(\sigma_k)].$$

Since $\{\sigma_k\}$ converges R-linearly, the last term is summable, which implies that

$$\lim_{k\to\infty} \frac{\|E_k \bar{s}_k\|^2}{\|\bar{s}_k\|^2} = 0.$$

Noting that $\bar{s}_k = \alpha_k d_k$ gives the desired result.

□

This lemma shows that in the limit S_k is an accurate approximation to $Z_*^T W_*$ along d_k, and Theorem 5.6 shows that, when updating takes place, B_k is an accurate approximation to $Z_*^T W_* Z_*$ along p_Y. We will use these two facts and the following lemma, which is an application of the well-known result of Boggs, Tolle, and Wang (1982).

Lemma 6.2 *Suppose that the iterates generated by Algorithm II converge R-linearly to a point x_* that satisfies Assumptions 5.1, and suppose that $\alpha_k = 1$ for all large k. If, in addition,*

$$\lim_{k\to\infty} \frac{\|B_k p_Z + w_k - Z_*^T W_* d_k\|}{\|d_k\|} = 0, \tag{6.3}$$

then the rate of convergence is 1-step Q-superlinear.

Proof. Nocedal and Overton (1985, Theorem 3.2) show that if an algorithm of the form

$$\begin{bmatrix} \tilde{S}_k \\ A_k^T \end{bmatrix} d_k = -\begin{bmatrix} Z_k^T g_k \\ c_k \end{bmatrix}, \tag{6.4}$$

$$x_{k+1} = x_k + d_k,$$

converges to a point x_* that satisfies Assumptions 5.1, and if

$$\lim_{k\to\infty} \frac{\|(\tilde{S}_k - Z_*^T W_*) d_k\|}{\|d_k\|} = 0, \tag{6.5}$$

then the rate of convergence is superlinear. Algorithm II clearly satisfies the second equation in (6.4), $A_k^T d_k = -c_k$. Now, since $d_k = Y_k p_Y + Z_k p_Z$, we have

$$[Y_k \; Z_k]^{-1} d_k = \begin{bmatrix} p_Y \\ p_Z \end{bmatrix}. \tag{6.6}$$

Let us write $w_k = T_k p_Y$ for some matrix T_k. Then, recalling that $\zeta_k = 1$ for all large k, we have from (3.43) that

$$[T_k \; B_k][Y_k \; Z_k]^{-1} d_k = -Z_k^T g_k.$$

Thus we can define $\tilde{S}_k = [T_k \; B_k][Y_k \; Z_k]^{-1}$, and the condition (6.5) for superlinear convergence is

$$\lim_{k\to\infty} \frac{\|([T_k \; B_k][Y_k \; Z_k]^{-1} - Z_*^T W_*) d_k\|}{\|d_k\|} = 0.$$

However, using (6.6) and $w_k = T_k p_Y$, we have that $[T_k B_k][Y_k Z_k]^{-1} d_k = T_k p_Y + B_k p_Z = w_k + B_k p_Z$, giving the desired result.

□

We can now prove the final result of this section. The analysis is complicated by the fact that BFGS updating may not always take place, and by the fact that the correction terms are sometimes computed by finite differences and sometimes by Broyden's method. We therefore consider the following three sets of iterates, based on Update Criterion I and illustrated in Figure 2.

- $R_1 = \{j | \, \|p_Y^{(j)}\| \le \gamma_j^2 \|p_Z^{(j)}\|\}$,
- $R_2 = \{j \notin R_1 | \, \|p_Y^{(j)}\| \le \|p_Z^{(j)}\| / \sigma_j^{1/2}\}$,
- $R_3 = \{j | \|p_Y^{(j)}\| > \|p_Z^{(j)}\| / \sigma_j^{1/2}\}$,

and note that both γ_k and σ_k are summable.

Theorem 6.3 *Suppose that the iterates generated by Algorithm II converge R-linearly to a point x_* that satisfies Assumptions 5.1, and suppose that $\alpha_k = 1$ for all large k. Then the rate of convergence is 1-step Q-superlinear.*

Proof. Since $d_k = Y_k p_{\rm Y} + Z_k p_{\rm Z}$, we have

$$\begin{bmatrix} p_{\rm Y} \\ p_{\rm Z} \end{bmatrix} = [Y_k \; Z_k]^{-1} d_k.$$

Therefore, Assumption (5.3) implies that

$$\|p_{\rm Y}\| = O(\|d_k\|), \qquad \|p_{\rm Z}\| = O(\|d_k\|). \tag{6.7}$$

Now

$$\begin{aligned} \|B_k p_{\rm Z} + w_k - Z_*^T W_* d_k\| &\leq \|B_k p_{\rm Z} - Z_*^T W_* Z_k p_{\rm Z}\| + \|w_k - Z_*^T W_* Y_k p_{\rm Y}\| \\ &\leq \|B_k p_{\rm Z} - Z_*^T W_* Z_* p_{\rm Z}\| + \|w_k - Z_*^T W_* Y_k p_{\rm Y}\| \\ &\quad + O(\|e_k\| \|p_{\rm Z}\|). \end{aligned}$$

Since by (6.7) the last term is of order $o(\|p_{\rm Z}\|) = o(\|d_k\|)$, the objective of the proof is to show that

$$\|B_k p_{\rm Z} - Z_*^T W_* Z_* p_{\rm Z}\| + \|w_k - Z_*^T W_* Y_k p_{\rm Y}\| = o(\|d_k\|), \tag{6.8}$$

for this together with (6.3) will give the desired result. We consider the three regions R_1, R_2, and R_3 separately. Algorithm II is designed so that, in R_2, w_k must be computed by finite differences. On the other hand, since $p_{\rm Z}$ is recomputed in Step 7, after which we can be in any of the three regions, we see that in R_1 and R_3 w_k may be computed by finite differences or by Broyden.

If $k \in R_1$, we have that $\|p_{\rm Y}\| = o(\|p_{\rm Z}\|) = o(\|d_k\|)$. We also know from (5.43) that $w_k = O(\|p_{\rm Y}\|)$ when the correction is computed by Broyden's method, and by (5.21) this relation also holds when w_k is computed by finite differences. Therefore, for $k \in R_1$,

$$\|w_k - Z_*^T W_* Y_k p_{\rm Y}\| = o(\|d_k\|). \tag{6.9}$$

Furthermore, since updating always takes place in R_1, (5.26) holds:

$$\|B_k p_{\rm Z} - Z_*^T W_* Z_* p_{\rm Z}\| = o(\|d_k\|). \tag{6.10}$$

We have thus established (6.8) for all $k \in R_1$.

Let us now suppose that $k \in R_2$, in which case w_k is computed by finite differences. Using (5.22), we have that

$$\|w_k - Z_*^T W_* Y_k p_{\rm Y}\| = o(\|p_{\rm Y}\|) = o(\|d_k\|), \tag{6.11}$$

where the last step follows from (6.7). Since updating always takes place in R_2, Equation (6.10) also holds in this case, and we conclude that (6.8) holds for all $k \in R_2$.

Finally we consider the case when $k \in R_3$. Now $p_{\rm Z}$ satisfies

$$p_{\rm Z} = o(\|p_{\rm Y}\|) = o(\|d_k\|). \tag{6.12}$$

If $k \in R_3$ and the correction term w_k is computed by Broyden's method as $w_k = S_k Y_k p_{\rm Y}$ (see (3.12)), we have

$$\begin{aligned} \|w_k - Z_*^T W_* Y_k p_{\rm Y}\| &= \|(S_k - Z_*^T W_*) Y_k p_{\rm Y}\| \\ &\leq \|(S_k - Z_*^T W_*) d_k\| + \|(S_k - Z_*^T W_*) Z_k p_{\rm Z}\|. \end{aligned}$$

Using (6.1), (6.12), and the boundedness of S_k, we see that the right-hand side is of order $o(\|d_k\|)$, so that (6.11) holds. On the other hand, if w_k is computed by finite differences, we have directly from (5.22) that (6.11) holds. In addition, (6.12) and the boundedness of B_k show that (6.10) holds for all $k \in R_3$, regardless of whether finite differences or Broyden's method are used.

□

7. Final Remarks

We have presented a new reduced Hessian algorithm for large-scale equality-constrained optimization. The motivation for this work has been practical: our earlier reduced Hessian code, designed for large problems, was often subject to instabilities, and we have aimed to develop a more robust algorithm that resembles the full-space SQP method but is less expensive to implement. In a forthcoming paper (Biegler, Nocedal, and Schmid (1993)), we discuss our computational experience with the new method. That paper describes how to handle inequality constraints and discusses numerous important details of implementation not considered here. These include the choices of all constants and tolerances, the strategy for coping with the case when the basis matrix C in (2.1) changes, and the procedure for computing the damping parameter ζ_k, which was only outlined in (3.39). We also discuss in that paper how to apply the updating criterion away from the solution. We believe that the new algorithm can be very useful for solving large problems, especially those with few degrees of freedom.

We have focused only on convergence results that helped us in the design of the algorithm and that revealed its main properties. The analysis was complicated by two factors. We did not assume that the BFGS matrices B_k or the Broyden matrices S_k were bounded, which required careful consideration of their behavior. This analysis paid off by suggesting safeguards that are useful in practice and ensure a superlinear rate of convergence. The other complicating factor was the fact that the frequency of BFGS updating can vary drastically: it can take place at every iteration, never, or in various patterns. As was found earlier by Xie (1991), it is necessary to develop the theory in sufficient generality to cover all of these cases, and this significantly increased the complexity of some of the results.

Acknowledgments. We thank R. Byrd for many interesting discussions on the subject of this paper. We are also thankful to a referee who made very useful suggestions on how to improve the presentation of the results.

8. *

References

[1] L. T. Biegler, J. Nocedal, and C. Schmid, in preparation, 1993.

[2] P. T. Boggs, J. W. Tolle, and P. Wang, *On the local convergence of a quasi-Newton method for constrained optimization*, SIAM J. Control Optim., 20 (1982), pp. 161–171.

[3] R. H. Byrd, *An example of irregular convergence in some constrained optimization methods that use the projected Hessian*, Math. Programming, 32 (1985), pp. 232–237.

[4] R. H. Byrd, *On the convergence of constrained optimization methods with accurate Hessian information on a subspace*, SIAM J. Numer. Anal., 27 (1990), pp. 141–153.

[5] R. H. Byrd and J. Nocedal, *A tool for the analysis of quasi-Newton methods with application to unconstrained minimization*, SIAM J. Numer. Anal., 26 (1989), pp. 727–739.

[6] R. H. Byrd and J. Nocedal, *An analysis of reduced Hessian methods for constrained optimization*, Math. Programming, 49 (1991), pp. 285–323.

[7] R. H. Byrd, J. Nocedal, and R. B. Schnabel, *Representations of quasi-Newton matrices and their application to limited memory methods*, to appear in Mathematical Programming, 1992.

[8] R. H. Byrd, J. Nocedal, and Y. Yuan, *Global convergence of a class of quasi-Newton methods on convex problems*, SIAM J. Numer. Anal., 24 (1987), pp. 1171–1190.

[9] R. M. Chamberlain, C. Lemarechal, H. C. Pedersen, and M. J. D. Powell, *The watchdog technique for forcing convergence in algorithms for constrained optimization*, Math. Programming Studies, 16 (1982), pp. 1–17.

[10] T. F. Coleman and A. R. Conn, *Nonlinear programming via an exact penalty function: global analysis*, Math. Programming, 24 (1982), pp. 137–161.

[11] T. F. Coleman and A. R. Conn, *On the local convergence of a quasi-Newton method for the nonlinear programming problem*, SIAM J. Numer. Anal., 21 (1984), pp. 755–769.

[12] T. F. Coleman and P. A. Fenyes, *Partitioned quasi-Newton methods for nonlinear equality constrained optimization*, Math. Programming, 53 (1992), pp. 17–44.

[13] A. R. Conn, *Constrained optimization using a nondifferentiable penalty function*, SIAM J. Numer. Anal., 13 (1973), pp. 145–154.

[14] J. E. Dennis and J. J. Moré, *A characterization of superlinear convergence and its application to quasi-Newton methods*, Math. Comp., 28 (1974), pp. 549–560.

[15] J. E. Dennis, jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1983.

[16] R. Fletcher, *An exact penalty for nonlinear programming with inequalities*, Math. Programming, 5 (1973), pp. 129–150.

[17] R. Fletcher, *Practical Methods of Optimization* (second edition), John Wiley and Sons, Chichester, 1987.

[18] D. Gabay, *Reduced quasi-Newton methods with feasibility improvement for nonlinearly constrained optimization*, Math. Programming Studies, 16 (1982), pp. 18–44.

[19] J. C. Gilbert, *On the local and global convergence of a reduced quasi-Newton method*, Optimization, 20 (1989), pp. 421–450.

[20] J. C. Gilbert, *Maintaining the positive definiteness of the matrices in reduced Hessian methods for equality constrained optimization*, Math. Programming, 50 (1991), pp. 1–28.

[21] A. Griewank, *The "global" convergence of Broyden-like methods with a suitable line search*, J. Austral. Math. Soc. Ser. B 28 (1986), pp. 75–92.

[22] C. B. Gurwitz, *Local convergence of a two-piece update of a projected Hessian matrix*, Tech. Report, Department of Computer and Information Science, Brooklyn College, to appear in SIAM J. Optimization, 1993.

[23] C. B. Gurwitz and M. L. Overton, *SQP methods based on approximating a projected Hessian matrix*, SIAM. J. Sci. Stat. Comp., 10 (1989), pp. 631–653.

[24] S. P. Han, *A globally convergent method for nonlinear programming*, J. Optimization Theory and Application, 22/3 (1977), pp. 297–309.

[25] W. Murray and F. J. Prieto, *A sequential quadratic programming algorithm using an incomplete solution of the subproblem*, Tech Report, Department of Operations Research, Stanford University, (1992).

[26] J. Nocedal and M. L. Overton, *Projected Hessian updating algorithms for nonlinearly constrained optimization*, SIAM J. Numer. Anal., 22 (1985), pp. 821–850.

[27] M. J. D. Powell, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, in: O. Mangasarian, R. Meyer, and S. Robinson, eds., Nonlinear Programming, 3, pp. 27–63, Academic Press, New York, 1978.

[28] F. Tagliaferro, *On a quasi-Newton update and its application to equality constrained optimization*, Technical Report, Universita di Trieste, Italy, 1989.

[29] Y. Xie, *Reduced Hessian algorithms for solving large-scale equality constrained optimization problems*, Ph.D. dissertation, Department of Computer Science, University of Colorado, Boulder, 1991.

[30] Y. Yuan, *An only 2-step Q-superlinear convergence example for some algorithms that use reduced Hessian approximations*, Math. Programming, 32 (1985), pp. 224–231.