DAMTP 2008/NA03

On nonlinear optimization since 1959[1]

M.J.D. Powell

Abstract: This view of the development of algorithms for nonlinear optimization is based on the research that has been of particular interest to the author since 1959, including several of his own contributions. After a brief survey of classical methods, which may require good starting points in order to converge successfully, the huge impact of variable metric and conjugate gradient methods is addressed. It encouraged the use of penalty and barrier functions for expressing constrained calculations in unconstrained forms, which are introduced briefly, including the augmented Lagrangian method. Direct methods that make linear approximations to constraints became more popular in the late 1970s, especially sequential quadratic programming, which receives attention too. Sometimes the linear approximations are satisfied only if the changes to the variables are so large that the approximations become unsuitable, which stimulated the development of trust region techniques that make partial corrections to the constraints. That work is also introduced, noting that quadratic models of the objective or Lagrange function do not have to be convex. We consider the sequence of models that is given by the symmetric Broyden updating formula in unconstrained optimization, including the case when first derivatives are not available. The emphasis of the paper is on algorithms that can be applied without good initial estimates of the variables.

Department of Applied Mathematics and Theoretical Physics,
Centre for Mathematical Sciences,
Wilberforce Road,
Cambridge CB3 0WA,
England.

January, 2008.

[1] Presented at the symposium on the Birth of Numerical Analysis, Leuven (October, 2007).

1. Earlier algorithms

The year 1959 is stated in the title of this paper, because Davidon (1959) published then the report that describes his variable metric method for the unconstrained minimization of a general differentiable function, $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, say. That work provides many of the ideas and techniques that are fundamental to later developments, especially the construction and accumulation of useful second derivative information from changes in first derivatives that become available automatically as the calculation proceeds. That information is held in a positive definite matrix, which can give a downhill search direction whenever the gradient $\underline{\nabla} F(\underline{x})$ is nonzero. Thus an initial vector of variables that is close to the solution is not required, and usually the rate of convergence of the iterations of the variable metric method is superlinear. It is a fortunate coincidence that I started my research on numerical analysis in 1959. Therefore, beginning in Section 2, a personal view of major advances in nonlinear optimization during my career is presented.

First we recall some classical foundations of optimization, beginning with Newton's method for solving the nonlinear system of equations $\underline{f}(\underline{x})=0$, where $\underline{f}$ is a continuously differentiable function from $\mathcal{R}^n$ to $\mathcal{R}^n$. For any $\underline{x}_k \in \mathcal{R}^n$, let $J(\underline{x}_k)$ be the Jacobian matrix that has the elements

$$[J(\underline{x}_k)]_{ij} \;=\; df_i(\underline{x}_k)\,/\,dx_j, \qquad 1 \le i,j \le n. \tag{1.1}$$

Then the first order Taylor series provides the approximation

$$\underline{f}(\underline{x}_k + \underline{d}_k) \;\approx\; \underline{f}(\underline{x}_k) + J(\underline{x}_k)\,\underline{d}_k, \qquad \underline{d}_k \in \mathcal{R}^n. \tag{1.2}$$

Newton's method is based on the remark that, if $\underline{d}_k$ is defined by equating the right hand side of this expression to zero, then $\underline{x}_k + \underline{d}_k$ may be a good estimate of a vector that satisfies $\underline{f}(\underline{x}_k + \underline{d}_k)=0$. Indeed, given a starting vector $\underline{x}_1 \in \mathcal{R}^n$, the formula

$$\underline{x}_{k+1} \;=\; \underline{x}_k - J(\underline{x}_k)^{-1}\,\underline{f}(\underline{x}_k), \qquad k=1,2,3,\ldots, \tag{1.3}$$

is applied, assuming every $J(\underline{x}_k)$ is nonsingular. It is well known that, if $\underline{x}^*$ satisfies $\underline{f}(\underline{x}^*)=0$ and if $J(\underline{x}^*)$ is nonsingular, then $\underline{x}_k$ converges at a superlinear rate to $\underline{x}^*$ as $k \to \infty$, provided that $\underline{x}_1$ is sufficiently close to $\underline{x}^*$.

It happens often in practice, however, that such a starting point $\underline{x}_1$ is not available. Then it is highly useful to employ $\underline{d}_k = -J(\underline{x}_k)^{-1}\underline{f}(\underline{x}_k)$ as a search direction, letting $\underline{x}_{k+1}$ be the vector

$$\underline{x}_{k+1} \;=\; \underline{x}_k + \alpha_k\,\underline{d}_k \tag{1.4}$$

for some choice of $\alpha_k > 0$. A usual way of helping convergence is to seek a value of α_k that provides the reduction $\|\underline{f}(\underline{x}_{k+1})\| < \|\underline{f}(\underline{x}_k)\|$ in the Euclidean norm of $\underline{f}$. This strict reduction can be achieved whenever $J(\underline{x}_k)$ is nonsingular and $\|\underline{f}(\underline{x}_k)\|$ is nonzero. One way of establishing this property begins with the remark that the first derivatives at $\alpha = 0$ of the functions $\|\underline{f}(\underline{x}_k + \alpha\,\underline{d}_k)\|^2$, $\alpha \in \mathcal{R}$, and

$\phi(\alpha) = \|\underline{f}(\underline{x}_k) + \alpha\, J(\underline{x}_k)\, \underline{d}_k\|^2$, $\alpha \in \mathcal{R}$, are the same due to the use of the first order Taylor series. Moreover, $\phi(\alpha)$, $\alpha \in \mathcal{R}$, is a nonnegative quadratic that takes the values $\phi(0) = \|\underline{f}(\underline{x}_k)\|^2$ and $\phi(1) = 0$. Thus we deduce the required condition

$$\Big[\frac{d}{d\alpha} \|\underline{f}(\underline{x}_k + \alpha\, \underline{d}_k)\|^2 \Big]_{\alpha=0} \;=\; \phi'(0) \;=\; -2\, \|\underline{f}(\underline{x}_k)\|^2 \;<\; 0. \tag{1.5}$$

Probably this enhancement of Newton's method is also classical. It is easy to show that the line searches may fail to provide $\|\underline{f}(\underline{x}_k)\| \to 0$ as $k \to \infty$, by picking a system of equations that does not have a solution.

Let every α_k in the method of the last paragraph be the value of α that minimizes $\|\underline{f}(\underline{x}_k + \alpha\, \underline{d}_k)\|^2$, $\alpha \geq 0$. It is possible for each iteration to be well-defined, and for $\underline{x}_k$, $k = 1, 2, 3, \ldots$, to converge to a limit $\underline{x}^*$ where the gradient of the function $F(\underline{x}) = \|\underline{f}(\underline{x})\|^2$, $\underline{x} \in \mathcal{R}^n$, is nonzero, but of course $J(\underline{x}^*)$ is singular (Powell, 1970). Then the search direction $\underline{d}_k$ tends to be orthogonal to $\underline{\nabla} F(\underline{x}_k)$ as $k \to \infty$, which is unwelcome when seeking the least value of a differentiable function F.

Such orthogonality is avoided as much as possible in the steepest descent method for minimizing $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, where F is now any continuously differentiable function from $\mathcal{R}^n$ to $\mathcal{R}$. The k-th iteration sets $\underline{d}_k = -\underline{\nabla} F(\underline{x}_k)$, where $\underline{x}_1$ is given and where $\underline{x}_k$, $k \geq 2$, is provided by the previous iteration. Termination occurs if $\|\underline{d}_k\|$ is zero or acceptably small, but otherwise a positive step-length α_k is sought, in order to apply formula (1.4). Typically, values of α_k that are too long or too short are avoided by imposing the conditions

$$\left.\begin{array}{rcl} F(\underline{x}_k + \alpha_k\, \underline{d}_k) & \leq & F(\underline{x}_k) + c_1\, \alpha_k\, \underline{d}_k^T \underline{\nabla} F(\underline{x}_k) \\ \underline{d}_k^T \underline{\nabla} F(\underline{x}_k + \alpha_k\, \underline{d}_k) & \geq & c_2\, \underline{d}_k^T \underline{\nabla} F(\underline{x}_k) \end{array}\right\}, \tag{1.6}$$

where c_1 and c_2 are prescribed constants that satisfy $0 < c_1 < 0.5$ and $c_1 < c_2 < 1$. Termination occurs too if, in the search for α_k, it is found that F is not bounded below. This method has a very attractive convergence property, namely that, if the number of iterations is infinite and if the points $\underline{x}_k$ remain in a bounded region of $\mathcal{R}^n$, then the sequence of gradients $\underline{\nabla} F(\underline{x}_k)$ tends to zero as $k \to \infty$.

Often in practice, however, the steepest descent method is intolerably slow. For example, we let m and M be positive constants and we apply the method to the quadratic function

$$F(\underline{x}) \;=\; m\, x_1^2 + M x_2^2, \qquad \underline{x} \in \mathcal{R}^2, \tag{1.7}$$

starting at the point $\underline{x}_1 = (M, m)^T$. Further, we satisfy the line search conditions (1.6) by letting α_k provide the least value of $F(\underline{x}_{k+1}) = F(\underline{x}_k + \alpha_k\, \underline{d}_k)$ on every iteration. A simple calculation shows that $\underline{x}_{k+1}$ has the components $M\theta^k$ and $m\,(-\theta)^k$, $k = 1, 2, 3, \ldots$, where $\theta = (M - m)/(M + m)$. Thus, if $\nabla^2 F(\underline{x}^*)$ is very ill-conditioned, then a large number of iterations may be required to obtain a vector of variables that is close enough to the solution $\underline{x}^* = 0$. This slow convergence occurs for most starting points $\underline{x}_1$, but our choice of $\underline{x}_1$ simplifies the analytic derivation of $\underline{x}_{k+1}$.

The classical way of achieving a superlinear rate of convergence when minimizing a twice continuously differentiable function $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, is to apply the Newton–Raphson algorithm. In its basic form it is identical to Newton's method for calculating $\underline{x} \in \mathcal{R}^n$ that solves the nonlinear system $\underline{\nabla} F(\underline{x}) = 0$. Putting $\underline{f} = \underline{\nabla} F$ in the definition (1.1) gives a symmetric Jacobian matrix with the elements

$$[G(\underline{x}_k)]_{ij} \;=\; d^2 F(\underline{x}_k) \,/\, dx_i\, dx_j, \qquad 1 \leq i, j \leq n, \tag{1.8}$$

so equation (1.3) takes the form

$$\underline{x}_{k+1} \;=\; \underline{x}_k - G(\underline{x}_k)^{-1}\, \underline{\nabla} F(\underline{x}_k), \qquad k = 1, 2, 3, \ldots\,. \tag{1.9}$$

The line search version (1.4), with $\underline{d}_k = -G(\underline{x}_k)^{-1} \underline{\nabla} F(\underline{x}_k)$, can help convergence sometimes when $\underline{x}_1$ is not sufficiently close to the optimal vector of variables $\underline{x}^*$. Then, as in the given extension to Newton's method, one may seek a step-length α_k that provides the reduction $\|\underline{\nabla} F(\underline{x}_{k+1})\| < \|\underline{\nabla} F(\underline{x}_k)\|$. This approach is objectionable, however, because trying to solve $\underline{\nabla} F(\underline{x}) = 0$ can be regarded as seeking a stationary point of F without paying any attention to minimization. Therefore it may be more suitable to let α_k be an estimate of the value of α that minimizes $F(\underline{x}_k + \alpha\, \underline{d}_k)$, $\alpha \in \mathcal{R}$, but this minimum may occur at $\alpha = 0$, even if $\underline{\nabla} F(\underline{x}_k)$ is nonzero.

The remarks of this section have drawn attention to some major disadvantages of classical methods for optimization. Thus we may be able to appreciate better the gains that have been achieved since 1959.

2. Two major advances in unconstrained optimization

I was fortunate in 1962 to obtain a copy of the report of Davidon (1959), after finding a reference to it in a monograph. The report describes an algorithm for unconstrained minimization, which I programmed for a Ferranti Mercury computer, in order to try some numerical experiments. The results were staggering, especially the minimization of a periodic function $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, with 100 variables, although problems with $n = 20$ were considered large at that time. The k-th iteration requires a vector of variables $\underline{x}_k$, an $n \times n$ positive definite symmetric matrix H_k, and the gradient $\underline{\nabla} F(\underline{x}_k)$, which is available from the $(k-1)$-th iteration for $k \geq 2$. The sequence of iterations is terminated if $\|\underline{\nabla} F(\underline{x}_k)\|$ is sufficiently small, but otherwise formula (1.4) gives the next vector of variables, $\underline{d}_k$ being the search direction $\underline{d}_k = -H_k \underline{\nabla} F(\underline{x}_k)$, which has the downhill property $\underline{d}_k^T \underline{\nabla} F(\underline{x}_k) < 0$, and α_k being a step-length that satisfies the conditions (1.6), usually with $|\underline{d}_k^T \underline{\nabla} F(\underline{x}_k + \alpha_k \underline{d}_k)|$ much less than $|\underline{d}_k^T \underline{\nabla} F(\underline{x}_k)|$. Finally, the iteration replaces H_k by the matrix

$$H_{k+1} \;=\; H_k \;-\; \frac{H_k\, \underline{\gamma}_k\, \underline{\gamma}_k^T H_k}{\underline{\gamma}_k^T H_k\, \underline{\gamma}_k} \;+\; \frac{\underline{\delta}_k\, \underline{\delta}_k^T}{\underline{\delta}_k^T\, \underline{\gamma}_k}, \tag{2.1}$$

where $\underline{\delta}_k = \underline{x}_{k+1} - \underline{x}_k$, where $\underline{\gamma}_k = \underline{\nabla} F(\underline{x}_{k+1}) - \underline{\nabla} F(\underline{x}_k)$, and where the superscript "T" distinguishes a row vector from a column vector. The positive definiteness of H_{k+1} is inherited from H_k, because the second of the conditions (1.6) implies $\underline{\delta}_k^T \underline{\gamma}_k > 0$.

Davidon (1959) explains that, if the objective function F is strictly convex and quadratic, and if each α_k is the value of α that minimizes $F(\underline{x}_k + \alpha\, \underline{d}_k)$, $\alpha > 0$, which is the condition $\underline{d}_k^T \underline{\nabla} F(\underline{x}_k + \alpha_k \underline{d}_k) = 0$, then, in exact arithmetic, the least value of F is calculated after at most n iterations. His arguments include some variable metric points of view, familiar to experts in the theory of relativity, but many researchers including myself do not understand them properly. Therefore other proofs of quadratic termination have been constructed, which depend strongly on the fact that the algorithm with exact line searches gives the conjugacy property $\underline{d}_k^T \nabla^2 F \underline{d}_j = 0$, $j \neq k$, in the quadratic case. Thus the orthogonality conditions

$$\underline{d}_j^T \underline{\nabla} F(\underline{x}_{k+1}) \;=\; 0, \qquad j = 1, 2, \ldots, k, \tag{2.2}$$

are achieved. There are no restrictions on the choices of $\underline{x}_1$ and the symmetric matrix H_1 for the first iteration, except that H_1 must be positive definite.

The brilliant advantage of this algorithm over classical methods is that it can be applied easily to minimize a general differentiable function F, even if a good initial vector of variables $\underline{x}_1$ is not available, and it gives fast convergence when F is quadratic. Not having to calculate second derivatives is welcome, and it brings two more benefits over the Newton–Raphson procedure. Firstly there is no need to devise a remedy for loss of positive definiteness in $\nabla^2 F(\underline{x})$, and secondly the amount of routine work of each iteration is only $\mathcal{O}(n^2)$ instead of $\mathcal{O}(n^3)$. Another attractive property is invariance under linear transformations of the variables. Specifically, let $\underline{x}_k$, $k = 1, 2, 3, \ldots$, and $\underline{z}_k$, $k = 1, 2, 3, \ldots$, be the vectors of variables that are generated when the algorithm is applied to the functions $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, and $F(S^{-1}\underline{z})$, $\underline{z} \in \mathcal{R}^n$, respectively, where S is any constant real $n \times n$ nonsingular matrix. Then, if the initial vector of variables in the second case is $\underline{z}_1 = S\, \underline{x}_1$, if the initial positive definite matrix is changed from H_1 to $S\, H_1 S^T$ for the second case, and if there are no changes to the procedure for choosing each step-length α_k, then the second sequence of variables is $\underline{z}_k = S\, \underline{x}_k$, $k = 1, 2, 3, \ldots$. It follows that good efficiency does not require the variables to be scaled so that their magnitudes are similar. Furthermore, one can simplify the theoretical analysis when F is quadratic by assuming without loss of generality that $\nabla^2 F$ is the unit matrix.

The investigations of Roger Fletcher into Davidon's recent algorithm were similar to my own, so we reported them in a joint paper (Fletcher and Powell, 1963), and the original algorithm has become known as the DFP method. One can view $B_{k+1} = H_{k+1}^{-1}$ as an approximation to $\nabla^2 F$, partly because equation (2.1) gives $H_{k+1} \underline{\gamma}_k = \underline{\delta}_k$, which implies $\underline{\gamma}_k = B_{k+1} \underline{\delta}_k$, while $\underline{\gamma}_k = \nabla^2 F \underline{\delta}_k$ holds when F is quadratic. The matrix $B_{k+1} = H_{k+1}^{-1}$ can be calculated directly from $B_k = H_k^{-1}$, equation (2.1) being equivalent to the formula

$$B_{k+1} \;=\; \Big(I \;-\; \frac{\underline{\gamma}_k\,\underline{\delta}_k^T}{\underline{\delta}_k^T\,\underline{\gamma}_k}\Big)\; B_k \,\Big(I \;-\; \frac{\underline{\delta}_k\,\underline{\gamma}_k^T}{\underline{\delta}_k^T\,\underline{\gamma}_k}\Big) \;+\; \frac{\underline{\gamma}_k\,\underline{\gamma}_k^T}{\underline{\delta}_k^T\,\underline{\gamma}_k}. \tag{2.3}$$

It is sometimes helpful that working with B_k provides the quadratic model

$$F(\underline{x}_k+\underline{d}) \;\approx\; F(\underline{x}_k) \;+\; \underline{d}^T\underline{\nabla}F(\underline{x}_k) \;+\; \tfrac{1}{2}\,\underline{d}^T B_k\,\underline{d}, \qquad \underline{d}\in\mathcal{R}^n. \tag{2.4}$$

Expression (2.3) allows the Cholesky factorization of B_{k+1} to be derived from the Cholesky factorization of B_k in $\mathcal{O}(n^2)$ operations. Thus positive definiteness is preserved in the presence of computer rounding errors, and it is inexpensive to obtain the usual search direction $\underline{d}_k=-H_k\underline{\nabla}F(\underline{x}_k)$ from the linear system $B_k\underline{d}_k=-\underline{\nabla}F(\underline{x}_k)$. A comparison of equations (2.1) and (2.3) suggests the formula

$$H_{k+1} \;=\; \Big(I \;-\; \frac{\underline{\delta}_k\,\underline{\gamma}_k^T}{\underline{\delta}_k^T\,\underline{\gamma}_k}\Big)\; H_k \,\Big(I \;-\; \frac{\underline{\gamma}_k\,\underline{\delta}_k^T}{\underline{\delta}_k^T\,\underline{\gamma}_k}\Big) \;+\; \frac{\underline{\delta}_k\,\underline{\delta}_k^T}{\underline{\delta}_k^T\,\underline{\gamma}_k}. \tag{2.5}$$

If it replaces equation (2.1) in the DFP method, then we have the well-known BFGS method, which is usually faster than the DFP method in practice.

The other major advance in unconstrained optimization that we consider in this section is the conjugate gradient method of Fletcher and Reeves (1964). It can be applied to general differentiable functions $F(\underline{x})$, $\underline{x}\in\mathcal{R}^n$, it is designed to be efficient when F is quadratic, and it has the strong advantage over the variable metric algorithm of not requiring any $n\times n$ matrices. It can be regarded as an extension of the steepest descent method, retaining $\underline{d}_1=-\underline{\nabla}F(\underline{x}_1)$, but the search directions of later iterations have the form

$$\underline{d}_k \;=\; -\underline{\nabla}F(\underline{x}_k) \;+\; \beta_k\,\underline{d}_{k-1}, \qquad k\geq 2, \tag{2.6}$$

where β_k is allowed to be nonzero. Then a line search picks the step-length that provides the new vector of variables (1.4), which completes the description of the k-th iteration except for the choices of α_k and β_k.

These choices are made in a way that achieves the orthogonality conditions (2.2) for each iteration number k when F is a strictly convex quadratic function, assuming exact arithmetic and termination if $\|\underline{\nabla}F(\underline{x}_k)\|=0$ occurs. We satisfy $\underline{d}_k^T\,\underline{\nabla}F(\underline{x}_{k+1})=0$ by letting α_k be the α that minimizes $F(\underline{x}_k+\alpha\underline{d}_k)$, $\alpha>0$, while the $(k{-}1)$-th of the conditions (2.2) defines β_k. Specifically, because the line search of the previous iteration gives $\underline{d}_{k-1}^T\underline{\nabla}F(\underline{x}_k)=0$, we require $\underline{d}_{k-1}^T\{\underline{\nabla}F(\underline{x}_{k+1})-\underline{\nabla}F(\underline{x}_k)\}=0$, which is equivalent to $\{\underline{\nabla}F(\underline{x}_k)-\underline{\nabla}F(\underline{x}_{k-1})\}^T\underline{d}_k=0$ in the quadratic case. It follows from equation (2.6) that β_k should take the value

$$\begin{aligned}\beta_k \;&=\; \{\underline{\nabla}F(\underline{x}_k)-\underline{\nabla}F(\underline{x}_{k-1})\}^T\,\underline{\nabla}F(\underline{x}_k)\Big/\{\underline{\nabla}F(\underline{x}_k)-\underline{\nabla}F(\underline{x}_{k-1}\}^T\,\underline{d}_{k-1}\\ &=\; \{\underline{\nabla}F(\underline{x}_k)-\underline{\nabla}F(\underline{x}_{k-1})\}^T\,\underline{\nabla}F(\underline{x}_k)\Big/\|\underline{\nabla}F(\underline{x}_{k-1})\|^2,\end{aligned} \tag{2.7}$$

the denominator in the second line being derived from exact line searches and the form of $\underline{d}_{k-1}$. The description of the algorithm is now complete when F is quadratic, and we note that $\nabla^2 F$ is not required. Further analysis in this case can establish the first $k-2$ of the conditions (2.2). It exposes not only the conjugacy property $\underline{d}_k^T \nabla^2 F \underline{d}_j = 0$, $j \neq k$, but also that the gradients $\underline{\nabla} F(\underline{x}_k)$, $k = 1, 2, 3, \ldots$, are mutually orthogonal.

The second line of expression (2.7) states the formula for β_k that is preferred by Polak and Ribière (1969) for general F, but Fletcher and Reeves (1964) propose $\beta_k = \|\underline{\nabla} F(\underline{x}_k)\|^2 / \|\underline{\nabla} F(\underline{x}_{k-1})\|^2$. These two choices are equivalent in the theory of the quadratic case, due to the mutual orthogonality of gradients that has been mentioned, but they are quite different for general F, especially if the changes to $\underline{\nabla} F(\underline{x}_k)$ become relatively small as k is increased. The alternative of Polak and Ribière seems to be more efficient in practice, and it is even better to increase their β_k to zero if it becomes negative. Another reason for modifying β_k (or α_{k-1}) is that, if β_k is nonzero, then the conditions (1.6) of the previous iteration may fail to supply the descent property $\underline{d}_k^T \underline{\nabla} F(\underline{x}_k) < 0$.

We see that the conjugate gradient technique is nearly as easy to apply as steepest descents, and usually it provides huge gains in efficiency. The DFP and BFGS algorithms with $H_1 = I$ (the unit matrix) are equivalent to the conjugate gradient method when F is quadratic and all line searches are exact, but the use of the matrices H_k, $k = 1, 2, 3, \ldots$, brings a strong advantage for general F. In order to explain it, we assume that the sequence $\underline{x}_k$, $k = 1, 2, 3, \ldots$, converges to $\underline{x}^*$, say, and that F becomes exactly quadratic only in a neighbourhood of $\underline{x}^*$. The excellent convergence properties of variable metric algorithms are enjoyed automatically when the points $\underline{x}_k$ enter the neighbourhood, without any restrictions on the current $\underline{x}_k$ and the positive definite matrix H_k. On the other hand, the corresponding convergence properties of the conjugate gradient method require a special choice of the initial search direction, $\underline{d}_1 = -\underline{\nabla} F(\underline{x}_1)$ being suitable, except that the implications of this choice would be damaged by the generality of F on the early iterations. The perfect remedy would set $\beta_k = 0$ as soon as the variables $\underline{x}_k$ stay within the neighbourhood, and perhaps on some earlier iterations too. In practice, β_k can be set to zero when, after the most recent steepest descent iteration, a substantial loss of orthogonality in the sequence of gradients $\underline{\nabla} F(\underline{x}_k)$ is observed.

3. Unconstrained objective functions for constrained problems

The methods of Section 2 provide huge improvements over classical algorithms for unconstrained optimization. Therefore it was attractive in the 1960s to include constraints on the variables by modifying the objective functions of unconstrained calculations. In particular, the techniques in the book of Fiacco and McCormick (1968) were very popular. Some of them are addressed below.

Let the least value of $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, be required, subject to the inequality constraints

$$c_i(\underline{x}) \;\geq\; 0, \qquad i=1,2,\ldots,m. \tag{3.1}$$

Then a typical objective function of a barrier method has the form

$$\left.\begin{array}{rrcl} & \Phi(\underline{x},\mu) & = & F(\underline{x})+\mu \sum_{i=1}^m \{c_i(\underline{x})\}^{-1} \\ \text{or} & \Phi(\underline{x},\mu) & = & F(\underline{x})-\mu \sum_{i=1}^m \log\{c_i(\underline{x})\} \end{array}\right\}, \quad \underline{x}\in\mathcal{R}^n, \tag{3.2}$$

if $\underline{x}$ satisfies all the constraints (3.1) as strict inequalities, but otherwise $\Phi(\underline{x},\mu)$ is defined to be $+\infty$. Here μ is a positive parameter that remains fixed during the unconstrained minimization of $\Phi(\underline{x},\mu)$, $\underline{x}\in\mathcal{R}^n$. The starting point $\underline{x}_1$ of this calculation has to satisfy $c_i(\underline{x}_1)>0$, $i=1,2,\ldots,m$, because $\Phi(\underline{x}_1,\mu)$ is required to be finite. Let $\underline{x}[\mu]$ be the vector of variables that is produced by this calculation.

The constraints (3.1) are also satisfied as strict inequalities at $\underline{x}[\mu]$, because the unconstrained algorithm provides $\Phi(\underline{x}[\mu],\mu)\leq\Phi(\underline{x}_1,\mu)$ automatically, but it is usual for the solution, $\underline{x}^*$ say, of the original problem to be on the boundary of the feasible region. In this case, the theory of barrier methods requires F and c_i, $i=1,2,\ldots,m$, to be continuous functions, and it requires every neighbourhood of $\underline{x}^*$ to include a strictly interior point of the feasible region. Then it is straightforward to establish $F(\underline{x}[\mu]) < F(\underline{x}^*)+\varepsilon$ for sufficiently small μ, where ε is any positive constant, assuming that $\Phi(\underline{x}[\mu],\mu)$ is sufficiently close to the least value of $\Phi(\underline{x},\mu)$, $\underline{x}\in\mathcal{R}^n$.

Equality constraints, however, cannot be included in barrier function methods, because they cannot be satisfied as strict inequalities. Therefore, when minimizing $F(\underline{x})$, $\underline{x}\in\mathcal{R}^n$, subject to the conditions

$$c_i(\underline{x}) \;=\; 0, \qquad i=1,2,\ldots,m, \tag{3.3}$$

it was usual to apply an algorithm for unconstrained minimization to the function

$$\left.\begin{array}{rrcl} & \Phi(\underline{x},\mu) & = & F(\underline{x})+\mu^{-1} \sum_{i=1}^m \{c_i(\underline{x})\}^{2} \\ \text{or} & \Phi(\underline{x},\mu) & = & F(\underline{x})+\mu^{-1} \sum_{i=1}^m |c_i(\underline{x})| \end{array}\right\}, \quad \underline{x}\in\mathcal{R}^n, \tag{3.4}$$

where μ is still a positive parameter, fixed during each unconstrained calculation, that has to become sufficiently small. A new difficulty is shown by the minimization of $F(x)=x^3$, $x\in\mathcal{R}$, subject to $x=1$, namely that, for any fixed $\mu>0$, the functions (3.4) are not bounded below. On the other hand, if $\underline{x}[\mu]$ is the minimizer of $\Phi(\underline{x},\mu)$, $\underline{x}\in\mathcal{R}^n$, if the points $\underline{x}[\mu]$, $\mu>0$, all lie in a compact region of $\mathcal{R}^n$, and if the objective and constraint functions are continuous, then all limit points of the sequence $\underline{x}[\mu]$ as $\mu\rightarrow 0$ are solutions of the original problem. The two main ingredients in a proof of this assertion are that the constraints are satisfied at the limit points, and that, for every positive μ, $F(\underline{x}[\mu])$ is a lower bound on the required value of F.

Penalty function methods are also useful for inequality constraints. If $c_i(\underline{x})=0$ were replaced by $c_i(\underline{x})\geq 0$, then, in expression (3.4), it would be suitable to replace the terms $\{c_i(\underline{x})\}^2$ and $|c_i(\underline{x})|$ by $\{\min[0,c_i(\underline{x})]\}^2$ and $\max[0,-c_i(\underline{x})]$, respectively.

The dependence of the error $\underline{x}[\mu]-\underline{x}^*$ on μ, for both the inequality and equality constrained problems that have been mentioned, can be investigated by comparing the condition for an unconstrained minimum of $\Phi(\underline{x},\mu)$, $\underline{x}\in\mathcal{R}^n$, with the KKT conditions for a solution of the original problem. We consider this approach briefly when the constraints are the equations (3.3), when Φ is the first of the functions (3.4), when the objective and constraint functions have continuous first derivatives, when $\underline{\nabla}F(\underline{x}^*)$ is nonzero, and when the constraint gradients $\underline{\nabla}c_i(\underline{x}^*)$, $i=1,2,\ldots,m$, are linearly independent. Then $\underline{\nabla}\Phi(\underline{x}[\mu],\mu)=0$ is the equation

$$\underline{\nabla}F(\underline{x}[\mu]) \;+\; 2\,\mu^{-1}\textstyle\sum_{i=1}^m c_i(\underline{x}[\mu])\,\underline{\nabla}c_i(\underline{x}[\mu]) \;=\; 0, \tag{3.5}$$

while the first order KKT conditions include the existence of unique Lagrange multipliers $\lambda_i^*\in\mathcal{R}$, $i=1,2,\ldots,m$, not all zero, such that $\underline{\nabla}F(\underline{x}^*)$ can be expressed in the form

$$\underline{\nabla}F(\underline{x}^*) \;=\; \textstyle\sum_{i=1}^m \lambda_i^*\,\underline{\nabla}c_i(\underline{x}^*). \tag{3.6}$$

Therefore, if $\underline{x}[\mu]$ tends to $\underline{x}^*$ as expected when $\mu\rightarrow 0$, we have the estimates $c_i(\underline{x}[\mu])\approx-\frac{1}{2}\mu\lambda_i^*$, $i=1,2,\ldots,m$. It follows that the distance from $\underline{x}[\mu]$ to any point in $\mathcal{R}^n$ that satisfies the constraints is at least of magnitude μ. Typically, $\|\underline{x}[\mu]-\underline{x}^*\|$ is also of this magnitude, but there are exceptions, such as the minimization of $x_1^4+x_1x_2+x_2$, $\underline{x}\in\mathcal{R}^2$, subject to $x_2=0$.

The efficiency of these barrier and penalty function methods depends strongly on suitable stopping conditions for the unconstrained calculations, on the size of the reductions in μ, and on obtaining a good starting vector and second derivative estimates for each new unconstrained problem from the sequence of unconstrained problems that have been solved already. Much attention has been given to these questions recently, because the path $\underline{x}[\mu]$, $\mu>0$, in $\mathcal{R}^n$ is a part of the central path of a primal-dual algorithm (see Nocedal and Wright, 1999, for instance). In the early 1970s, however, barrier and penalty function methods became unpopular, due to the development of new techniques for constraints that avoid the difficulties that arise when μ is tiny. In particular, the functions $\Phi(\underline{x},\mu)$, $\underline{x}\in\mathcal{R}^n$, tend to have some huge first derivatives, so a descent method for unconstrained minimization can reach the bottom of a cliff easily. Then the remainder of the route to $\underline{x}[\mu]$ has to stay at the bottom of the cliffs that are caused by the barrier or penalty terms, which is a daunting situation, especially if the constraints are nonlinear.

The augmented Lagrangian method, proposed by Hestenes (1969) and Powell (1969) independently, is a highly useful extension to the minimization of the first of the functions (3.4), when seeking the least value of $F(\underline{x})$, $\underline{x}\in\mathcal{R}^n$, subject to the equality constraints (3.3). The new penalty function has the form

$$\Lambda(\underline{x},\underline{\lambda},\mu) \;=\; F(\underline{x}) \;-\; \textstyle\sum_{i=1}^m \lambda_i\,c_i(\underline{x}) \;+\; \mu^{-1}\textstyle\sum_{i=1}^m \{c_i(\underline{x})\}^2, \qquad \underline{x}\in\mathcal{R}^n, \tag{3.7}$$

its unconstrained minimum being calculated approximately for each fixed choice of the parameters $\underline{\lambda}\in\mathcal{R}^m$ and $\mu>0$. Let this calculation give the vector of variables $\underline{x}[\underline{\lambda},\mu]$. The main feature of the augmented Lagrangian method is that it tries to satisfy the constraints $c_i(\underline{x}[\underline{\lambda},\mu])=0$, $i=1,2,\ldots,m$, by adjusting $\underline{\lambda}\in\mathcal{R}^m$ without further reductions in μ when μ becomes sufficiently small. It follows from equation (3.6) and from the assumed linear independence of the constraint gradients that, if $\mu>0$, then the solution $\underline{x}^*$ of the original problem is at the unconstrained minimum of the function (3.7) only if $\underline{\lambda}$ has the components $\lambda_i=\lambda_i^*$, $i=1,2,\ldots,m$.

In the awkward problem that has been mentioned of minimizing $x_1^4+x_1x_2+x_2$, $\underline{x}\in\mathcal{R}^2$, subject to $x_2=0$, we find $\lambda_1^*=1$ and that expression (3.7) is the function

$$\Lambda(\underline{x},\underline{\lambda}^*,\mu) \;=\; x_1^4+x_1\,x_2+\mu^{-1}\,x_2^2, \qquad \underline{x}\in\mathcal{R}^2, \tag{3.8}$$

which is stationary at the required solution $\underline{x}^*=0$. Unfortunately this stationary point is never a minimum when μ is fixed and positive. Usually, however, the given problem satisfies the second order condition $\underline{d}^T\{\nabla^2F(\underline{x}^*)-\sum_{i=1}^m\lambda_i^*\nabla^2c_i(\underline{x}^*)\}\underline{d}>0$, where $\underline{d}$ is any nonzero vector that is orthogonal to $\underline{\nabla}c_i(\underline{x}^*)$, $i=1,2,\ldots,m$. In this case, the function (3.7) with $\underline{\lambda}=\underline{\lambda}^*$ is not only stationary at $\underline{x}=\underline{x}^*$, but also the second derivative matrix $\nabla^2\Lambda(\underline{x}^*,\underline{\lambda}^*,\mu)$ is positive definite for sufficiently small μ. It follows that $\underline{x}^*$ can be calculated by the unconstrained minimization of $\Lambda(\underline{x},\underline{\lambda}^*,\mu)$, $\underline{x}\in\mathcal{R}^n$.

The initial choice of μ and any later reductions should provide suitable local minima in the unconstrained calculations and should help the achievement of $\underline{\lambda}\rightarrow\underline{\lambda}^*$. Usually the components of $\underline{\lambda}$ are set to zero initially. A convenient way of adjusting $\underline{\lambda}$ is based on the remark that, if $\underline{x}$ is a stationary point of the function (3.7), then it satisfies the equation

$$\underline{\nabla}\Lambda(\underline{x},\underline{\lambda},\mu) \;=\; \underline{\nabla}F(\underline{x}) \;-\; \textstyle\sum_{i=1}^m\{\lambda_i-2\,\mu^{-1}\,c_i(\underline{x})\}\,\underline{\nabla}c_i(\underline{x}) \;=\; 0. \tag{3.9}$$

Specifically, a comparison of equations (3.6) and (3.9) suggests the formula

$$\lambda_i \;\leftarrow\; \lambda_i-2\,\mu^{-1}\,c_i(\underline{x}[\underline{\lambda},\mu]), \qquad i=1,2,\ldots,m, \tag{3.10}$$

where "$\leftarrow$" denotes "is replaced by ". The success of this technique requires μ to be sufficiently small. Other techniques for updating $\underline{\lambda}$ have been derived from the remark that $\underline{\lambda}^*$ should be the value of $\underline{\lambda}$ that maximizes $\Lambda(\underline{x}[\underline{\lambda},\mu],\underline{\lambda},\mu)$, $\underline{\lambda}\in\mathcal{R}^m$. Indeed, the calculation of $\underline{x}[\underline{\lambda},\mu]$ should provide the bound

$$\Lambda(\underline{x}[\underline{\lambda},\mu],\underline{\lambda},\mu) \;\leq\; \Lambda(\underline{x}^*,\underline{\lambda},\mu) \;=\; F(\underline{x}^*) \;=\; \Lambda(\underline{x}^*,\underline{\lambda}^*,\mu) \tag{3.11}$$

for every choice of $\underline{\lambda}$, the last two equations being elementary consequences of the constraints $c_i(\underline{x}^*)=0$, $i=1,2,\ldots,m$.

The augmented Lagrangian method became even more useful when Rockafellar (1973) proposed and analysed a version of expression (3.7) that is suitable for inequality constraints. Specifically, when the original problem is the minimization of $F(\underline{x})$, $\underline{x}\in\mathcal{R}^n$, subject to the conditions (3.1), then $\underline{x}[\underline{\lambda},\mu]$ is calculated by applying an algorithm for unconstrained minimization to the function

$$\Lambda(\underline{x},\underline{\lambda},\mu) \;=\; F(\underline{x})+\mu^{-1}\textstyle\sum_{i=1}^m\{\min[0,c_i(\underline{x})-\frac{1}{2}\mu\lambda_i]\}^2, \qquad \underline{x}\in\mathcal{R}^n, \tag{3.12}$$

for a sequence of fixed values of $\underline{\lambda} \in \mathcal{R}^m$ and $\mu \in \mathcal{R}$. Again the constraints are satisfied by adjusting only $\underline{\lambda}$ if possible for sufficiently small μ. We see that, if $\underline{x}$ is a stationary point of the new $\Lambda(\underline{x}, \underline{\lambda}, \mu)$, $\underline{x} \in \mathcal{R}^n$, then it satisfies the equation

$$\underline{\nabla} F(\underline{x}) \;-\; \textstyle\sum_{i=1}^m \max[0, \lambda_i - 2\mu^{-1} c_i(\underline{x})] \, \underline{\nabla} c_i(\underline{x}) = 0. \tag{3.13}$$

Therefore we modify formula (3.10) for adjusting $\underline{\lambda}$ by letting $\max[0, \lambda_i – 2\mu^{-1} c_i(\underline{x})]$ at $\underline{x} = \underline{x}[\underline{\lambda}, \mu]$ be the new right hand side. Thus the components of $\underline{\lambda}$ are nonnegative, as required in the KKT condition (3.6) of the original problem when the constraints are inequalities. Further, λ_i^* should be zero in equation (3.6) for every i that satisfies $c_i(\underline{x}^*) > 0$, and, if μ is sufficiently small, the modification of formula (3.10) gives λ_i this property automatically. A mixture of equality and inequality constraints can be treated by taking their contributions to $\Lambda(\underline{x}, \underline{\lambda}, \mu)$ from expressions (3.7) and (3.12), respectively.

4. Sequential quadratic programming

Often the methods of the last section are too elaborate and too sophisticated. An extreme example is the minimization of $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, subject to $x_n = 0$. The constraint allows the number of variables to be decreased by one, and then a single unconstrained calculation with n–1 variables can be solved, instead of a sequence of unconstrained calculations with n variables. The sequence of subproblems can also be avoided when the constraints are nonlinear by making linear approximations to the constraints. In particular, if the least value of $F(\underline{x})$ is required subject to the equality constraints (3.3), and if the objective and constraint functions have continuous second derivatives, then one can apply Newton's method for solving nonlinear equations to the system that is given by the first order KKT conditions at the solution. That approach has several disadvantages. Many of them were removed by the development of sequential quadratic programming (SQP), which is addressed below, because SQP became a popular successor to the augmented Lagrangian method for constrained calculations in the late 1970s.

In the application of Newton's method that has just been mentioned, the unknowns are not only the variables x_i, $i = 1, 2, \ldots, n$, but also the Lagrange multipliers of condition (3.6). Specifically, we seek vectors $\underline{x} \in \mathcal{R}^n$ and $\underline{\lambda} \in \mathcal{R}^m$ that satisfy the square system of equations

$$\left. \begin{array}{r} \underline{\nabla} F(\underline{x}) - \sum_{i=1}^m \lambda_i \underline{\nabla} c_i(\underline{x}) = 0 \quad \text{and} \\ -c_i(\underline{x}) = 0, \qquad i = 1, 2, \ldots, m, \end{array} \right\} \tag{4.1}$$

the signs of the equality constraints (3.3) being reversed in order that the Jacobian matrix of Newton's method is symmetric. Let $\underline{f}(\underline{x}, \underline{\lambda})$, $\underline{x} \in \mathcal{R}^n$, $\underline{\lambda} \in \mathcal{R}^m$, be the vector in $\mathcal{R}^{m+n}$ whose components are the left hand sides of expression (4.1). As in Section 1, the k-th iteration of Newton's method without line searches calculates $\underline{x}_{k+1} = \underline{x}_k + \underline{d}_k$ and $\underline{\lambda}_{k+1} = \underline{\lambda}_k + \underline{\eta}_k$ by equating to zero a first order Taylor series approximation to the function $\underline{f}(\underline{x}_k+\underline{d},\underline{\lambda}_k+\underline{\eta})$, $\underline{d}\in\mathcal{R}^n$, $\underline{\eta}\in\mathcal{R}^m$. Specifically, the analogue of equation (1.3) is that $\underline{d}_k$ and $\underline{\eta}_k$ are derived from the linear system

$$\left(\begin{array}{c|c} W(\underline{x}_k,\underline{\lambda}_k) & -J(\underline{x}_k)^T \\ \hline -J(\underline{x}_k) & 0 \end{array}\right)\left(\begin{array}{c} \underline{d}_k \\ \hline \underline{\eta}_k \end{array}\right) = \left(\begin{array}{c} -\underline{\nabla}F(\underline{x}_k)+J(\underline{x}_k)^T\underline{\lambda}_k \\ \hline \underline{c}(\underline{x}_k) \end{array}\right), \qquad (4.2)$$

where $W(\underline{x},\underline{\lambda})=\nabla^2F(\underline{x})-\sum_{i=1}^m \lambda_i\nabla^2c_i(\underline{x})$, where $J(\underline{x})$ is now the $m\times n$ matrix that has the elements

$$[J(\underline{x})]_{ij} \;=\; dc_i(\underline{x})/dx_j, \quad 1\le i\le m, \quad 1\le j\le n, \qquad (4.3)$$

and where $\underline{c}(\underline{x})$ is the vector in $\mathcal{R}^m$ with the components $c_i(\underline{x})$, $i=1,2,\ldots,m$.

This application of Newton's method has the following three disadvantages. The calculation breaks down if the partitioned matrix of the linear system (4.2) becomes singular. No attempt is made to help convergence when good initial values of the variables are not available. The minimization ingredient of the original problem is absent from the formulation (4.1). On the other hand, the method provides a highly useful answer to a very important question, which is to identify the second derivatives that are usually sufficient for a fast rate of convergence. We see that the k-th iteration in the previous paragraph requires second derivatives of the objective and constraint functions only to assemble the matrix $W(\underline{x}_k,\underline{\lambda}_k)$. Therefore, when second derivatives are estimated, one should construct an approximation to the combination $\nabla^2F(\underline{x})-\sum_{i=1}^m \lambda_i\nabla^2c_i(\underline{x})$, which is much more convenient than estimating all the matrices $\nabla^2F(\underline{x})$ and $\nabla^2c_i(\underline{x})$, $i=1,2,\ldots,m$, separately.

We recall from Section 2 that variable metric algorithms for unconstrained optimization bring huge advantages over the Newton–Raphson method by working with positive definite approximations to ∇^2F. Similar gains can be achieved in constrained calculations over the Newton iteration above by making a positive definite approximation to $W(\underline{x}_k,\underline{\lambda}_k)$ in the system (4.2). We let B_k be such an approximation, and we consider the minimization of the strictly convex quadratic function

$$Q_k(\underline{x}_k+\underline{d}) \;=\; F(\underline{x}_k)+\underline{d}^T\underline{\nabla}F(\underline{x}_k)+\tfrac{1}{2}\,\underline{d}^TB_k\,\underline{d}, \qquad \underline{d}\in\mathcal{R}^n, \qquad (4.4)$$

subject to the linear constraints

$$c_i(\underline{x}_k)+\underline{d}^T\underline{\nabla}c_i(\underline{x}_k) \;=\; 0, \qquad i=1,2,\ldots,m, \qquad (4.5)$$

still assuming that the constraint gradients are linearly independent. The vector $\underline{d}=\underline{d}_k$ is the solution to this problem if and only if it satisfies the constraints (4.5) and the gradient $\underline{\nabla}Q_k(\underline{x}_k+\underline{d}_k)=\underline{\nabla}F(\underline{x}_k)+B_k\underline{d}_k$ is in the linear space spanned by $\underline{\nabla}c_i(\underline{x}_k)$, $i=1,2,\ldots,m$. In other words, $\underline{d}_k$ has to satisfy the equations (4.2) with $W(\underline{x}_k,\underline{\lambda}_k)$ replaced by B_k. Thus the calculation of $\underline{d}_k$ by the Newton iteration is equivalent to the solution of the strictly convex quadratic programming problem, which captures the minimization ingredient that has been mentioned. A more important benefit of the alternative calculation of $\underline{d}_k$ is that it has a natural extension for inequality constraints, by continuing to let $\underline{d}_k$ be the vector $\underline{d}$ that minimizes the strictly convex quadratic function (4.4) subject to first order Taylor series approximations to all the constraints. Specifically, for each constraint index i, the original constraint $c_i(\underline{x})=0$ or $c_i(\underline{x})\geq 0$ contributes the condition $c_i(\underline{x}_k)+\underline{d}^T\underline{\nabla} c_i(\underline{x}_k)=0$ or $c_i(\underline{x}_k)+\underline{d}^T\underline{\nabla} c_i(\underline{x}_k)\geq 0$, respectively, to the quadratic programming problem, without any change to $Q_k(\underline{x}_k+\underline{d})$, $\underline{d}\in\mathcal{R}^n$, after B_k has been chosen.

The DFP formula (2.3) (or the well-known BFGS formula) may be used to define B_{k+1} for the next iteration, where $\underline{\delta}_k$ is the step $\underline{x}_{k+1}-\underline{x}_k$ as before, but the selection of $\underline{\gamma}_k$ requires further consideration. The updating formula gives $B_{k+1}\underline{\delta}_k=\underline{\gamma}_k$, so $\underline{\gamma}_k$ must satisfy $\underline{\delta}_k^T\underline{\gamma}_k>0$, in order that B_{k+1} inherits positive definiteness from B_k. On the other hand, because B_{k+1} should be an estimate of the combination $\nabla^2 F(\underline{x}_{k+1})-\sum_{i=1}^m \lambda_i \nabla^2 c_i(\underline{x}_{k+1})$, as mentioned already, it it suitable to let the difference

$$\widehat{\underline{\gamma}}_k \;=\; \underline{\nabla} F(\underline{x}_{k+1})-\underline{\nabla} F(\underline{x}_k)-\textstyle\sum_{i=1}^m \lambda_i\,\{\underline{\nabla} c_i(\underline{x}_{k+1})-\underline{\nabla} c_i(\underline{x}_k)\} \qquad (4.6)$$

be a provisional choice of $\underline{\gamma}_k$, where the multipliers λ_i, $i=1,2,\ldots,m$, can be taken from the quadratic programming problem that defines $\underline{d}_k$, even if some of the constraints are inequalities. It is possible, however, for the original problem to be the minimization of $F(\underline{x})=-\frac{1}{2}\|\underline{x}\|^2$, $\underline{x}\in\mathcal{R}^n$, subject to constraints that are all linear. Then equation (4.6) gives $\widehat{\underline{\gamma}}_k=-\underline{x}_{k+1}+\underline{x}_k=-\delta_k$, which implies $\underline{\delta}_k^T\widehat{\underline{\gamma}}_k<0$, although we require $\underline{\delta}_k^T\underline{\gamma}_k>0$. Therefore the form $\underline{\gamma}_k=\theta_k\widehat{\underline{\gamma}}_k+(1-\theta_k)B_k\underline{\delta}_k$ is proposed in Powell (1978) for the DFP or BFGS updating formula, where θ_k is the largest number from $[0,1]$ that satisfies $\underline{\delta}_k^T\underline{\gamma}_k\geq 0.1\,\underline{\delta}_k^T B_k\underline{\delta}_k$. A device of this kind was necessary in order to provide software.

Another challenge for SQP software is forcing convergence from poor starting points. A remedy in Section 1 is to seek $\underline{x}_{k+1}$ by a line search from $\underline{x}_k$ along the direction $\underline{d}_k$, but, if all the early iterations require tiny step-lengths, then the progress towards constraint boundaries is very slow, even if the constraints are linear. Therefore some implementations of the SQP method employ two kinds of changes to the variables, namely horizontal and vertical steps, where horizontal steps include line searches and try to reduce the objective function without worsening constraint violations, and where the main purpose of vertical steps is to correct the departures from feasibility (see Coleman and Conn, 1982, for instance). Several techniques have also been proposed for deciding whether or not to accept a trial step in a line search, the difficulty being that improvements in the objective function and decreases in constraint violations may not occur together. The usual compromise is to seek a reduction in the penalty function

$$\Phi(\underline{x},\mu) \;=\; F(\underline{x})+\mu^{-1}\{\,\textstyle\sum_{i\in\mathcal{E}}|c_i(\underline{x})|+\sum_{i\in\mathcal{I}}\max\,[0,-c_i(\underline{x})]\,\}, \qquad (4.7)$$

where $\mathcal{E}$ and $\mathcal{I}$ contain the indices of the equality and inequality constraints, respectively, and where μ has to be selected automatically. Alternatively, instead of taking dubious decisions in the line searches, one can keep options open by applying the filter method of Fletcher and Leyffer (2002). Many different versions of the SQP method have been developed for constrained calculations when first derivatives are available, and usually they are excellent at keeping down the total number of function and gradient evaluations.

5. Trust region methods

We recall that, in line search methods for forcing convergence from general starting points, the sequence of iterations gives the variables

$$\underline{x}_{k+1} \;=\; \underline{x}_k + \alpha_k\, \underline{d}_k, \qquad k=1,2,3,\ldots, \tag{5.1}$$

where usually the search direction $\underline{d}_k$ is derived from a simple model of the original problem, and where the choice of the step-length α_k should make $\underline{x}_{k+1}$ better than $\underline{x}_k$ according to the criteria of the original problem, the simplest example being the condition $F(\underline{x}_{k+1}) < F(\underline{x}_k)$ when the least value of $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, is required. We expect the model of the k-th iteration to provide useful accuracy in a neighbourhood of $\underline{x}_k$, but $\underline{x}_k + \underline{d}_k$ may be far from that neighbourhood, so often the step-lengths of line search methods are substantially less than one for many consecutive iterations. Then it is reasonable to take the view that each new value of $\|\underline{x}_{k+1} - \underline{x}_k\|$ is not going to be much larger than the magnitudes of the changes to the variables of recent iterations. Under this assumption, one may be able to make much better use of the simple model. For example, moves to constraint boundaries can be made more quickly in the situation that is mentioned in the last paragraph of Section 4. Therefore a bound of the form $\|\underline{d}_k\| \leq \Delta_k$ is imposed by a trust region method, the remaining freedom in $\underline{d}_k$ being taken up by consideration of the current simple model. The positive parameter Δ_k is chosen automatically before the start of the k-th iteration. Some details and advantages of this technique are addressed below, because, since the 1970s, trust region methods have become fundamental within many highly successful algorithms for optimization.

We begin with the unconstrained minimization of $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, when first derivatives are available, and when the calculation of $\underline{x}_{k+1}$ from $\underline{x}_k$ employs the model

$$F(\underline{x}_k + \underline{d}) \;\approx\; Q_k(\underline{x}_k + \underline{d}) \;=\; F(\underline{x}_k) + \underline{d}^T \underline{\nabla} F(\underline{x}_k) + \tfrac{1}{2}\, \underline{d}^T B_k\, \underline{d}, \qquad \underline{d} \in \mathcal{R}^n, \tag{5.2}$$

as in expressions (2.4) and (4.4), but now there is no need for the symmetric matrix B_k to be positive definite. We assume that termination occurs if $\|\underline{\nabla} F(\underline{x}_k)\|$ is sufficiently small. Otherwise, we require $\underline{d}_k$ to be an estimate of the vector $\underline{d}$ that minimizes $Q_k(\underline{x}_k + \underline{d})$, $\underline{d} \in \mathcal{R}^n$, subject to $\|\underline{d}\| \leq \Delta_k$. If $\underline{d}_k$ is an exact solution to this subproblem, then there exists $\lambda_k \geq 0$ such that the equation

$$(B_k + \lambda_k I)\, \underline{d}_k \;=\; -\underline{\nabla} F(\underline{x}_k) \tag{5.3}$$

holds, and $B_k + \lambda_k I$ is positive definite or semi-definite, where I is the identity matrix. Thus reliable procedures for calculating $\underline{d}_k$ with control of accuracy are given by Moré and Sorensen (1983), but often they are too expensive when n is large. Instead it is usual to apply the conjugate gradient minimization procedure of Section 2 to the quadratic model (5.2), starting at $\underline{d}=0$. It generates a piecewise linear path in $\mathcal{R}^n$, the difference between the end and the beginning of the ℓ-th line segment of the path being the change that is made to the vector of variables $\underline{d}$ on the ℓ-th iteration. The conjugate gradient iterations are terminated if the path reaches the boundary of the region $\{\underline{d} : \|\underline{d}\| \leq \Delta_k\}$, or if the reduction in $Q(\underline{x}_k+\underline{d})$ by an iteration is much less than the total reduction so far. Then $\underline{d}_k$ is chosen to be the final point of the path, except that some algorithms seek further reductions in $Q_k(\underline{x}_k+\underline{d})$ in the case $\|\underline{d}_k\|=\Delta_k$ (Conn, Gould and Toint, 2000).

After picking $\underline{d}_k$, the new function value $F(\underline{x}_k+\underline{d}_k)$ is calculated. The ratio

$$\rho_k \;=\; \{F(\underline{x}_k) - F(\underline{x}_k+\underline{d}_k)\} \,/\, \{Q_k(\underline{x}_k) - Q_k(\underline{x}_k+\underline{d}_k)\} \tag{5.4}$$

is important, because a value close to one suggests that the current model is good for predicting the behaviour of $F(\underline{x}_k+\underline{d})$, $\|\underline{d}\| \leq \Delta_k$. Therefore the value of Δ_{k+1} for the next iteration may be set to $\max[\Delta_k, 2\|\underline{d}_k\|]$, Δ_k or $\frac{1}{2}\|\underline{d}_k\|$ in the cases $\rho_k \geq 0.8$, $0.2 \leq \rho_k < 0.8$ or $\rho_k < 0.2$, respectively, for example. No other values of F are calculated on the k-th iteration of most trust region methods, $\underline{x}_{k+1}$ being either $\underline{x}_k$ or $\underline{x}_k+\underline{d}_k$. It seems obvious to prefer $\underline{x}_{k+1}=\underline{x}_k+\underline{d}_k$ whenever the strict reduction $F(\underline{x}_k+\underline{d}_k) < F(\underline{x}_k)$ is achieved, which is the condition $\rho_k > 0$. Many trust region algorithms, however, set $\underline{x}_{k+1}$ to $\underline{x}_k+\underline{d}_k$ only if ρ_k is sufficiently large. If $F(\underline{x}_k+\underline{d}_k) \geq F(\underline{x}_k)$ occurs in a trust region method, then the conditions $\underline{x}_{k+1}=\underline{x}_k$ and $\|\underline{d}_{k+1}\| \leq \Delta_{k+1} < \|\underline{d}_k\|$ are satisfied. Hence, if the vector $\underline{x}_{k+1}+\underline{d}_{k+1}$ of the $(k+1)$-th iteration is regarded as the result of a step from $\underline{x}_k$, then the length of the step is less than $\|\underline{d}_k\|$ automatically. Thus trust region methods include a main ingredient of line search methods. Attention is given later to the choice of the new matrix B_{k+1} at the end of the k-th iteration.

As in Section 4, a difficulty in constrained calculations is the need for a balance between reducing $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, and correcting violations of the constraints. We retain the compromise of the penalty function (4.7), and we estimate $\Phi(\underline{x}_k+\underline{d}, \mu)$, $\underline{d} \in \mathcal{R}^n$, by the model

$$\begin{aligned}\Xi_k(\underline{x}_k+\underline{d}, \mu) \;=\; & F(\underline{x}_k) + \underline{d}^T \underline{\nabla} F(\underline{x}_k) + \tfrac{1}{2}\,\underline{d}^T B_k\, \underline{d} + \mu^{-1}\{\, \textstyle\sum_{i\in\mathcal{E}} |c_i(\underline{x}_k) + \underline{d}^T \underline{\nabla} c_i(\underline{x}_k)| \\ & + \textstyle\sum_{i\in\mathcal{I}} \max\,[0, -c_i(\underline{x}_k) - \underline{d}^T \underline{\nabla} c_i(\underline{x}_k)]\,\}, \qquad \underline{d} \in \mathcal{R}^n,\end{aligned} \tag{5.5}$$

which reduces to expression (5.2) if there are no constraints. It is usual to terminate the sequence of iterations if the residuals of the first order KKT conditions are sufficiently small at $\underline{x}=\underline{x}_k$. Otherwise, $\underline{d}_k$ and μ are chosen in a way that satisfies $\|\underline{d}_k\| \leq \Delta_k$ and $\Xi_k(\underline{x}_k+\underline{d}_k, \mu) < \Xi_k(\underline{x}_k, \mu)$. Let $F(\cdot)$ and $Q_k(\cdot)$ be replaced by $\Phi(\cdot, \mu)$ and $\Xi_k(\cdot, \mu)$ throughout the remarks of the previous paragraph, the new version of the definition (5.4) being the ratio

$$\rho_k \;=\; \{\Phi(\underline{x}_k, \mu) - \Phi(\underline{x}_k+\underline{d}_k, \mu)\} \,/\, \{\Xi_k(\underline{x}_k, \mu) - \Xi_k(\underline{x}_k+\underline{d}_k, \mu)\}. \tag{5.6}$$

The modified remarks give suitable techniques for choosing Δ_{k+1} and $\underline{x}_{k+1}$ in calculations with constraints on the variables.

If the ∞-norm is used instead of the 2-norm in the bound $\|\underline{d}\| \leq \Delta_k$, then the minimization of the function (5.5) for fixed μ subject to the bound is a quadratic programming problem. Thus the $\underline{d}_k$ of the previous paragraph can be calculated (Fletcher, 1985), with occasional decreases in μ if necessary in order to give enough weight to the constraints. Another way of generating $\underline{d}_k$ begins by letting $\widehat{\underline{d}}_k$ be an estimate of the $\underline{d}$ that minimizes $\Gamma_k(\underline{d})$, $\underline{d} \in \mathcal{R}^n$, subject to $\|\underline{d}\| \leq \frac{1}{2}\Delta_k$, where $\Gamma_k(\underline{d})$ is the term inside the braces of expression (5.5). Then $\underline{d}_k$ has to satisfy $\|\underline{d}_k\| \leq \Delta_k$ and $\Gamma_k(\underline{d}_k) \leq \Gamma_k(\widehat{\underline{d}}_k)$, which leaves some freedom in $\underline{d}_k$. It is taken up by trying to make $Q_k(\underline{x}_k + \underline{d}_k)$ substantially smaller than $Q_k(\underline{x}_k + \widehat{\underline{d}}_k)$, where $Q_k(\underline{x}_k + \underline{d})$ is still the quadratic term (5.2). This technique has the property that $\underline{d}_k$ is independent of μ, which is adjusted separately in a way that controls the required reduction $\Xi_k(\underline{x}_k + \underline{d}_k, \mu) < \Xi_k(\underline{x}_k, \mu)$.

Several advantages are provided by the fact that, in trust region methods, the second derivative matrix B_k of the model does not have to be positive definite. In particular, if the sparsity structure of $\nabla^2 F$ is known in unconstrained optimization, then B_{k+1} may be required to have the same structure in addition to satisfying the equation $B_{k+1}\underline{\delta}_k = \underline{\gamma}_k$ of Section 2, which may not allow B_{k+1} to be positive definite, even if we retain $\underline{\delta}_k^T \gamma_k > 0$. Moreover, we recall from Section 4 that, in constrained calculations, it is suitable to replace the condition $B_{k+1}\underline{\delta}_k = \underline{\gamma}_k$ by $B_{k+1}\underline{\delta}_k = \widehat{\underline{\gamma}}_k$, where $\widehat{\underline{\gamma}}_k$ is the difference (4.6). There is now no need for an unwelcome device to maintain positive definiteness, as described after equation (4.6). In both of these settings the conditions on the elements of B_{k+1} are linear equality constraints. A highly successful and convenient way of taking up the freedom in B_{k+1} is to minimize $\|B_{k+1} - B_k\|_F$, where the subscript F denotes the Frobenius norm. In other words, we let the new model be as close as possible to the old model subject to the linear constraints, where closeness is measured by the sum of squares of the changes to the elements of the second derivative matrix of the model. Some very useful properties of this technique are given in the next section.

Trust region methods are also more robust than line search methods when the Newton iteration (1.3) is modified, in case the starting point $\underline{x}_1$ is not "sufficiently close" to a solution. We recall that a line search method applies formula (1.4), but a trust region method would choose between the alternatives $\underline{x}_{k+1} = \underline{x}_k + \underline{d}_k$ and $\underline{x}_{k+1} = \underline{x}_k$, where $\underline{d}_k$ is an estimate of the vector $\underline{d}$ that minimizes $\|\underline{f}(\underline{x}_k) + J(\underline{x}_k)\underline{d}\|$ subject to $\|\underline{d}\| \leq \Delta_k$. The usual ways of selecting $\underline{x}_{k+1}$ and Δ_{k+1} for the next iteration are similar to those that have been described already.

6. Further remarks

In my experience, the question that has been most useful to the development of successful algorithms for unconstrained optimization is "Does the method work well when the objective function is quadratic?". The answer is very welcome and encouraging for the updating of second derivative matrices of quadratic models by the symmetric Broyden method, which is the technique of taking up freedom in the new model by minimizing $\|B_{k+1}-B_k\|_F$, mentioned in the paragraph before last. We are going to consider this method in unconstrained calculations when the current quadratic model has the form

$$F(\underline{x}_k+\underline{d}) \;\approx\; Q_k(\underline{x}_k+\underline{d}) \;=\; F(\underline{x}_k)+\underline{d}^T\underline{g}_k+\tfrac{1}{2}\,\underline{d}^TB_k\,\underline{d}, \qquad \underline{d}\in\mathcal{R}^n, \tag{6.1}$$

where $F(\underline{x}_k)$ and B_k are retained from expression (5.2), but $\underline{g}_k$ is allowed to be an estimate of $\underline{\nabla}F(\underline{x}_k)$ that is given to the k-th iteration, which is useful if first derivatives of F are not available.

Some constraints on the parameters of the new model

$$Q_{k+1}(\underline{x}_{k+1}+\underline{d}) \;=\; F(\underline{x}_{k+1})+\underline{d}^T\underline{g}_{k+1}+\tfrac{1}{2}\,\underline{d}^TB_{k+1}\,\underline{d}, \qquad \underline{d}\in\mathcal{R}^n, \tag{6.2}$$

have been stated already for algorithms that employ first derivatives. In addition to $\underline{g}_j=\underline{\nabla}F(\underline{x}_j)$, $j=1,2,3,\ldots$, they include the equation

$$B_{k+1}\,\underline{\delta}_k \;=\; \underline{\gamma}_k \;=\; \underline{\nabla}F(\underline{x}_k+\underline{\delta}_k)-\underline{\nabla}F(\underline{x}_k), \tag{6.3}$$

where $\underline{\delta}_k$ is $\underline{x}_{k+1}-\underline{x}_k$ or $\underline{d}_k$ in a line search or trust region method, respectively. In algorithms without derivatives, however, the new model Q_{k+1} may be derived from the current model Q_k and from interpolation conditions of the form

$$Q_{k+1}(\underline{z}_j) \;=\; F(\underline{z}_j), \qquad j=1,2,\ldots,m, \tag{6.4}$$

where the points $\underline{z}_j$, $j=1,2,\ldots,m$, are chosen automatically, one of them being $\underline{x}_{k+1}$. I prefer to keep m fixed at about $2n+1$ and to change only one of the interpolation points on each iteration, which can provide suitable data for the selection of both $\underline{g}_{k+1}$ and B_{k+1}. The matrix B_{k+1} is required to be symmetric in all of these algorithms, and sometimes B_{k+1} is given the sparsity structure of ∇^2F.

Let $F(\underline{x})$, $\underline{x}\in\mathcal{R}^n$, be a quadratic function. Then all the constraints on the parameters of Q_{k+1} in the previous paragraph are satisfied if we pick $Q_{k+1}\equiv F$. It follows from the linearity of the constraints that they allow any multiple of the difference $F-Q_{k+1}$ to be added to Q_{k+1}. Therefore, if B_{k+1} is calculated by minimizing $\|B_{k+1}-B_k\|_F$ subject to the constraints, which is the symmetric Broyden method, then the least value of $\phi(\theta)=\|B_{k+1}-B_k+\theta\,(\nabla^2F-B_{k+1})\|_F^2$, $\theta\in\mathcal{R}$, occurs at $\theta=0$. We consider this remark algebraically by introducing the notation $\langle V,W\rangle$ for the sum $\sum_{i=1}^n\sum_{j=1}^n V_{ij}W_{ij}$, where V and W are any $n\times n$ symmetric matrices. The definition of the Frobenius norm gives the expression

$$\phi(\theta)=\langle\,(B_{k+1}-B_k)+\theta\,(\nabla^2F-B_{k+1}),\,(B_{k+1}-B_k)+\theta\,(\nabla^2F-B_{k+1})\,\rangle, \tag{6.5}$$

$\theta\in\mathcal{R}$, which is least at $\theta=0$ if and only if the scalar product $\langle B_{k+1}-B_k,\,\nabla^2F-B_{k+1}\rangle$ is zero. This remark implies the identity

$$\|\nabla^2F-B_{k+1}\|_F^2 \;=\; \|\nabla^2F-B_k\|_F^2-\|B_{k+1}-B_k\|_F^2, \tag{6.6}$$

which is a well-known property of least squares projection methods. Thus, if F is quadratic, the symmetric Broyden method causes the Frobenius norms of the error matrices $\nabla^2 F - B_k$, $k=1,2,3,\ldots$, to decrease monotonically as the iterations proceed.

Equation (6.6) is highly relevant to the important breakthrough in convergence theory by Broyden, Dennis and Moré (1973). They find that, if $\underline{\nabla} F$ is available in the unconstrained minimization of $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, then usually the sequence $\underline{x}_k$, $k=1,2,3\ldots$, converges at a superlinear rate if the matrices B_k have the property

$$\lim_{k\to\infty} \|\underline{\nabla} F(\underline{x}_k + \underline{\delta}_k) - \{\underline{\nabla} F(\underline{x}_k) + B_k\, \underline{\delta}_k\}\| \,/\, \|\underline{\delta}_k\| \;=\; 0, \qquad (6.7)$$

for the choices of $\underline{\delta}_k$ considered already. The term $\underline{\nabla} F(\underline{x}_k) + B_k \underline{\delta}_k$ is the estimate of $\underline{\nabla} F(\underline{x}_k + \underline{\delta}_k)$ given by the quadratic model (6.1) in the case $\underline{g}_k = \underline{\nabla} F(\underline{x}_k)$. Many researchers had believed previously, however, that fast convergence in practice would require B_k to be sufficiently close to $\nabla^2 F(\underline{x}_k)$. Equation (6.6) shows that $\|B_{k+1} - B_k\|_F$ tends to zero as k increases. Therefore $\|B_{k+1}\underline{\delta}_k - B_k \underline{\delta}_k\| / \|\underline{\delta}_k\|$ tends to zero too, and we let B_{k+1} be constrained by condition (6.3). Thus the condition (6.7) for superlinear convergence is satisfied by the symmetric Broyden method even if $\|\nabla^2 F - B_k\|_F$ does not become small.

Some successes of the symmetric Broyden method in minimization without derivatives are stunning. In the NEWUOA software of Powell (2006), the constraints on the parameters of the new model (6.2) are the interpolation conditions (6.4) and the symmetry condition $B_{k+1}^T = B_{k+1}$. The volume of the convex hull of the points $\underline{z}_j$, $j=1,2,\ldots,m$, is forced to be nonzero, in order that both $\underline{g}_{k+1}$ and B_{k+1} are defined uniquely when they provide the least value of $\|B_{k+1} - B_k\|_F$ subject to the interpolation and symmetry constraints. The test function that was used most in the development of NEWUOA has the form

$$F(\underline{x}) \;=\; \sum_{i=1}^{2n} \Big\{ b_i - \sum_{j=1}^{n} \Big(S_{ij} \sin(\theta_j x_j) + C_{ij} \cos(\theta_j x_j) \Big) \Big\}^2, \qquad \underline{x} \in \mathcal{R}^n, \qquad (6.8)$$

which is equation (8.5) of Powell (2006). Details are given there, including the choices of the parameters b_i, S_{ij}, θ_j and C_{ij} and of a starting point $\underline{x}_1$, several choices being made randomly for each n. In each experiment, the objective function (6.8) is minimized to high accuracy and the total number of calculations of $F(\underline{x})$ is noted. The average values of these counts with $m=2n+1$ are 931, 1809, 3159 and 6013 for $n=20$, 40, 80 and 160, respectively. We see that these figures are roughly proportional to n, which is not very surprising if one attributes the good rate of convergence to the property $\|B_{k+1} - B_k\|_F \to 0$. On the other hand, an algorithm that constructed a careful quadratic model would require more than $n^2/2$ calculations of $F(\underline{x})$. These observations are analogous to the remark that, if $\underline{\nabla} F$ is available, if $F(\underline{x})$, $\underline{x} \in \mathcal{R}^n$, is minimized by one of the methods of Section 2, and if n is large, then it is not unusual for the required accuracy to be achieved in far fewer than n iterations.

The material of this paper leans strongly towards my own contributions to nonlinear optimization. Therefore the presentation should be regarded as a personal view of an active researcher instead of an attempt at being comprehensive. Most of the algorithms that have been addressed do not require a review, because they, with several other methods, are now studied carefully in books, such as Fletcher (1987), Nocedal and Wright (1999) and Sun and Yuan (2006). The main exception is the brief consideration of minimization without derivatives in the previous paragraph, the NEWUOA software being only five years old. An excellent survey of another part of this field is given by Kolda, Lewis and Torczon (2003). It includes some work on optimization without derivatives when there are constraints on the variables. There is a strong need in that area for new algorithms that provide high accuracy efficiently.

References

C.G. Broyden, J.E. Dennis and J.J. Moré (1973), "On the local and superlinear convergence of quasi-Newton methods", *J. Inst. Math. Appl.*, Vol. 12, pp. 223–245.

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

A.R. Conn, N.I.M. Gould and Ph.L. Toint (2000), *Trust-Region Methods*, MPS/SIAM Series on Optimization, SIAM (Philadelphia).

W.C. Davidon (1959), "Variable metric method for minimization", Report ANL 5990 (rev.), Argonne National Laboratory, Illinois.

A.V. Fiacco and G.P. McCormick (1968), *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley & Sons (New York).

R. Fletcher (1985), "An ℓ_1 penalty method for nonlinear constraints", in *Numerical Optimization 1984*, eds. P.T. Boggs, R.H. Byrd and R.B. Schnabel, SIAM (Philadelphia), pp. 26–40.

R. Fletcher (1987), *Practical Methods of Optimization*, John Wiley & Sons (Chichester).

R. Fletcher and S. Leyffer (2002), "Nonlinear programming without a penalty function", *Math. Programming*, Vol. 91, pp. 239–269.

R. Fletcher and M.J.D. Powell (1963), "A rapidly convergent descent method for minimization", *Comput. J.*, Vol. 6, pp. 163–168.

R. Fletcher and C.M. Reeves (1964), "Function minimization by conjugate gradients", *Comput J.*, Vol. 7, pp. 149–154.

M.R. Hestenes (1969), “Multiplier and gradient methods”, *J. Optim. Theory Appl.*, Vol. 4, pp. 303–320.

T.G. Kolda, R.M. Lewis and V. Torczon (2003), “Optimization by direct search: new perspectives on some classical and modern methods”, *SIAM Review*, Vol. 45, pp. 385–482.

J.J. Moré and D.C. Sorensen (1983), “Computing a trust region step”, *SIAM J. Sci. Stat. Comput.*, Vol. 4, pp. 553–572.

J. Nocedal and S.J. Wright (1999), *Numerical Optimization*, Springer (New York).

E. Polak and G. Ribière (1969), “Note sur la convergence de méthodes de directions conjuguées”, *Rev. Française Informat. Recherche Opérationnelle*, 3$^{\text{e}}$ Année, No. 16, pp. 35–43.

M.J.D. Powell (1969), “A method for nonlinear constraints in minimization problems”, in *Optimization*, ed. R. Fletcher, Academic Press (London), pp. 283–298.

M.J.D. Powell (1970), “A hybrid method for nonlinear equations”, in *Numerical Methods for Nonlinear Algebraic Equations*, ed. P. Rabinowitz, Gordon and Breach (London), pp. 87–114.

M.J.D. Powell (1978), “A fast algorithm for nonlinearly constrained optimization calculations”, in *Numerical Analysis, Dundee 1977, Lecture Notes in Mathematics 630*, ed. G.A. Watson, Springer-Verlag (Berlin), pp. 144–157.

M.J.D. Powell (2006), “The NEWUOA software for unconstrained optimization without derivatives”, in *Large-Scale Nonlinear Optimization*, eds. G. Di Pillo and M. Roma, Springer (New York), pp. 255–297.

R.T. Rockafellar (1973), “A dual approach to solving nonlinear programming problems by unconstrained optimization”, *Math. Programming*, Vol. 5, pp. 354–373.

W. Sun and Y. Yuan (2006), *Optimization Theory and Methods: Nonlinear Programming*, Springer (New York).