Mathematical Programming 35 (1986) 265–278
North-Holland

A RECURSIVE QUADRATIC PROGRAMMING ALGORITHM THAT USES DIFFERENTIABLE EXACT PENALTY FUNCTIONS

M.J.D. POWELL and Y. YUAN

Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Silver Street, Cambridge CB3 9EW, England

Received 6 December 1984
Revised manuscript received 1 December 1985

In this paper, a recursive quadratic programming algorithm for solving equality constrained optimization problems is proposed and studied. The line search functions used are approximations to Fletcher's differentiable exact penalty function. Global convergence and local superlinear convergence results are proved, and some numerical results are given.

Key words: Constrained optimization, exact penalty functions, global convergence, line search functions, recursive quadratic programming.

1. Introduction

The problem we consider is the equality constrained nonlinear optimization calculation:

$$\text{minimize} \quad f(x) \tag{1.1}$$

subject to

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

where $f(x) \in \mathbb{R}^{n\times 1}$ and $c(x) \in \mathbb{R}^{n\times m}$ are twice continuously differentiable.

The recursive quadratic programming method for solving (1.1)–(1.2) is iterative. At the beginning of an iteration an estimate of the solution is available, and a search direction is calculated by solving a quadratic programming subproblem which is an approximation to the original problem. Then a new estimate of the solution is obtained by moving from the old one along the search direction, where the step-length is calculated by some technique. Such methods have been studied by many authors, including Wilson [20], Bartholomew-Biggs [1, 2], Han [12, 13], Powell, [14, 15, 16, 17] and Schittkowski [18, 19].

It is known [15] that, if the step-length of one is eventually acceptable on every iteration, then the recursive quadratic programming method converges superlinearly under certain conditions. Several recent papers consider how to choose line search functions so that the unit step-length is acceptable near the solution (see Chamberlain et al. [6], Gill et al. [11] and Schittkowski [18, 19]). It is also suitable [3, 4] to use

Fletcher's [8, 9, 10] differentiable exact penalty function as a line search function. We propose and study an algorithm of this type, that has the important feature that approximations to the derivatives of Lagrange multipliers are employed that avoid the need to calculate any second derivatives.

Our algorithm is described in Section 2. Its global convergence properties are studied in Section 3, and local convergence is considered in Section 4. Some numerical results that were obtained by a particular implementation of the algorithm are presented in Section 5. Finally, there is a brief discussion of the given theory and results.

2. The algorithm

Fletcher's [8] exact penalty function has the form

$$\phi(x,\sigma)=f(x)-c(x)^{\mathrm{T}}\lambda(x)+\tfrac{1}{2}\sigma\|c(x)\|_2^2,\quad x\in\mathbb{R}^n, \tag{2.1}$$

where $\sigma\geqslant 0$ is a parameter and where, for every x,

$$\lambda(x)=A(x)^{+}g(x),\qquad g(x)=\nabla f(x),\qquad A(x)=\nabla c(x)^{\mathrm{T}}, \tag{2.2}$$

A^{+} being the Moore–Penrose generalized inverse of A and $\|\cdot\|$ being the 2-norm. When the columns of A are linearly independent, which is assumed in Sections 3 and 4, then A^{+} is the matrix $(A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}$. Unfortunately the gradient $\nabla_x\phi(x,\sigma)$ depends on second derivatives of $f(x)$ and $c_i(x)$ $(i=1,2,\ldots,m)$. Therefore on the iteration that calculates x_{k+1} from x_k, once a search direction d_k and a trial step-length $\beta_{k,i}$ are chosen, we approximate $\lambda(x_k+\alpha\beta_{k,i}d_k)$ by $\lambda(x_k)+\alpha[\lambda(x_k+\beta_{k,i}d_k)-\lambda(x_k)]$, which gives the line search function

$$\begin{aligned}\Phi_{k,i}(\alpha\beta_{k,i})=&f(x_k+\alpha\beta_{k,i}d_k)\\&-[\lambda(x_k)+\alpha(\lambda(x_k+\beta_{k,i}d_k)-\lambda(x_k))]^{\mathrm{T}}c(x_k+\alpha\beta_{k,i}d_k)\\&+\tfrac{1}{2}\sigma_{k,i}\|c(x_k+\alpha\beta_{k,i}d_k)\|_2^2,\quad 0\leqslant\alpha\leqslant 1.\end{aligned} \tag{2.3}$$

Here i is the number of step-lengths that have been tried already on the current iteration, so we let $\beta_{k,0}=1$. For $i\geqslant 1$, $\beta_{k,i}$ may be any number from the interval $[\beta_1\beta_{k,i-1},\beta_2\beta_{k,i-1}]$ where β_1 and β_2 are constants such that $0<\beta_1\leqslant\beta_2<1$. A procedure that determines the penalty parameter $\sigma_{k,i}$ will be given. Our algorithm will make use of $\Phi_{k,i}(0)$, $\Phi_{k,i}(\beta_{k,i})$ and the derivative

$$\begin{aligned}\Phi'_{k,i}(0)=&g(x_k)^{\mathrm{T}}d_k-\frac{1}{\beta_{k,i}}[\lambda(x_k+\beta_{k,i}d_k)-\lambda(x_k)]^{\mathrm{T}}c(x_k)\\&-\lambda(x_k)^{\mathrm{T}}A(x_k)^{\mathrm{T}}d_k-\sigma_{k,i}\|c(x_k)\|_2^2,\end{aligned} \tag{2.4}$$

where the last term depends on the fact that d_k will satisfy equation (2.6). We see that $\lambda(x_k+\beta_{k,i}d_k)$ has to be calculated for every $\beta_{k,i}$.

We use the notation $g_k = g(x_k)$, $A_k = A(x_k)$, $c_k = c(x_k)$, etc. As mentioned in Section 1, the algorithm is iterative. An initial guess of the solution $x_1 \in \mathbb{R}^n$ is required. Constants $\beta_1, \beta_2, \mu \in (0, \frac{1}{2})$ and $\sigma_{1,-1} > 0$ are provided and a suitable $n \times n$ symmetric matrix B_1 is also given.

At the beginning of the k-th iteration, x_k, $\sigma_{k,-1}$ and B_k are available. We let the search direction d_k be the solution of the quadratic programming subproblem:

$$\text{minimize} \quad g_k^{\mathrm{T}} d + \tfrac{1}{2} d^{\mathrm{T}} B_k d \tag{2.5}$$

subject to the constraints

$$c_k + A_k^{\mathrm{T}} d = 0, \tag{2.6}$$

which we assume has a unique solution.

For each integer i that occurs, the following procedure gives a value of $\sigma_{k,i}$ that satisfies the conditions

$$\Phi'_{k,i}(0) \leq -\tfrac{1}{2}[d_k^{\mathrm{T}} B_k d_k + \sigma_{k,i} \| c(x_k) \|_2^2] \leq -\tfrac{1}{4} \sigma_{k,i} \| c(x_k) \|_2^2. \tag{2.7}$$

We let $\sigma_{k,i} = \sigma_{k,i-1}$ whenever (2.7) holds for $\sigma_{k,i} = \sigma_{k,i-1}$. Otherwise we let

$$\begin{aligned} \sigma_{k,i} = \max\Bigg[2\sigma_{k,i-1}, -2\frac{d_k^{\mathrm{T}} B_k d_k}{\|c_k\|^2}, \frac{2}{\|c_k\|^2} \Bigg[\tfrac{1}{2} d_k^{\mathrm{T}} B_k d_k \\ + g_k^{\mathrm{T}} d_k - \frac{1}{\beta_{k,i}} [\lambda(x_k + \beta_{k,i} d_k) - \lambda_k]^{\mathrm{T}} c_k - \lambda_k^{\mathrm{T}} A_k^{\mathrm{T}} d_k \Bigg] \Bigg]. \end{aligned} \tag{2.8}$$

but the middle term on the right hand side is redundant if B_k is positive definite or positive semi-definite. It follows that either

$$\sigma_{k,i} = \sigma_{k,i-1}, \tag{2.9}$$

or

$$\sigma_{k,i} \geq 2\sigma_{k,i-1}. \tag{2.10}$$

We try the values $i = 0, 1, 2, \ldots,$ until $i = i_k$, which is defined to be the smallest non-negative integer i such that the condition

$$\Phi_{k,i}(\beta_{k,i}) \leq \Phi_{k,i}(0) + \mu \beta_{k,i} \Phi'_{k,i}(0) \tag{2.11}$$

is satisfied. Then we set

$$x_{k+1} = x_k + \beta_{k,i_k} d_k \tag{2.12}$$

and

$$\sigma_{k+1,-1} = \sigma_{k,i_k}. \tag{2.13}$$

A suitable symmetric matrix B_{k+1} is generated by some means, which completes the k-th iteration.

A summary of this algorithm is as follows:

Step 0. $x_1 \in \mathbb{R}^n$, $\beta_1 \in (0, 1)$, $\beta_2 \in [\beta_1, 1)$, $\mu \in (0, \frac{1}{2})$, $\sigma_{1,-1} > 0$, and B_1 are given. Set $k := 1$.
Step 1. Solve (2.5)-(2.6), giving d_k.
Stop if $d_k = 0$.
Otherwise set $i = 0$ and $\beta_{k,0} = 1$.
Step 2. Choose $\sigma_{k,i}$ in the way described.
If (2.11) is satisfied then go to Step 3.
Otherwise set $i := i+1$, choose $\beta_{k,i} \in [\beta_1\beta_{k,i-1}, \beta_2\beta_{k,i-1}]$ and repeat Step 2.
Step 3. Set $i_k := i$, $x_{k+1} := x_k + \beta_{k,i_k} d_k$, and $\sigma_{k+1,-1} := \sigma_{k,i_k}$.
Generate B_{k+1}.
Set $k := k+1$ and go to Step 1.

The following lemma shows that division by zero does not occur in formula (2.8).

Lemma 2.1. *If* $\|d_k\| > 0$, *and if* (2.7) *is not satisfied for* $\sigma_{k,i} = \sigma_{k,i-1}$, *then* $\|c_k\| \neq 0$.

Proof. Assume the lemma is not true. Then

$$c_k = 0 \tag{2.14}$$

and, in view of equation (2.6), the failure of condition (2.7) implies

$$g_k^{\mathrm{T}} d_k > -\tfrac{1}{2} d_k^{\mathrm{T}} B_k d_k \quad \text{or} \quad d_k^{\mathrm{T}} B_k d_k < 0\,. \tag{2.15}$$

If the first part of (2.15) holds, then replacing d_k by 0 would reduce the objective function (2.5) and would preserve the constraint (2.6), which is a contradiction. If the second part holds, then the multiplication of d_k by a large positive factor would also reduce (2.5) and preserve (2.6). Therefore the lemma is true. □

3. Global convergence

To prove global convergence, we make the following assumptions:

Condition 3.1. (i) $\{x_k\}$, $\{d_k\}$ and $\{B_k\}$ are bounded.
(ii) $A(x)$ has full column rank for all $x \in \mathbb{R}^n$.
(iii) Each matrix B_k is such that, if $h \in \mathbb{R}^n$ is any vector such that $A_k^{\mathrm{T}} h = 0$, then $h^{\mathrm{T}} B_k h \geq \delta \|h\|^2$, where δ is a positive constant.

These conditions imply that every quadratic programming subproblem has a solution, and they give the following useful bound on the initial derivative of the line search function.

Lemma 3.2. *Expression* (2.7) *and Conditions* 3.1 *imply the inequality*

$$\Phi'_{k,i}(0) \leq -\eta \|d_k\|^2, \tag{3.1}$$

where η is a positive constant.

Proof. Let $\varepsilon > 0$ satisfy the condition

$$2\varepsilon \|B_k\| + \varepsilon^2 \|B_k\| \leq \tfrac{1}{2}\delta \tag{3.2}$$

for all k. We express d_k as $(e_k + h_k)$, where e_k and h_k are in the column space of A_k and the null space of A_k^T respectively. Therefore, because e_k is the shortest vector d that can satisfy equation (2.6), we have the relations

$$\|e_k\| \leq M_0 \|c_k\|, \qquad h_k^T B_k h_k \geq \delta \|h_k\|^2 \tag{3.3}$$

where M_0 is a constant. Hence, when $\|e_k\| \geq \varepsilon \|h_k\|$, the bound

$$\|c_k\|^2 \geq M_0^{-2} \|e_k\|^2 \geq M_0^{-2} (\|e_k\| + \|h_k\|)^2 / (1 + \varepsilon^{-1})^2 = M_0^{-2} \|d_k\|^2 / (1 + \varepsilon^{-1})^2 \tag{3.4}$$

is satisfied. Alternatively, when $\|e_k\| < \varepsilon \|h_k\|$, inequalities (3.2) and (3.3) imply the condition

$$\begin{aligned} d_k^T B_k d_k &= (e_k + h_k)^T B_k (e_k + h_k) \geq \|h_k\|^2 (\delta - 2\varepsilon \|B_k\| - \varepsilon^2 \|B_k\|) \\ &\geq \tfrac{1}{2}\delta \|h_k\|^2 > \tfrac{1}{2}\delta (\|h_k\| + \|e_k\|)^2 / (1 + \varepsilon)^2 \\ &= \tfrac{1}{2}\delta \|d_k\|^2 / (1 + \varepsilon)^2. \end{aligned} \tag{3.5}$$

By using condition (3.4) in the last part of expression (2.7), or by using condition (3.5) in the middle part of this expression, it follows that inequality (3.1) holds for all k and i if η has the value

$$\eta = \tfrac{1}{4} \min[\sigma_{1,-1} M_0^{-2} / (1 + \varepsilon^{-1})^2, \delta / (1 + \varepsilon)^2], \tag{3.6}$$

which completes the proof of the lemma. □

Next we show that, if x_k is not a stationary point of (1.1)-(1.2), the algorithm will cycle within Step 2 only a finite number of times. A stationary point of (1.1)-(1.2) is a point x^* such that, for some $\lambda^* \in \mathbb{R}^m$,

$$c(x^*) = 0, \qquad g(x^*) - A(x^*)\lambda^* = 0. \tag{3.7}$$

Lemma 3.3. *Conditions* 3.1 *and* $\|d_k\| > 0$ *imply that* i_k *is finite.*

Proof. If the lemma is not true, we have that

$$\Phi_{k,i}(\beta_{k,i}) > \Phi_{k,i}(0) + \mu \beta_{k,i} \Phi'_{k,i}(0) \tag{3.8}$$

for all $i = 0, 1, \ldots$. Due to (2.9) and (2.10), it follows that either

$$\lim_{i \to \infty} \sigma_{k,i} = \infty \tag{3.9}$$

or

$$\sigma_{k,i} = \sigma_{k,i_0} = \bar{\sigma}_k, \quad i \geq i_0, \tag{3.10}$$

for some i_0. First we assume (3.9). By Lemma 2.1 we must have $\|c_k\| \neq 0$. Therefore, remembering that $\sigma_{k,i} = \sigma_{k,i-1}$ if conditions (2.7) hold, the limit (3.9) occurs only if the expression

$$g_k^{\mathrm{T}} d_k - \lambda_k^{\mathrm{T}} A_k^{\mathrm{T}} d_k + \tfrac{1}{2} d_k^{\mathrm{T}} B_k d_k - \frac{1}{\beta_{k,i}} [\lambda(x_k + \beta_{k,i} d_k) - \lambda_k]^{\mathrm{T}} c_k \tag{3.11}$$

becomes unbounded as $i \to \infty$. However, only the last term of this expression depends on i, and, due to the continuity of second derivatives and the full rank of A_k, it remains finite. Therefore the case (3.9) is not possible.

To complete our proof, we assume (3.10). Considering sufficiently large i and using (2.3), (2.4) and Taylor series, it can be shown that

$$\Phi_{k,i}(\beta_{k,i}) - \Phi_{k,i}(0) - \mu \beta_{k,i} \Phi'_{k,i}(0) = (1-\mu) \beta_{k,i} \Phi'_{k,i}(0) + \mathrm{o}(\beta_{k,i}). \tag{3.12}$$

Lemma 3.2 implies that the right hand side of this equation becomes negative as i increases, which contradicts inequality (3.8). This contradiction proves the lemma. □

Our next lemma shows that the numbers $\{\sigma_{k,i}\}$ are uniformly bounded.

Lemma 3.4. *Assuming Conditions* 3.1, *there exists* k_0 *such that*

$$\sigma_{k,i} = \sigma_{k_0,0} = \bar{\sigma} \tag{3.13}$$

for all $k \geq k_0$ *and* $0 \leq i \leq i_k$.

Proof. Because each $\sigma_{k,i}$ satisfies condition (2.9) or (2.10), it is sufficient to show that there exists a constant $\hat{\sigma}$ such that the inequalities (2.7) hold whenever $\sigma_{k,i} \geq \hat{\sigma}$. This result depends partly on the bound

$$g_k^{\mathrm{T}} d_k + d_k^{\mathrm{T}} B_k d_k - \lambda_k^{\mathrm{T}} A_k^{\mathrm{T}} d_k = \mathrm{O}(\|d_k\| \, \|c_k\|), \tag{3.14}$$

which is proved as follows.

Let $\bar{\lambda}_k$ be the Lagrange multipliers of the quadratic programming problem that determines d_k, so they satisfy the equation

$$g_k + B_k d_k = A_k \bar{\lambda}_k, \tag{3.15}$$

which gives the bound

$$|g_k^{\mathrm{T}} d_k + d_k^{\mathrm{T}} B_k d_k - \lambda_k^{\mathrm{T}} A_k^{\mathrm{T}} d_k| = |(\bar{\lambda}_k - \lambda_k)^{\mathrm{T}} A_k^{\mathrm{T}} d_k| \leq \|\bar{\lambda}_k - \lambda_k\| \, \|c_k\|. \tag{3.16}$$

Further, because A_k has full rank uniformly for bounded $\|x_k\|$, we deduce the condition

$$\begin{aligned} \|\bar{\lambda}_k - \lambda_k\| &= \mathrm{O}(\|A_k \bar{\lambda}_k - A_k \lambda_k\|) = \mathrm{O}(\|A_k \bar{\lambda}_k - g_k\| + \|A_k \lambda_k - g_k\|) \\ &= \mathrm{O}(\|B_k d_k\|), \end{aligned} \tag{3.17}$$

where the last line depends on the fact that the definition of λ_k gives the inequality

$$\|A_k\lambda_k - g_k\| \leq \|A_k\bar{\lambda}_k - g_k\|. \tag{3.18}$$

Therefore, remembering that the matrices $\{B_k\}$ are bounded, equation (3.14) is true.

Now the term in equation (2.4) that includes the change in λ is also bounded above by a multiple of $\|d_k\|\,\|c_k\|$. Therefore the lemma is true if we can choose $\hat{\sigma}$ satisfying the conditions

$$-d_k^{\mathrm{T}}B_kd_k + M_1\|d_k\|\,\|c_k\| - \hat{\sigma}\|c_k\|^2 \leq -\tfrac{1}{2}[d_k^{\mathrm{T}}B_kd_k + \hat{\sigma}\|c_k\|^2] \leq -\tfrac{1}{4}\hat{\sigma}\|c_k\|^2 \tag{3.19}$$

for all k, where M_1 is any positive constant. Both these conditions hold if we achieve the bound

$$-\tfrac{1}{2}d_k^{\mathrm{T}}B_kd_k + M_1\|d_k\|\,\|c_k\| \leq \tfrac{1}{4}\hat{\sigma}\|c_k\|^2. \tag{3.20}$$

As in the proof of Lemma 3.2, we express d_k as $(e_k + h_k)$, and we bound $\|d_k\|$ by $[\|e_k\| + \|h_k\|]$. It follows from inequality (3.3) that the left hand side of inequality (3.20) is bounded above by the expression

$$-\tfrac{1}{2}\delta\|h_k\|^2 + M_2\|h_k\|\,\|c_k\| + M_3\|c_k\|^2, \tag{3.21}$$

where M_2 and M_3 are constants. Therefore condition (3.20) is satisfied if $\hat{\sigma}$ has the value $(4M_3 + 2M_2^2/\delta)$, which completes the proof of the lemma. □

We use this lemma to show that the search directions become small.

Lemma 3.5. *Our conditions imply the limit*

$$\|d_k\| \to 0 \quad \textit{as } k \to \infty. \tag{3.22}$$

Proof. Lemma 3.4 and definitions (2.1) and (2.3) give the values

$$\Phi_{k,i}(0) = \phi(x_k, \bar{\sigma}), \quad \Phi_{k,i}(\beta_{k,i}) = \phi(x_k + \beta_{k,i}d_k, \bar{\sigma}), \qquad k \geq k_0. \tag{3.23}$$

Because inequality (2.11) holds when $i = i_k$, it follows from equation (3.23) and Lemma 3.2 that the sequence $\{\phi(x_k, \bar{\sigma}): k = k_0, k_0+1, \ldots\}$ decreases monotonically. Further, the sequence is convergent due to the finiteness of $\{x_k\}$. Therefore we prove the lemma by showing that, if $\|d_k\| > \varepsilon$, where ε is any positive constant, and if $k \geq k_0$, then $[\phi(x_k, \bar{\sigma}) - \phi(x_{k+1}, \bar{\sigma})]$ is bounded away from zero.

As in equation (3.12), we deduce the relation

$$\Phi_{k,i}(\beta_{k,i}) - \Phi_{k,i}(0) - \mu\beta_{k,i}\Phi'_{k,i}(0) = (1-\mu)\beta_{k,i}\Phi'_{k,i}(0) + \Delta_{k,i}, \tag{3.24}$$

where $\Delta_{k,i}$ has the value

$$\begin{aligned}\Delta_{k,i} &= \Phi_{k,i}(\beta_{k,i}) - \Phi_{k,i}(0) - \beta_{k,i}\Phi'_{k,i}(0) \\ &= \phi(x_k + \beta_{k,i}d_k, \bar{\sigma}) - \phi(x_k, \bar{\sigma}) - \beta_{k,i}\nabla\phi(x_k, \bar{\sigma})^{\mathrm{T}}d_k \\ &\quad + [\lambda(x_k + \beta_{k,i}d_k) - \lambda_k]^{\mathrm{T}}c_k - \beta_{k,i}d_k^{\mathrm{T}}\nabla\lambda(x_k)^{\mathrm{T}}c_k\end{aligned} \tag{3.25}$$

for $k \geq k_0$. We see that $|\Delta_{k,i}|$ is $o(\beta_{k,i})$. Therefore we have $\Delta_{k,i} \leq (1-\mu)\varepsilon^2\eta\beta_{k,i}$ if $\beta_{k,i} < \hat{\varepsilon}$ for some positive constant $\hat{\varepsilon}$, where η occurs in Lemma 3.2. Further, because x_k and d_k are uniformly bounded, we can choose $\hat{\varepsilon}$ to be independent of k. It follows that, for $\|d_k\| > \varepsilon$ and $k \geq k_0$, expression (3.24) is negative if $\beta_{k,i} < \hat{\varepsilon}$. Hence the step-length of the k-th iteration is at least $\beta_1\hat{\varepsilon}$, so we have the inequality

$$\phi(x_k, \bar{\sigma}) - \phi(x_{k+1}, \bar{\sigma}) = \Phi_{k,i_k}(0) - \Phi_{k,i_k}(\beta_{k,i_k}) \geq -\mu\beta_{k,i_k}\Phi'_{k,i_k}(0) \geq \mu\beta_1\hat{\varepsilon}\eta\varepsilon^2. \tag{3.26}$$

Therefore the lemma is true. □

It is now straightforward to prove global convergence.

Theorem 3.6. *If Conditions* 3.1 *are satisfied, each accumulation point of* $\{x_k\}$ *is a stationary point of* (1.1)-(1.2).

Proof. Lemma 3.5 and equations (2.6) and (3.15) give the limits

$$c_k \to 0, \qquad g_k - A_k\bar{\lambda}_k \to 0 \tag{3.27}$$

as $k \to \infty$. Therefore the theorem is true. □

4. A local convergence property

It is shown in this section that, if eventually step-lengths of one would give a superlinear rate of convergence, then the line search condition (2.11) allows $\beta_{k,i} = 1$. We assume Conditions 3.1 and

Condition 4.1. $x_k \to x^*$.

Due to the global convergence theorem in the previous section, we know that x^* is a stationary point of (1.1)-(1.2).

Our superlinear convergence property is as follows:

Lemma 4.2. *If Conditions* 3.1 *and Condition* 4.1 *hold, and if* $\{t\}$ *is an infinite subsequence of* $\{k\}$ *such that*

$$\|x_t + d_t - x^*\| = o(\|x_t - x^*\|), \tag{4.1}$$

then, for all large t,

$$x_{t+1} = x_t + d_t. \tag{4.2}$$

Proof. We assume without loss of generality that $t \geq k_0$. Writing k instead of t, it is sufficient to show the inequality

$$\Phi_{k,0}(1) - \Phi_{k,0}(0) - \mu\Phi'_{k,0}(0) < 0. \tag{4.3}$$

The definition (2.3) and Lemma 3.4 give the value

$$\Phi_{k,0}(1) = f(x_k + d_k) - \lambda(x_k + d_k)^{\mathrm{T}}c(x_k + d_k) + \tfrac{1}{2}\bar{\sigma}\|c(x_k + d_k)\|^2. \tag{4.4}$$

Further, from the continuity of second derivatives, the equation

$$f(x_k+d_k)=f(x_k)+\tfrac{1}{2}d_k^{\mathrm{T}}[g_k+g(x_k+d_k)]+\mathrm{o}(\|d_k\|^2)$$
$$=f(x_k)+\tfrac{1}{2}d_k^{\mathrm{T}}[g_k+g(x^*)]+\mathrm{o}(\|d_k\|^2) \quad (4.5)$$

is satisfied, where the last line depends on condition (4.1), and there is an analogous expression for $c(x_k+d_k)$. Thus, noting that the last term in expression (4.4) is $\mathrm{o}(\|d_k\|^2)$, we deduce the relation

$$\Phi_{k,0}(1)-\Phi_{k,0}(0)=\tfrac{1}{2}d_k^{\mathrm{T}}[g_k+g(x^*)]-\lambda(x_k+d_k)^{\mathrm{T}}[c_k+\tfrac{1}{2}A_k^{\mathrm{T}}d_k+\tfrac{1}{2}A(x^*)^{\mathrm{T}}d_k]$$
$$-[-\lambda_k^{\mathrm{T}}c_k+\tfrac{1}{2}\bar{\sigma}\|c_k\|^2]+\mathrm{o}(\|d_k\|^2). \quad (4.6)$$

We compare the right hand side with $\tfrac{1}{2}\Phi'_{k,0}(0)$. Remembering equation (2.6), expressions (2.4) and (4.6) give the identity

$$\Phi_{k,0}(1)-\Phi_{k,0}(0)=\tfrac{1}{2}\Phi'_{k,0}(0)+\tfrac{1}{2}d_k^{\mathrm{T}}[g(x^*)-A(x^*)\lambda(x_k+d_k)]+\mathrm{o}(\|d_k\|^2)$$
$$=\tfrac{1}{2}\Phi'_{k,0}(0)+\mathrm{o}(\|d_k\|^2), \quad (4.7)$$

the last line being derived from equations (3.7) and (4.1). Inequality (4.3) now follows for sufficiently large k from $\mu<\tfrac{1}{2}$ and Lemma 3.2, which completes the proof. □

This lemma gives the following theorem.

Theorem 4.3. *If Conditions* 3.1 *and* 4.1 *hold, and if the limit*

$$\lim_{k\to\infty}\frac{\|x_k+d_k-x^*\|}{\|x_k-x^*\|}=0 \quad (4.8)$$

is obtained, then the algorithm sets $x_{k+1}=x_k+d_k$ *for all sufficiently large* k.

Proof. The result follows immediately from Lemma 4.2. □

Conditions on $\{B_k\}$ that ensure the limit (4.8) are given by Boggs, Tolle and Wang [5] and by Powell [16].

5. Some numerical results

In order to test the given algorithm, some parameters, the details of the technique of adjusting the step-length, and a formula that defines B_{k+1} have to be chosen. We select $\beta_1=0.05$, $\beta_2=0.5$, $\mu=0.1$, $\sigma_{1,-1}=10$ and $B_1=I$ because these values seem suitable, but there was no empirical tuning of parameters.

If the initial step-length $\beta_{k,0}=1$ is unacceptable, then $\beta_{k,i+1}$ is chosen from the interval $[\beta_1\beta_{k,i}, \beta_2\beta_{k,i}]$ in the following way. Because condition (2.11) has been tested, the numbers $\Phi_{k,i}(0)$, $\Phi'_{k,i}(0)$ and $\Phi_{k,i}(\beta_{k,i})$ are available, and by interpolation they provide a convex quadratic approximation to the line search function $\{\Phi_{k,i}(\alpha\beta_{k,i});\ 0\leq\alpha\leq 1\}$. We set $\beta_{k,i+1}$ to the step-length that minimizes this quadratic approximation subject to the conditions that depend on β_1 and β_2. Thus we obtain the value

$$\beta_{k,i+1}=\max\{\beta_1, \min[\beta_2, \alpha_{k,i}]\}\beta_{k,i}, \tag{5.1}$$

where

$$\alpha_{k,i}=\frac{-\beta_{k,i}\Phi'_{k,i}(0)}{2[\Phi_{k,i}(\beta_{k,i})-\Phi_{k,i}(0)-\beta_{k,i}\Phi'_{k,i}(0)]}. \tag{5.2}$$

The formula for calculating B_{k+1} depends, as usual, on an estimate of the change in gradient of the Lagrangian function when the variables are altered from x_k to x_{k+1}. Here decisions have to be taken that deserve careful investigation, but we only chose an estimate that seemed suitable, namely the vector

$$\gamma_k=\left[\nabla F(x_{k+1})-\sum_{i=1}^{m}\lambda_i(x_{k+1})\nabla c_i(x_{k+1})\right]-\left[\nabla F(x_k)-\sum_{i=1}^{m}\lambda_i(x_k)\nabla c_i(x_k)\right], \tag{5.3}$$

where $\lambda(x)$ is defined in the first part of equation (2.2). Thus γ_k is the change in gradient of the exact penalty function (2.1) when $\sigma=0$, except that we have dropped the change in $\nabla\lambda(x)$, which depends on second derivatives, and which is multiplied by constraint functions that should tend to zero. Notwithstanding its disadvantages [17], we adopted the technique of Powell [14] for preserving positive definiteness. Therefore B_{k+1} is the matrix

$$B_{k+1}=B_k-\frac{B_k\delta_k\delta_k^{\mathrm{T}}B_k}{\delta_k^{\mathrm{T}}B_k\delta_k}+\frac{\eta_k\eta_k^{\mathrm{T}}}{\delta_k^{\mathrm{T}}\eta_k} \tag{5.4}$$

where

$$\delta_k=x_{k+1}-x_k, \qquad \eta_k=\theta_k\gamma_k+(1-\theta_k)B_k\delta_k \tag{5.5}$$

and

$$\theta=\begin{cases}1, & \delta_k^{\mathrm{T}}\gamma_k\geq 0.1\ \delta_k^{\mathrm{T}}B_k\delta_k,\\ 0.9\ \delta_k^{\mathrm{T}}B_k\delta_k/[\delta_k^{\mathrm{T}}B_k\delta_k-\delta_k^{\mathrm{T}}\gamma_k], & \text{otherwise.}\end{cases} \tag{5.6}$$

This algorithm was applied to a set of test problems, proposed by Powell [17], that can include highly nonlinear constraints. Each objective function has the form

$$F(x)=\hat{F}(z(x))=\sum_{i=1}^{m}u_iz_i+\tfrac{1}{2}\sum_{i=1}^{n}\left[\sum_{j=1}^{n}A_{ij}z_j\right]^2, \tag{5.7}$$

where z is defined by

$$z_i = \sum_{j=1}^{i} C_{ij}x_j + \tfrac{1}{2}\rho \sum_{j=1}^{n}\sum_{k=1}^{n} D_{ijk}x_jx_k, \quad i=1,2,\ldots,n. \tag{5.8}$$

The constraints are $\{z_i=0;\ i=1,2,\ldots,m\}$. As suggested by Powell [17], the constants $u_i\in[0.1, 1.1]$, $A_{ij}\in[-1,1]$, $C_{ij}\in[-1,1]$ but $C_{ii}\in[0.1,1.1]$, and $D_{ijk}\in[-1,1]$ but $D_{ijk}=D_{ikj}$, are generated randomly. The parameter ρ controls the nonlinearity of the constraints. The point $x=0$ is a solution, but other solutions and local minima can occur. In all cases we set $n=4$ and $x_1=(1,1,1,1)^{\mathrm{T}}$. For five different choices of random numbers we tried $m=1,2,3$ and 4, and $\rho=0.01$, 0.1 and 1, so we have 60 different test problems.

The calculations were done by an IBM 3081 computer in double precision arithmetic, using the condition

$$[\|d_k\|^2+\|c(x_k)\|^2]^{1/2}\leq 10^{-12} \tag{5.9}$$

for terminating the iterations. However, because it was not always possible to achieve this accuracy, a calculation was also terminated if ten unsuccessful function evaluations were made during a line search. Table 1 gives the total number of function and gradient evaluations (and the number of iterations in brackets) for each of the test problems. A '*' indicates an error return due to an unsuccessful line search, and a '†' indicates that the sequence $\{x_k;\ k=1,2,3,\ldots\}$ seems to be converging to a non-zero limit. In 38 of the 60 test problems $\sigma_{k,i}$ remained at its initial value of 10, and the final value of this parameter exceeded 1000 in just 3 cases, namely the ones mentioned below where the constraint gradients are nearly linearly dependent.

These figures are similar to Table 1 and Table 2 of Powell [17], but in that paper all constraints are inequalities. They show that, on most iterations, our modification of Fletcher's exact differentiable penalty function allows a step-length of one. The case 5, $m=1$, $\rho=1.0$ and Case 2, $m=2$, $\rho=1.0$ error returns are due to rounding

Table 1
Numbers of function evaluations and iterations

		Case 1	Case 2	Case 3	Case 4	Case 5
$m=1$	$\rho=0.01$	13 (12)	17 (16)	13 (12)	19 (17)	17 (16)
$m=1$	$\rho=0.1$	15 (14)	15 (14)	15 (14)	18 (16)	18 (17)
$m=1$	$\rho=1.0$	17 (16)	25 (21)	38 (34)†	28 (23)	36 (22)*†
$m=2$	$\rho=0.01$	13 (12)	12 (11)	13 (12)	14 (13)	12 (11)
$m=2$	$\rho=0.1$	14 (13)	11 (10)	15 (14)	15 (14)	12 (11)
$m=2$	$\rho=1.0$	21 (16)†	33 (18)*†	23 (20)†	20 (18)	31 (27)†
$m=3$	$\rho=0.01$	8 (7)	9 (8)	9 (8)	8 (7)	8 (7)
$m=3$	$\rho=0.1$	10 (9)	10 (9)	13 (12)	9 (8)	12 (11)
$m=3$	$\rho=1.0$	41 (25)	30 (21)	18 (15)†	22 (17)	18 (15)†
$m=4$	$\rho=0.01$	5 (4)	5 (4)	6 (5)	4 (3)	5 (4)
$m=4$	$\rho=0.1$	6 (5)	6 (5)	5 (4)	5 (4)	6 (5)
$m=4$	$\rho=1.0$	8 (6)†	92 (12)*†	11 (9)	7 (6)	70 (11)*†

errors, because the final values of the left hand side of expression (5.9) are 5×10^{-12} and 36×10^{-12} respectively. However, the two error returns that are shown in the last row of Table 1 are due to the fact that $\{x_k;\ k=1,2,3,\ldots\}$ is converging to a point at which the matrix A_k of constraint gradients is singular and constraint violations are large. Thus, due to equation (2.6), the final values of $\|d_k\|$ are 5×10^6 and 1×10^6 for Cases 2 and 5 respectively, so ten step-length reductions are insufficient to provide an acceptable vector of variables. The apparent inefficiency of the Case 1, $m=3$, $\rho=1.0$ case is due to the fact that near linear dependence of constraint gradients causes $\|d_k\|$ to be of magnitude 10 on several iterations.

6. Discussion

The main purpose of this paper is to show that Fletcher's differentiable exact penalty function can be used as a line search function without the calculation of any second derivatives. The analysis of Section 3 establishes that this line search function gives global convergence for a wide class of constrained optimization problems, and it is proved in Section 4 that the Maratos effect cannot occur. The numerical results of Section 5 are quite favourable, but many of the details of the computer program should be regarded as provisional.

It is particularly encouraging that rounding errors caused difficulties in only two of the calculations of Table 1, because condition (5.9) demands high accuracy, and our technique makes difference approximations to first derivatives of $\{\lambda(x);\ x\in\mathbb{R}^n\}$. Probably these difference approximations are adequate in practice because each one is multiplied by a constraint function, and $\{c(x_k);\ k=1,2,3,\ldots\}$ should tend to zero. The difficulties due to singularity of the Jacobian matrix when $m=n$ are probably not due to our technique, because it is well known that, if Newton's method with exact line searches is used to solve a square system of nonlinear equations, then convergence can occur to a point that is not a solution, unless the condition numbers of the Jacobian matrices remain bounded.

Partly because of this last remark, it is often suggested that one should relax condition (2.6) on the search direction. A suitable method is described by Powell [14], but a useful alternative that has received much attention recently is to calculate d_k by minimizing expression (2.5) plus a weighted sum of moduli of linear approximations to constraint violations. Thus one is minimizing an approximation to the nondifferentiable line search function that is usually employed to force convergence in recursive quadratic programming algorithms. Perhaps one should continue to choose the search direction in this way, even if the step-length of each iteration is determined by the differentiable line search function that we have studied. The possibility of using a differentiable line search function for the calculation of search directions is considered by Di Pillo and Grippo [7], but one loses the highly useful sharp restrictions on the search directions that are provided by the linear approximations to the constraints.

We have not yet given any attention to inequality constraints, or to the possibility of forcing convergence by means of trust regions instead of line searches. Some examples in Powell [17] show a need for differentiable line search functions in some problems with inequality constraints, and the use of trust regions may allow the removal of Condition 3.1(iii), which imposes a severe restriction on the matrices $\{B_k; k=1,2,3,\ldots\}$. Our results indicate that research on these questions would be valuable.

References

[1] M.C. Biggs, "On the convergence of some constrained minimization algorithms based on recursive quadratic programming", *J. Inst. Math. Appl.* 21 (1978) 67-82.

[2] M.C. Bartholomew-Biggs, "A recursive quadratic programming algorithm based on the augmented Lagrangian function", Technical Report No. 139, Numerical Optimisation Centre, The Hatfield Polytechnic, 1983.

[3] D.P. Bertsekas, "Augmented Lagrangian and differentiable exact penalty methods", in: M.J.D. Powell, ed., *Nonlinear optimization* 1981 (Academic Press, London, 1982) pp. 223-234.

[4] D.P. Bertsekas, *Constrained optimization and Lagrange multiplier methods* (Academic Press, New York, 1982).

[5] P.T. Boggs, J.W. Tolle and P. Wang, "On the local convergence of quasi-Newton methods for constrained optimization", *SIAM Journal of Control and Optimization* 20 (1982) 161-171.

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

[7] G. Di Pillo, L. Grippo and F. Lampariello, "A method for solving equality constrained optimization problems by constrained minimization", in: K. Iracki, K. Malanowski and S. Walukiewicz, eds., *Optimization techniques Part* 2, Lecture Notes in Control and Information Sciences 23 (Springer-Verlag, Berlin, 1980) pp. 96-105.

[8] R. Fletcher, "A class of methods for nonlinear programming with termination and convergence properties", in: J. Abadie, ed., *Integer and nonlinear programming* (North Holland, Amsterdam, 1970).

[9] R. Fletcher, "An exact penalty function for nonlinear programming with inequalities", *Mathematical Programming* 5 (1973) 129-150.

[10] R. Fletcher, "Penalty functions", in: A. Bachem, M. Grotschel and B. Korte, eds., *Mathematical programming, the state of the art* (Springer-Verlag, Berlin, 1983) pp. 87-113.

[11] P.E. Gill, W. Murray, M.A. Saunders and M.H. Wright, "User's guide for SOL/NPSOL: A FORTRAN package for nonlinear programming", Technical Report SOL 83-12. Department of Operations Research, Stanford University, Stanford.

[12] S.P. Han, "Superlinearly convergent variable metric algorithms for general nonlinear programming problems", *Mathematical Programming* 11 (1976) 263-282.

[13] S.P. Han, "A globally convergent method for nonlinear programming", *Journal of Optimization Theory and Applications* 22 (1977) 297-309.

[14] M.J.D. Powell, "A fast algorithm for nonlinearly constrained optimization calculations", in: G.A. Watson, ed., *Numerical analysis Dundee* 1977, Lecture Notes in Mathematics 630 (Springer-Verlag, Berlin, 1978) pp. 144-157.

[15] M.J.D. Powell, "The convergence of variable metric methods for nonlinear constrained optimization calculations", in: O.L. Mangasarian, R.R. Meyer and S.M. Robinson, eds., *Nonlinear programming* 3 (Academic Press, New York, 1978) pp. 27-63.

[16] M.J.D. Powell, "Variable metric methods for constrained optimization", in: A. Bachem, M. Grotschel and B. Korte, eds., *Mathematical programming, the state of the art* (Springer-Verlag, Berlin, 1983) pp. 288-311.

[17] M.J.D. Powell, "The performance of two subroutines for constrained optimization on some difficult test problems", in: P.T. Boggs, R.H. Boyd and R.B. Schnabel, eds., *Numerical optimization 1984* (SIAM, Philadelphia, 1985) pp. 160-177.

[18] K. Schittkowski, "The nonlinear programming method of Wilson, Han, and Powell with an augmented Lagrangian type line search function, Part I: convergence analysis", *Numerische Mathematik* 38 (1981) 83-114.

[19] K. Schittkowski, "On the convergence of a sequential quadratic programming method with an augmented Lagrangian line search function", *Mathematische Operationsforschung und Statistik, Ser. Optimization* 14 (1983) 197-216.

[20] R.B. Wilson, A simplicial algorithm for concave programming, Ph.D. Dissertation, Graduate School of Business Administration, Harvard University, Boston, 1963.