JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 67, No. 2, NOVEMBER 1990

Recursive Quadratic Programming Algorithm That Uses an Exact Augmented Lagrangian Function

S. LUCIDI[1]

Communicated by L. C. W. Dixon

Abstract. An algorithm for nonlinear programming problems with equality constraints is presented which is globally and superlinearly convergent. The algorithm employs a recursive quadratic programming scheme to obtain a search direction and uses a differentiable exact augmented Lagrangian as line search function to determine the steplength along this direction. It incorporates an automatic adjustment rule for the selection of the penalty parameter and avoids the need to evaluate second-order derivatives of the problem functions. Some numerical results are reported.

Key Words. Recursive quadratic programming, exact augmented Lagrangians, nonlinear programming, constrained optimization.

1. Introduction

Recursive quadratic programming (RQP) algorithms have gained considerable attention in the last few years and are considered to be effective tools for solving constrained nonlinear optimization problems. At each iteration, these methods compute a search direction by solving a quadratic programming subproblem which is an approximation to the original problem. Then, a stepsize along this direction is calculated by some technique in order to obtain global convergence towards Kuhn-Tucker points. Such methods have been investigated by many authors (see, for instance Refs. 1-12).

It is well known (see, e.g., Refs 8, 9, and 13) that, under standard assumptions, a recursive quadratic programming method converges superlinearly if the unit stepsize is eventually acceptable on every iteration. Several different techniques have been proposed to ensure that the unit steplength is acceptable near the solution (see, e.g., Refs. 4, 6, 11, 12, 14-20).

[1] Researcher, Istituto di Analisi dei Sistemi ed Informatica, CNR, Rome, Italy.

0022-3239/90/1100-0227$06.00/0

Recently, Powell (Ref. 21) has proposed an algorithm that, under very mild assumptions, is globally and superlinearly convergent. His algorithm uses, as line search function, Fletcher's differentiable exact penalty function (Ref. 22). However, each evaluation of this penalty function requires a matrix inversion and this may limit somewhat the applicability of Powell's algorithm.

Here, following an approach similar to Ref. 21, we propose an algorithm which tries to overcome this difficulty. Our algorithm is strongly based on Dixon's work (Ref. 20; see also Refs. 4 and 23). In fact, we use as line search function the differentiable exact augmented Lagrangian proposed in Ref. 24, and we employ a finite-difference approximation of part of its directional derivative in order to avoid the calculation of any second derivatives. This new algorithm makes use of an Armijo-type line search procedure and incorporates automatic adjustment rules both for the selection of the penalty parameter and for the choice of the step of the finite-difference approximations. By using these features of the algorithm, we can prove that it converges globally and superlinearly under the same assumption as Ref. 21.

The paper is organized as follows. In Section 2, we define the problem; in Section 3, we describe our algorithm; in Section 4, we show its global convergence toward K-T points of the problem; in Section 5, we consider its local convergence; in Section 6, we report some numerical results obtained by a preliminary implementation of our algorithm.

2. Problem Formulation and Preliminaries

The problem considered in this paper is the nonlinear programming problem with equality constraints:

$$\text{(P)} \quad \text{minimize} \quad f(x),$$
$$\text{s.t.} \quad h(x)=0,$$

where $f:\mathbb{R}^n \to \mathbb{R}$ and $h:\mathbb{R}^n \to \mathbb{R}^m$ are assumed to be twice continuously differentiable.

We denote by

$$L(x, \lambda) := f(x) + \lambda' h(x)$$

the Lagrangian function for Problem (P) and by $\nabla_x L(x, \lambda)$ the gradient of $L(x, \lambda)$ with respect to x. A Kuhn-Tucker pair for Problem (P) is a pair $(\hat{x}, \hat{\lambda})$ which satisfies the conditions $\nabla_x L(\hat{x}, \hat{\lambda}) = 0$ and $h(\hat{x}) = 0$.

In Ref. 24, it was shown that, under suitable hypotheses, a solution of Problem (P) and its corresponding Lagrange multiplier can be found by a single unconstrained minimization, with respect to x and λ of the following exact augmented Lagrangian:

$$S(x, \lambda; c) := f(x) + \lambda' h(x) + (c/2)\|h(x)\|^2 + (\eta/2)\|\nabla h(x)' \nabla_x L(x, \lambda)\|^2, \quad (1)$$

for a sufficiently large value of the penalty coefficient c and for any $\eta > 0$.

This result suggests that a possible approach for solving Problem (P) is to define an unconstrained minimization algorithm for $S(x, \lambda; c)$. This function has continuous first derivatives given by

$$\nabla_x S(x, \lambda; c) = \nabla_x L(x, \lambda) + c\nabla h(x) h(x) + \eta G(x, \lambda) \nabla h(x)' \nabla_x L(x, \lambda), \quad (2)$$

$$\nabla_\lambda S(x, \lambda; c) = h(x) + \eta \nabla h(x)' \nabla h(x) \nabla h(x)' \nabla_x L(x, \lambda), \quad (3)$$

where

$$G(x, \lambda) := \nabla_x [\nabla h(x)' \nabla_x L(x, \lambda)].$$

We can note that the computation of $\nabla_x S(x, \lambda; c)$ involves the term $G(x, \lambda)$, which requires the use of second derivatives of the original problem. Since sometimes these second derivatives are unavailable or are difficult to compute, it is interesting to define algorithms for minimizing $S(x, \lambda; c)$ which use only the first-order derivatives of the original problem.

In this contest, for the solution of Problem (P) we consider an algorithm defined by an iterative scheme of the form

$$x_{k+1} = x_k + \alpha_k \Delta x_k, \quad (4)$$

$$\lambda_{k+1} = \lambda_k + \alpha_k \Delta \lambda_k(\alpha_k). \quad (5)$$

The stepsize α_k is computed by an Armijo-type procedure to give a suitable reduction of the function $S(x, \lambda; c)$.

The vector Δx_k is generated as solution of the quadratic programming subproblem $QP(x_k, B_k)$, defined by

$$\min_{\Delta x} (1/2)\Delta x' B_k \Delta x + \nabla f(x_k)' \Delta x, \quad (6)$$

$$\text{s.t.} \quad \nabla h(x_k)' \Delta x + h(x_k) = 0, \quad (7)$$

where B_k is an $n \times n$ matrix obtained by some updating rule.

The vector $\Delta \lambda_k(\alpha)$ is computed in a way to satisfy the following condition:

$$\lim_{\alpha \to 0} \nabla h(x_k)' \nabla h(x_k) \Delta \lambda_k(\alpha) = -\nabla h(x_k)' \nabla_x L(x_k, \lambda_k) - G(x_k, \lambda_k)' \Delta x_k. \quad (8)$$

In the sequel, we adopt sometimes the notation

$$f_k := f(x_k), \qquad h_k := h(x_k), \qquad L_k := L(x_k, \lambda_k), \qquad G_k := G(x_k, \lambda_k),$$

and so on.

3. Algorithm

In this section, we describe our algorithm which is based on the unconstrained minimization of the exact augmented Lagrangian $S(x, \lambda; c)$. This algorithm avoids the calculation of second-order derivatives and is based on an automatic adjustment of the penalty parameter c.

In particular, the algorithm makes use of the scalar

$$DS_k(\Delta x, \Delta\lambda; c) := \nabla_x L_k' \Delta x - c\|h_k\|^2 - \eta\|\nabla h_k' \nabla_x L_k\|^2 + h_k' \Delta\lambda \tag{9}$$

and the vector

$$y_k(t) = [[\nabla h(x_k + t\Delta x_k)' \nabla_x L(x_k + t\Delta x_k, \lambda_k) - \nabla h(x_k)' \nabla_x L(x_k, \lambda_k)]/t], \tag{10}$$

which is a finite difference approximation of the vector $G(x_k, \lambda_k)' \Delta x_k$, where the stepsize t is controlled by the algorithm in order to ensure global and superlinear convergence.

Algorithm 3.1.

Data. $x_1, \lambda_1, \delta_1 \in (0, 1)$, $\delta_2 \in [\delta_1, 1)$, $\mu \in (0, 1/2)$, B_1, $c_{1,-1} > 0$, $\eta > 0$, $\sigma > 0$, $\gamma \in (0, 1)$, $\rho_1 \geq 1$, $\rho_2 \in (0, 1]$.

Step 0. Set $k = 1$.

Step 1. Compute Δx_k as solution of $QP(x_k, B_k)$ given by (6), (7). If $\Delta x_k = 0$, stop; otherwise, set $i = 0$, $\alpha_{k,0} = 1$, and choose $t_{k,0} \in (0, \sigma)$.

Step 2. Compute $\Delta\lambda_{k,i}$ by solving the system

$$\nabla h_k' \nabla h_k \Delta\lambda_{k,i} = -\nabla h_k' \nabla_x L_k - y_k(t_{k,i}), \tag{11}$$

where $y_k(t_{k,i})$ is defined by (10).

Step 3. Compute $c_{k,i} \geq c_{k,i-1}$ such that

$$\begin{aligned} DS_{k,i} &:= DS_k(\Delta x_k, \Delta\lambda_{k,i}; c_{k,i}) \\ &\leq -(\gamma/2)[\Delta x_k' B_k \Delta x_k + \eta\|\nabla h_k' \nabla_x L_k\|^2 + c_{k,i}\|h_k\|^2] \\ &\leq -(\gamma/4)[c_{k,i}\|h_k\|^2 + \eta\|\nabla h_k' \nabla_x L_k\|^2]. \end{aligned} \tag{12}$$

That is, if the inequalities (12) are satisfied by $c_{k,i-1}$, set $c_{k,i}=c_{k,i-1}$; otherwise, set

$$c_{k,i}=\max[\rho_1 c_{k,i-1}, -[2\Delta x_k' B_k \Delta x_k + \eta\|\nabla h_k' \nabla_x L_k\|^2]/\|h_k\|^2,$$

$$[\gamma\Delta x_k' B_k \Delta x_k - \hat{\gamma}\eta\|\nabla h_k'\nabla_x L_k\|^2 + 2(\nabla_x L_k'\Delta x_k + h_k'\Delta\lambda_{k,i})]/\hat{\gamma}\|h_k\|^2], \tag{13}$$

where $\hat{\gamma}=2-\gamma$.

Step 4. If the following inequality is satisfied,

$$S(x_k+\alpha_{k,i}\Delta x_k, \lambda_k+\alpha_{k,i}\Delta\lambda_{k,i}; c_{k,i}) \leq S(x_k, \lambda_k; c_{k,i}) + \mu\alpha_{k,i} DS_{k,i}, \tag{14}$$

go to Step 6; otherwise, go to Step 5.

Step 5. Set $i=i+1$, choose $\alpha_{k,i}\in[\delta_1\alpha_{k,i-1}, \delta_2\alpha_{k,i-1}]$. Then, if $\alpha_{k,i}<\rho_2 t_{k,i-1}$, set $t_{k,i}=\alpha_{k,i}$, and go to Step 2; if $\alpha_{k,i}\geq\rho_2 t_{k,i-1}$, set $t_{k,i}=t_{k,i-1}$, $c_{k,i}=c_{k,i-1}$, $y_{k,i}=y_{k,i-1}$, $\Delta\lambda_{k,i}=\Delta\lambda_{k,i-1}$, $DS_{k,i}=DS_{k,i-1}$, and go to Step 4.

Step 6. Set $i_k=i$, $x_{k+1}=x_k+\alpha_{k,i_k}\Delta x_k$, $\lambda_{k+1}=\lambda_k+\alpha_{k,i_k}\Delta\lambda_{k,i_k}$, $c_{k+1,-1}=c_{k,i_k}$; generate B_{k+1}, set $k=k+1$, and go to Step 1.

At Step 1, as mentioned in Section 2, Algorithm 3.1 produces the search direction in the x-space in the same way of the RQP approach.

Step 2 provides the vector $\Delta\lambda_{k,i}$ which depends on the finite-difference approximation $y_k(t_{k,i})$; in particular, by the theorem of the mean applied to (11), we have

$$\nabla h_k'\nabla h_k\Delta\lambda_{k,i} = -\nabla h_k'\nabla_x L_k - G_k'\Delta x_k$$
$$-\int_0^1 (G(x_k+\xi t_{k,i}\Delta x_k, \lambda_k) - G(x_k, \lambda_k))'\Delta x_k\, d\xi; \tag{15}$$

since $t_{k,i}\to 0$ for $\alpha_{k,i}\to 0$, we have that $\Delta\lambda_{k,i}$ satisfies (8).

At Step 3, Algorithm 3.1 adjusts the penalty parameter $c_{k,i}$ so that $DS_{k,i}$ is sufficiently negative. The scalar $DS_{k,i}$ is an approximation of the direction derivative $\nabla S(x_k, \lambda_k; c_{k,i})'\Delta z_{k,i}$ along the direction $\Delta z_{k,i}=(\Delta x_k', \Delta\lambda_{k,i}')'$. In fact, recalling (7), (11), and then the theorem of the mean, we obtain

$$DS_{k,i} = \Delta x_k'(\nabla_x L_k + c_{k,i}\nabla h_k h_k) + \eta y_{k,i}'\nabla h_k'\nabla_x L_k$$
$$+\Delta\lambda_{k,i}'(h_k + \eta\nabla h_k'\nabla h_k\nabla h_k'\nabla_x L_k)$$
$$=\nabla S(x_k, \lambda_k; c_{k,i})'\Delta z_{k,i}$$
$$+\eta\Delta x_k'(G(x_k+\hat{\xi}\Delta x_k, \lambda_k) - G_k)\nabla h_k'\nabla_x L_k, \tag{16}$$

where $\hat{\xi}\in(0, t_{k,i})$.

In Steps 4-6, a line search procedure along the direction $\Delta z_{k,i}$ is performed in order to obtain a steplength α_{k,i_k} which satisfies an Armijo-type condition employing the approximate directional derivative $DS(\Delta x_k, \Delta\lambda_{k,i}; c_{k,i})$. During this procedure, it may be necessary to adjust the finite-difference stepsize $t_{k,i}$ and, hence, to recompute the vector $\Delta\lambda_{k,i}$ and the scalar $DS(\Delta x_k, \Delta\lambda_{k,i}; c_{k,i})$ by using a better approximation to the vector $G(x_k, \lambda_k)'\Delta x_k$. However, in practice, a suitable choice of the parameter $t_{k,0}$ should avoid that such an event could occur.

First, by repeating the same steps of the proof of Lemma 2.1 of Ref. 21, we can show that division by zero does not occur in formula (13).

Lemma 3.1. If $\|\Delta x_k\| > 0$ and if (12) is not satisfied for $c_{k,i} = c_{k,i-1}$, then $\|h_k\| > 0$.

Proof. For brevity's sake, the proof is omitted, but it can be found in Ref. 25. □

4. Global Convergence

In this section, we consider the global convergence of Algorithm 3.1. In order to prove this property in all the results of this section we need the following assumptions.

Assumption H1. (i) The sequences $\{x_k\}$, $\{\Delta x_k\}$ and $\{B_k\}$ are bounded.

(ii) The matrix $\nabla h(x)'$ has full row rank for all $x \in \mathbb{R}^n$.

(iii) Each matrix B_k is such that, if $d \in \mathbb{R}^n$ is any vector such that $\nabla h_k' d = 0$, then $d'B_k d \geq \beta\|d\|^2$, where β is a positive constant.

These conditions ensure that every quadratic problem $QP(x_k, B_k)$ has a solution.

First of all, we recall the following well-known result.

Proposition 4.1. The pair $(\hat{x}, \hat{\lambda})$ is a K-T point for Problem (P) if and only if the pair $(\hat{d}, \hat{\rho}) = (0, \hat{\lambda})$ is a K-T pair for Problem QP$(\hat{x}, B_k)$. □

Therefore, the preceding proposition and Assumption H1 (ii) ensure that, if the algorithm satisfies after a finite number of iterations the test at Step 1, then the pair (x_ν, λ_ν), where x_ν is the last point produced and λ_ν is the unique vector which solves the system $\|\nabla h(x_\nu)'\nabla_x L(x_\nu, \lambda_\nu)\| = 0$, is a K-T point for Problem (P).

In order to prove that Algorithm 3.1 is globally convergent toward K-T points of Problem (P), we need some intermediate results. In particular, by repeating the proof of Lemma 3.1 of Ref. 21, we obtain a bound on the approximate directional derivative $DS_{k,i}$.

Proposition 4.2. The inequalities (12) imply that

$$DS_{k,i} \leq -\tilde{\gamma}(\|\Delta x_k\|^2 + \|\nabla h_k' \nabla_x L_k\|^2), \tag{17}$$

where $\tilde{\gamma}$ is a positive constant. □

We now prove that the Algorithm 3.1 will satisfy the test (14) at Step 4 after a finite number of iterations.

Proposition 4.3. For any index k, the index i_k defined at Step 6 is finite.

Proof. The proof is similar to that of Lemma 3.3 of Ref. 21 and therefore is omitted, but it can be found in Ref. 25. □

Now, we can show that, for sufficiently large values of the iteration index k, the penalty parameter $c_{k,i}$ is constant.

Proposition 4.4. There exists an index $\tilde{k}$ such that

$$c_{k,i} = c_{\tilde{k},0} = \tilde{c}, \qquad \text{for all } k \geq \tilde{k} \text{ and } 0 \leq i \leq i_k. \tag{18}$$

Proof. The proof is by contradiction. Suppose that the assertion is false. Then, there exist subsequences, which we relabel $\{x_k\}$, $\{\lambda_k\}$, $\{\Delta x_k\}$, $\{\Delta\lambda_{k,\hat{i}_k}\}$, $\{B_k\}$, $\{c_{k,\hat{i}_k}\}$, with $\hat{i}_k \leq i_k$, $c_{k,\hat{i}_k} \to \infty$, and such that either

$$DS_{k,\hat{i}_k} > -(\gamma/2)[\Delta x_k' B_k \Delta x_k + \eta \|\nabla h_k' \nabla_x L_k\|^2 + c_{k,\hat{i}_k} \|h_k\|^2] \tag{19}$$

or

$$-\Delta x_k' B_k \Delta x_k > (1/2)[c_{k,\hat{i}_k} \|h_k\|^2 + \eta \|\nabla h_k' \nabla_x L_k\|^2]. \tag{20}$$

The proof is divided in three parts.

(i) First, we prove that the sequence $\{\lambda_k\}$ is bounded. If (19) holds, by recalling the definition of $DS_{k,\hat{i}_k}$ [see (9) and Step 3 of Algorithm 3.1], we have

$$\begin{aligned} &\gamma \Delta x_k' B_k \Delta x_k + 2(\nabla_x L_k' \Delta x_k + h_k' \Delta\lambda_{k,\hat{i}_k}) - \hat{\gamma} c_{k,\hat{i}_k} \|h_k\|^2 \\ &> \hat{\eta} \|\nabla h_k' \nabla_x L_k\|^2, \end{aligned} \tag{21}$$

where

$$\hat{\gamma} = 2 - \gamma \quad \text{and} \quad \hat{\eta} = \hat{\gamma}\eta.$$

Then, (21) yields

$$\begin{aligned} &|\gamma \Delta x_k' B_k \Delta x_k + 2\nabla_x f_k' \Delta x_k| - \hat{\eta} \|\nabla h_k' \nabla_x f_k\|^2 + 2\|h_k\| \|\Delta\lambda_{k,\hat{i}_k}\| \\ &> \hat{\eta}\gamma_m((\nabla h_k' \nabla h_k)^2) \|\lambda_k\|^2 - 2\|h_k + \hat{\eta} \nabla h_k' \nabla h_k \nabla h_k' \nabla f_k\| \|\lambda_k\|, \end{aligned} \tag{22}$$

where $\gamma_m[(\nabla h_k' \nabla h_k)^2]$ is the smallest eigenvalue of $(\nabla h_k' \nabla h_k)^2$.

From (15) and Assumption H1, it follows that there exist two positive constant K_1 and K_2 such that, for all k, we have

$$\|\Delta\lambda_{k,\hat{\imath}_k}\| \leq K_1\|\lambda_k\| + K_2. \tag{23}$$

By substituting (23) in (22) and by using again Assumption H1, we obtain, for all k,

$$K_3\|\lambda_k\|^2 - K_4\|\lambda_k\| < K_5, \tag{24}$$

where K_3, K_4, K_5 are fixed positive constants.

If (20) holds, we can easily derive that

$$\begin{aligned}&|\Delta x'_k B_k \Delta x_k|\\ &> (\eta/2)\gamma_m[(\nabla h'_k \nabla h_k)^2]\|\lambda_k\|^2 - \|\eta \nabla h'_k \nabla h_k \nabla h'_k \nabla f_k\|\|\lambda_k\|.\end{aligned} \tag{25}$$

Therefore, also in this case there exist constants $K_6, K_7, K_8 > 0$ such that, for all k, we have

$$K_6\|\lambda_k\|^2 - K_7\|\lambda_k\| < K_8. \tag{26}$$

We can conclude, by using (24) and (26), that the sequence $\{\lambda_k\}$ belongs to a compact set.

(ii) Now, we prove that, for sufficiently large value of c, the following inequality holds for all k:

$$\begin{aligned}&-(1/2)\Delta x'_k B_k \Delta x_k + \|h_k\|\|\Delta\lambda_{k,\hat{\imath}_k}\| + \|h_k\|\|\hat{\lambda}_k - \lambda_k\|\\ &\leq (1/4)[c\|h_k\|^2 + \eta\|\nabla h'_k \nabla_x L_k\|^2],\end{aligned} \tag{27}$$

where $\hat{\lambda}_k$ is the Lagrange multiplier of subproblem $Q(x_k, B_k)$. By recalling the K-T conditions of problem $Q(x_k, B_k)$, the definition (15) of $\Delta\lambda_{k,\hat{\imath}_k}$, Assumption H1, and part (i) of the proof, we have that there exist $K_9 > 0$ and $K_{10} > 0$ such that

$$\|\hat{\lambda}_k - \lambda_k\| \leq K_9\|\Delta x_k\| + K_{10}\|\nabla h'_k \nabla_x L_k\|, \tag{28a}$$

$$\|\Delta\lambda_{k,\hat{\imath}_k}\| \leq K_9\|\Delta x_k\| + K_{10}\|\nabla h'_k \nabla_x L_k\|. \tag{28b}$$

By using (28) we have

$$\begin{aligned}&-\Delta x'_k B_k \Delta x_k/2 + \|h_k\|\\ &\times[\|\hat{\lambda}_k - \lambda_k\| + \|\Delta\lambda_{k,\hat{\imath}_k}\|] - (1/4)[c\|h_k\|^2 + \eta\|\nabla h'_k \nabla_x L_k\|^2]\\ &\leq -(1/2)\Delta x'_k B_k \Delta x_k + 2\|h_k\|[K_9\|\Delta x_k\| + K_{10}\|\nabla h'_k \nabla_x L_k\|]\\ &-(1/4)[c\|h_k\|^2 + \eta\|\nabla h'_k \nabla_x L_k\|^2].\end{aligned} \tag{29}$$

As in Ref. 21, we can express Δx_k as $e_k + v_k$, where e_k and v_k are in the column space of ∇h_k and the null space of $\nabla h'_k$, respectively, and satisfy the relations

$$\|e_k\| \leq M_0\|h_k\|, \qquad v'_k B_k v_k \geq \beta\|v_k\|^2, \tag{30}$$

where M_0 is a positive constant. Hence, from (29), it follows that there exist positive constants M_1, M_2, M_3 so that we obtain

$$\begin{aligned}
&-\Delta x_k' B_k \Delta x_k/2 + \|h_k\| \\
&\times[\|\hat{\lambda}_k - \lambda_k\| + \|\Delta\lambda_{k,\hat{\iota}_k}\|] - (1/4)[c\|h_k\|^2 + \eta\|\nabla h_k' \nabla_x L_k\|^2] \\
&\leq -(1/2)\beta\|v_k\|^2 + M_1\|v_k\|\,\|h_k\| + M_2\|h_k\|\,\|\nabla h_k' \nabla_x L_k\| + M_3\|h_k\|^2 \\
&-(1/4)[c\|h_k\|^2 + \eta\|\nabla h_k' \nabla_x L_k\|^2] \\
&= -(\beta/2)(\|v_k\| - M_1/\beta\|h_k\|)^2 - (\eta/4) \\
&\times(\|\nabla h_k' \nabla_x L_k\| - (2M_2/\eta)\|h_k\|)^2 \\
&-(c/4 - M_1^2/2\beta - M_2^2/\eta - M_3^2)\|h_k\|^2.
\end{aligned}$$

Therefore, for

$$c \geq 2M_1^2/\beta + 4M_2^2/\eta + 4M_3^2, \tag{31}$$

the inequality (27) is satisfied.

(iii) For sufficiently large values of k, the penalty parameter $c_{k,\hat{\iota}_k}$ satisfies the relation (31). Therefore, by parts (i) and (ii), we have that the inequality (27) holds for such values of k.

From (27) it follows immediately that

$$-(1/2)\Delta x_k' B_k \Delta x_k \leq (1/4)(c_{k,\hat{\iota}_k}\|h_k\|^2 + \eta\|\nabla h_k' \nabla_x L_k\|^2). \tag{32}$$

Then, recalling that Δx_k satisfies the K-T conditions for problem QP(x_k, B_k), we have

$$\begin{aligned}
DS_{k,\hat{\iota}_k} &= -\Delta x_k' B_k \Delta x_k - c_{k,\hat{\iota}_k}\|h_k\|^2 - \eta\|\nabla h_k' \nabla_x L_k\|^2 \\
&\quad + h_k' \Delta\lambda_{k,\hat{\iota}_k} + h_k'(\hat{\lambda}_k - \lambda_k) \\
&\leq -\Delta x_k' B_k \Delta x_k - c_{k,\hat{\iota}_k}\|h_k\|^2 - \eta\|\nabla h_k' \nabla_x L_k\|^2 \\
&\quad + \|h_k'\|[\|\hat{\lambda}_k - \lambda_k\| + \|\Delta\lambda_{k,\hat{\iota}_k}\|].
\end{aligned}$$

Hence, by (27), we have

$$\begin{aligned}
DS_{k,\hat{\iota}_k} &\leq -(1/2)\Delta x_k' B_k \Delta x_k \\
&\quad -(3/4)[\eta\|\nabla h_k' \nabla_x L_k\|^2 + c_{k,\hat{\iota}_k}\|h_k\|^2].
\end{aligned} \tag{33}$$

Now, we can observe that (32) and (33) contradict (19) and (20). □

Before proving our main result we need the following proposition.

Proposition 4.5. If Algorithm 3.1 produces an infinite sequence $\{x_k, \lambda_k\}$, then it follows that

$$\lim_{k\to\infty} \|\Delta x_k\| = 0, \tag{34}$$

$$\lim_{k\to\infty} \|\nabla h_k' \nabla_x L_k\| = 0. \tag{35}$$

Proof. By Proposition 4.4, we can assume without loss of generality that $c_{k,i} = \tilde{c}$. The proof of the proposition is divided in two parts. First, we show that the sequences $\{\lambda_k\}$ and $\{\Delta\lambda_{k,i_k-1}\}$ produced by Algorithm 3.1 are bounded; then, we prove (34) and (35).

(i) By (12), (14), and Proposition 4.4, we have that there exist an index k_0 such that

$$S(x_k, \lambda_k; \tilde{c}) < S(x_{k_0}, \lambda_{k_0}; \tilde{c}), \qquad \text{for all } k \geq k_0.$$

Then recalling (1), we have

$$(\eta/2)\gamma_m((\nabla h_k' \nabla h_k)^2)\|\lambda_k\|^2 - \|h_k + \eta \nabla h_k' \nabla h_k \nabla h_k' \nabla f_k\| \, \|\lambda_k\| < S(x_{k_0}, \lambda_{k_0}; \tilde{c}) + |f(x_k)|, \tag{36}$$

where again $\gamma_m[(\nabla h_k' \nabla h_k)^2]$ is the smallest eigenvalue of $(\nabla h_k' \nabla h_k)^2$. Now, Assumption H1 and (26) ensure that the sequence $\{\lambda_k\}$ belongs to a compact set. Then, by using (15), Assumption H1, and the boundedness of $\{\lambda_k\}$, we obtain that also the sequence $\{\Delta\lambda_{k,i_k-1}\}$ is bounded.

(ii) By using again (12) and (14), we have

$$S(x_{k+1}, \lambda_{k+1}; \tilde{c}) < S(x_k, \lambda_k; \tilde{c});$$

and, from the boundedness of $\{x_k\}$ and $\{\lambda_k\}$, it follows that the sequence $\{S(x_k, \lambda_k, \tilde{c})\}$ is bounded from below. Therefore, the sequence $\{S(x_k, \lambda_k, \tilde{c})\}$ converges; hence, (14) yields

$$\lim_{k \to \infty} \alpha_{k,i_k} DS_{k,i_k} = 0. \tag{37}$$

In order to prove (34) and (35), we proceed now by contradiction. Let us assume that there exists a subsequence (which we relabel with k) such that, for all k,

$$\|\Delta x_k\| + \|\nabla h_k' \nabla_x L_k\| > \delta. \tag{38}$$

Then, from Proposition 4.2, (37), (38), and the instructions at Step 5, it follows that

$$\lim_{k \to \infty} \alpha_{k,i_k-1} = 0, \tag{39a}$$

$$\lim_{k \to \infty} t_{k,i_k-1} = 0. \tag{39b}$$

By Assumption H1 and part (i) of the proof, we can assume without loss of generality that there exist $\hat{x}, \Delta\hat{x} \in \mathbb{R}^n$ and $\hat{\lambda}, \Delta\hat{\lambda} \in \mathbb{R}^m$ such that $x_k \to \hat{x}$, $\Delta x_k \to \Delta\hat{x}$, $\lambda_k \to \hat{\lambda}$, and $\Delta\lambda_{k,i_k-1} \to \Delta\hat{\lambda}$. Then (39), (16), Proposition 4.2, and (38) give

$$\lim_{k \to \infty} DS_{k,i_{k-1}} = \nabla S(\hat{x}, \hat{\lambda}; \tilde{c})' \Delta\hat{z} < 0, \tag{40}$$

where

$$\Delta\hat{z} := (\Delta\hat{x}', \Delta\hat{\lambda}')'.$$

Now, by the instructions at Step 5 of Algorithm 3.1, we have

$$S(x_k + \alpha_{k,i_k-1}\Delta x_k, \lambda_k + \alpha_{k,i_k-1}\Delta\lambda_{k,i_k-1}; \tilde{c})$$
$$> S(x_k, \lambda_k; \tilde{c}) + \mu\alpha_{k,i_k-1} DS_{k,i_{k-1}}. \tag{41}$$

By using the theorem of the mean and (16), we obtain from (41)

$$\nabla S(x_k + \zeta_k \Delta x_k, \lambda_k + \zeta_{k,i_{k-1}}\Delta\lambda_{k,i_k-1}; \tilde{c})'\Delta z_{k,i_{k-1}}$$
$$> \mu[\nabla S(x_k, \lambda_k; \tilde{c})'\Delta z_{k,i_{k-1}}$$
$$+ \eta \Delta x_k'(G(x_k + \xi_k \Delta x_k, \lambda_k) - G(x_k, \lambda_k))\nabla h_k' \nabla_x L_k], \tag{42}$$

where

$$\zeta_{k,i_k-1} \in (0, \alpha_{k,i_k-1}), \qquad \xi_k \in (0, t_{k,i_k-1}), \qquad \Delta z_{k,i_{k-1}} = (\Delta x_k', \Delta\lambda_{k,i_k-1}')'.$$

Taking limits of (42) for $k \to \infty$, we have, by (39),

$$\nabla S(\hat{x}, \hat{\lambda}; \tilde{c})'\Delta\hat{z} \geq \mu \nabla S(\hat{x}, \hat{\lambda}; \tilde{c})'\Delta\hat{z}.$$

Since $\mu < 1$, this relation contradicts (40) and (38). Hence, the proposition is proved. □

It is now straightforward to prove our main result.

Theorem 4.1. Each accumulation point $(\hat{x}, \hat{\lambda})$ of the sequence $\{x_k, \lambda_k\}$ generated by Algorithm 3.1 is a K–T point for Problem (P).

Proof. By using Assumptions H1, we can introduce for all k the following projection matrix:

$$P_k = I - \nabla h_k(\nabla h_k' \nabla h_k)^{-1}\nabla h_k'. \tag{43}$$

Then, we can write

$$\|\nabla_x L_k\| \leq \|P_k \nabla_x L_k\| + \|(I - P_k)\nabla_x L_k\|$$
$$\leq \|P_k \nabla f_k\| + \|\nabla h_k(\nabla h_k' \nabla h_k)^{-1}\nabla h_k' \nabla_x L_k\|. \tag{44}$$

Now, recalling that Δx_k satisfies the K–T conditions for problem $QP(x_k, B_k)$, we have

$$\|\nabla_x L_k\| \leq \|P_k B_k \Delta x_k\| + \|\nabla h_k(\nabla h_k' \nabla h_k)^{-1}\nabla h_k' \nabla_x L_k\|, \tag{45a}$$

$$h_k = -\nabla h_k' \Delta x_k. \tag{45b}$$

Taking limits of (35) for $k \to \infty$, we obtain, by Assumption H1 and Proposition 4.5,

$$\nabla_x L(\hat{x}, \hat{\lambda}) = 0, \qquad h(\hat{x}) = 0,$$

which proves the theorem. □

5. Superlinear Convergence

In this section, we prove that, if a unit stepsize ensures superlinear rate of convergence, then the acceptability criterion (14) is eventually satisfied by $\alpha_{k,i_k} = 1$.

In this section, we use Assumption H1 and Assumption H2 below.

Assumption H2. Let $\{x_k\}$, $\{\lambda_k\}$, and $\{\Delta x_k\}$ be the sequences produced by Algorithm 3.1. Then,

(i) $\lim_{k \to \infty} x_k = \hat{x}, \qquad \lim_{k \to \infty} \lambda_k = \hat{\lambda};$

(ii) $\lim_{k \to \infty} \|x_k + \Delta x_k - \hat{x}\| / \|x_k - \hat{x}\| = 0.$

From Theorem 4.1, we know that the pair $(\hat{x}, \hat{\lambda})$ of Assumption H2 (i) is a K–T point for Problem (P). Assumption H2 (ii) implies that

$$\lim_{k \to \infty} \|\Delta x_k\| / \|x_k - \hat{x}\| = 1. \tag{46}$$

Before showing our superlinear convergence property, we state some preliminary results.

Lemma 5.1. If $(\hat{x}, \hat{\lambda})$ is a K–T point for Problem (P), then for any $(\bar{x}, \bar{\lambda}) \in \mathbb{R}^{n+m}$, we have

$$\begin{aligned} \nabla h(\bar{x})' \nabla_x L(\bar{x}, \bar{\lambda}) = {} & \nabla h(\bar{x})' \nabla h(\bar{x})(\bar{\lambda} - \hat{\lambda}) \\ & - \int_0^1 G(\bar{x} + t(\hat{x} - \bar{x}), \bar{\lambda})'(\hat{x} - \bar{x})\, dt \\ & + \int_0^1 Q(\bar{x} + t(\hat{x} - \bar{x}), \bar{\lambda} - \hat{\lambda})'(\hat{x} - \bar{x})\, dt, \end{aligned} \tag{47}$$

where

$$Q(x, \lambda) := \nabla_x [\nabla h(x)' \nabla h(x) \lambda].$$

Proof. In fact, by the theorem of the mean, we have

$$\nabla h(\bar{x})'\nabla_x L(\bar{x},\bar{\lambda}) = \nabla h(\bar{x})'\nabla_x f(\bar{x}) + \nabla h(\bar{x})'\nabla h(\bar{x})\hat{\lambda} + \nabla h(\bar{x})'\nabla h(\bar{x})(\bar{\lambda}-\hat{\lambda})$$

$$= \nabla h(\hat{x})'\nabla_x f(\hat{x}) - \int_0^1 \nabla_x[\nabla h'\nabla_x f]'(\bar{x}+t(\hat{x}-\bar{x}))(\hat{x}-\bar{x})\,dt$$

$$+ \nabla h(\hat{x})'\nabla h(\hat{x})\hat{\lambda} - \int_0^1 Q(\bar{x}+t(\hat{x}-\bar{x}),\hat{\lambda})'(\hat{x}-\bar{x})\,dt$$

$$+ \nabla h(\bar{x})'\nabla h(\bar{x})(\bar{\lambda}-\hat{\lambda})$$

$$= \nabla h(\hat{x})'\nabla L(\hat{x},\hat{\lambda}) - \int_0^1 G(\bar{x}+t(\hat{x}-\bar{x}),\bar{\lambda})'(\hat{x}-\bar{x})\,dt$$

$$+ \nabla h(\bar{x})'\nabla h(\bar{x})(\bar{\lambda}-\hat{\lambda}) - \int_0^1 Q(\bar{x}+t(\hat{x}-\bar{x}),\hat{\lambda})'(\hat{x}-\bar{x})\,dt$$

$$+ \int_0^1 Q(\bar{x}+t(\hat{x}-\bar{x}),\bar{\lambda})(\hat{x}-\bar{x})\,dt,$$

which, recalling that the pair $(\hat{x},\hat{\lambda})$ is K–T point for Problem (P), gives (47). □

Proposition 5.1. Assumptions H1 and H2 imply that

$$\lim_{k\to\infty} \|\lambda_k + \Delta\lambda_{k,0} - \hat{\lambda}\| / \|x_k - \hat{x}\| = 0, \tag{48}$$

where the direction $\Delta\lambda_{k,0} \in \mathbb{R}^m$ is defined in Algorithm 3.1.

Proof. By inserting (47) with $\bar{x} = x_k$ and $\bar{\lambda} = \lambda_k$ into (15), we obtain

$$\nabla h_k'\nabla h_k \Delta\lambda_{k,0} = -\nabla h_k'\nabla h_k(\lambda_k - \hat{\lambda}) - G(x_k,\lambda_k)'\Delta x_k$$

$$- \int_0^1 (G(x_k+\xi t_{k,0}\Delta x_k,\lambda_k) - G(x_k,\lambda_k))'\Delta x_k d\xi$$

$$+ \int_0^1 G(x_k + t(\hat{x}-x_k),\lambda_k)'(\hat{x}-x_k)\,dt$$

$$- \int_0^1 Q(x_k + t(\hat{x}-x_k),\lambda_k-\hat{\lambda})'(\hat{x}-x_k)\,dt, \tag{49}$$

where

$$Q(x,\lambda) = \nabla_x[\nabla h(x)'\nabla h(x)\lambda].$$

Now, (49) yields

$$\nabla h_k' \nabla h_k(\lambda_k + \Delta\lambda_{k,0} - \hat{\lambda}) = -G(x_k, \lambda_k)'(x_k + \Delta x_k - \hat{x}) - \int_0^1 (G(x_k + \xi t_{k,0}\Delta x_k, \lambda_k) - G(x_k, \lambda_k))'\Delta x_k d\xi + \int_0^1 (G(x_k + t(\hat{x} - x_k), \lambda_k) - G(x_k, \lambda_k))'(\hat{x} - x_k)\, dt - \int_0^1 Q(x_k + t(\hat{x} - x_k), \lambda_k - \hat{\lambda})'(\hat{x} - x_k)\, dt. \tag{50}$$

Now from (46), Assumptions H1 and H2, the boundedness of $t_{k,0}$, the definition of Q, and Proposition 4.5, it follows that (50) implies (48). □

Finally, we can prove the following result.

Theorem 5.1. If Assumptions H1 and H2 hold, then there exists an iteration index $\hat{k}$ such that

$$x_{k+1} = x_k + \Delta x_k, \quad \lambda_{k+1} = \lambda_k + \Delta\lambda_{k,0}, \quad \text{for all } k \geq \hat{k}. \tag{51}$$

Proof. We assume without loss of generality that $\hat{k} \geq \tilde{k}$, where $\tilde{k}$ is defined in Proposition 4.4. Using the theorem of the mean, we can write

$$f(x_k + \Delta x_k) = f(x_k) + \nabla f(x_k)'\Delta x_k + (1/2)\Delta x_k' \nabla^2 f(x_k + \zeta_k \Delta x_k)\Delta x_k, \tag{52}$$

$$\nabla f(x_k + \Delta x_k)'\Delta x_k = \nabla f(x_k)'\Delta x_k + \Delta x_k' \nabla^2 f(x_k + \xi_k \Delta x_k)\Delta x_k, \tag{53}$$

$$\nabla f(x_k + \Delta x_k)'\Delta x_k = \nabla f(\hat{x})'\Delta x_k + \Delta x_k' \nabla^2 f(\hat{x} + \sigma_k(x_k + \Delta x_k - \hat{x}))(x_k + \Delta x_k - \hat{x}), \tag{54}$$

for ζ_k, ξ_k, and $\sigma_k \in (0, 1)$. By (52), (53), and (54), we get

$$f(x_k + \Delta x_k) = f(x_k) + (1/2)(\nabla f(x_k) + \nabla f(\hat{x}))'\Delta x_k + (1/2)\Delta x_k'(\nabla^2 f(x_k + \zeta_k \Delta x_k) - \nabla^2 f(x_k + \xi_k \Delta x_k))\Delta x_k + (1/2)\Delta x_k' \nabla^2 f(\hat{x} + \sigma_k(x_k + \Delta x_k - \hat{x}))(x_k + \Delta x_k - \hat{x}). \tag{55}$$

Now, taking into account Assumption H1, Assumption H2, and (46), it follows from (55) that

$$f(x_k + \Delta x_k) = f(x_k) + (1/2)(\nabla f(x_k) + \nabla f(\hat{x}))'\Delta x_k + o(\|\Delta x_k\|^2). \tag{56}$$

Similarly, we obtain

$$h(x_k+\Delta x_k)=h(x_k)+(1/2)(\nabla h(x_k)+\nabla h(\hat{x}))'\Delta x_k + o(\|\Delta x_k\|^2). \tag{57}$$

Moreover, we can also write

$$h_i(x_k+\Delta x_k)=\nabla h_i(\hat{x}+\tilde{\sigma}_k(x_k+\Delta x_k-\hat{x}))'(x_k+\Delta x_k-\hat{x}), \tag{58}$$

for $i=1,\dots,m$, and where again $\tilde{\sigma}_k \in (0,1)$. Therefore, from Assumption H2, we have

$$h(x_k+\Delta x_k)=o(\|\Delta x_k\|). \tag{59}$$

In the same way and by using Proposition 5.1, we have

$$\nabla h(x_k+\Delta x_k)'\nabla_x L(x_k+\Delta x_k, \lambda_k+\Delta\lambda_{k,0})=o(\|\Delta x_k\|). \tag{60}$$

Now, (56)-(60) and (1) yield

$$\begin{aligned}
&S(x_k+\Delta x_k, \lambda_k+\Delta\lambda_{k,0}; \tilde{c})\\
&=f(x_k)+(1/2)(\nabla f(x_k)+\nabla f(\hat{x}))'\Delta x_k+(\lambda_k+\Delta\lambda_{k,0})'\\
&\times(h(x_k)+(1/2)(\nabla h(x_k)+\nabla h(\hat{x}))'\Delta x_k)+o(\|\Delta x_k\|^2).
\end{aligned} \tag{61}$$

By using (61), (7), and (12), we obtain

$$\begin{aligned}
&S(x_k+\Delta x_k, \lambda_k+\Delta\lambda_{k,0}; \tilde{c})-S(x_k, \lambda_k; \tilde{c})\\
&=(1/2)(\nabla f_k+\nabla f(\hat{x}))'\Delta x_k\\
&+(\lambda_k+\Delta\lambda_{k,0})'[h_k+(1/2)(\nabla h_k+\nabla h(\hat{x}))'\Delta x_k]-\lambda_k' h_k\\
&-(\tilde{c}/2)\|h_k\|^2-(\eta/2)\|\nabla h_k'\nabla_x L_k\|^2+o(\|\Delta x_k\|^2)\\
&=(1/2)\nabla_x L_k'\Delta x_k+(1/2)\nabla f(\hat{x})'\Delta x_k+\Delta\lambda_{k,0}(h_k+(1/2)\nabla h_k'\Delta x_k)\\
&+(1/2)(\lambda_k+\Delta\lambda_{k,0})'\nabla h(\hat{x})'\Delta x_k-(1/2)(\tilde{c}\|h_k\|^2+\eta\|\nabla h_k\nabla_x L_k\|^2)\\
&+o(\|\Delta x_k\|^2)\\
&=(1/2)(\nabla_x L_k'\Delta x_k-\Delta\lambda_{k,0}\nabla h(x_k)'\Delta x_k-\tilde{c}\|h_k\|^2-\eta\|\nabla h_k\nabla_x L_k\|^2)\\
&+(1/2)\Delta x_k'(\nabla f(\hat{x})+\nabla h(\hat{x})\hat{\lambda}+\nabla h(\hat{x})(\lambda_k+\Delta\lambda_{k,0}-\hat{\lambda}))+o(\|\Delta x_k\|^2)\\
&=(1/2)DS_{k,0}+o(\|\Delta x_k\|^2).
\end{aligned} \tag{62}$$

Now, recalling Proposition 4.2, we have

$$\begin{aligned}
&S(x_k+\Delta x_k, \lambda_k+\Delta\lambda_{k,0}; \tilde{c})-S(x_k, \lambda_k; \tilde{c})\\
&\le \mu DS_{k,0}-\tilde{\gamma}(1/2-\mu)\|\Delta x_k\|^2+o(\|\Delta x_k\|^2).
\end{aligned} \tag{63}$$

Since, by assumption, $\mu < 1/2$ and using Proposition 4.5, we obtain that, for sufficiently large values of k, the inequality

$$S(x_k + \Delta x_k, \lambda_k + \Delta\lambda_{k,0}; \bar{c}) \leq S(x_k, \lambda_k; \bar{c}) + \mu DS_{k,0}$$

is satisfied. Therefore, the theorem is proved. □

By using Theorem 5.1, we have that, if Assumption H2 is satisfied, the algorithm is superlinearly convergent. Conditions that ensure satisfaction of Assumption H2 (ii) are given, for instance, in Refs. 9 and 13.

6. Numerical Results

In this section, we report the numerical results obtained for a set of standard test problems by means of a preliminary implementation of Algorithm 3.1. We refer to Ref. 25 for a detailed description of this implementation; here, we only indicate that the matrix B_{k+1} has been updated by using the technique proposed by Powell in Ref. 7 for preserving positive definiteness.

The algorithm has been tested on problems 39, 40, 77, 78, 79 of Ref. 26 and on problems 373, 375 of Ref. 27. In these problems, we have assumed as starting points the vectors x_1 reported in Refs. 26 and 27 and the vectors λ_1 given by the optimal multipliers of the quadratic subproblems $QP(x_1, B_1)$; however, our numerical experience seems to indicate that different choices for λ_1 does not seriously affect the efficiency of the method.

The computations have been performed in double-precision arithmetic on DIGITAL VAX-11/780. The termination criterion has been

$$(\|\Delta x_k\|^2 + \|h(x_k)\|^2)^{1/2} \leq 10^{-6}.$$

The computational results are reported in Table 1 by specifying, for each test problem (TP), the number n of variables in the problem, the number

Table 1. Numerical results.

TP	n	m	n_I	n_f	n_g	f	$\|h\|$	$\|\nabla_x L\|$
39	4	2	11	11	22	−1.0000	0.1×10^{-10}	0.1×10^{-5}
40	4	3	6	6	12	−0.25000	0.7×10^{-9}	0.3×10^{-9}
77	5	2	13	14	27	0.241505	0.2×10^{-10}	0.3×10^{-6}
78	5	3	8	9	17	−0.29197	0.1×10^{-9}	0.7×10^{-9}
79	5	3	13	14	27	0.078777	0.6×10^{-12}	0.1×10^{-7}
373	9	6	83	263	346	13390.1	0.1×10^{-12}	0.3×10^{-9}
375	10	9	13	19	32	−15.1600	0.1×10^{-7}	0.4×10^{-8}

m of equality constrains, the number n_l of line searches required to attain the convergence, the number n_f of evaluations of the exact augmented Lagrangian (1), the number n_g of the evaluations of the first-order derivatives of the original problem, the value of the objective function f, the value of the norm of the constraints $\|h\|$, and the value of the norm of the gradient of the Lagrangian $\|\nabla_x L\|$ computed at the obtained solution. From Table 1 we can derive that the behavior of the proposed algorithm appears to be at least competitive with that of the most effective techniques presently available.

In order to get further information about the capability of the exact augmented Lagrangian (1) of eventually allowing units stepsizes, we have also applied Algorithm 3.1 to the following well-known problem proposed by Maratos (Ref. 28):

$$\min_{x_1, x_2} x_1^2 + x_2^2,$$

$$(x_1+1)^2 + x_2^2 - 4 = 0.$$

The solution of this problem is $x^* = (1, 0)'$ and the corresponding Lagrange multiplier is $\lambda^* = 0.5$. To initialize our algorithm close to the solution, we have used as starting points the following series of vectors (see Ref. 11):

$$x_1^i = \left(1 - 0.1^i, \sqrt{4-(2-0.1^i)^2}\right)', \qquad \lambda_1^i = 0.5 + 0.1^i,$$

for $i = 1, \ldots, 7$. In all the cases, Algorithm 3.1 performs very well; in fact, it uses the steplength one for all the iterations. The same results can be obtained if we choose as starting points λ_1^i the Lagrange multipliers of the quadratic subproblems $QP(x_1^i, B_1)$, for $i = 1, \ldots, 7$.

Finally, we remark that, in all test problems considered in this section, the system (11) is solved at any iteration of Algorithm 3.1 only once. This confirms that, as we hinted in Section 3, suitable choices for the parameters $t_{k,0}$ produce finite-difference approximations to first derivatives of $\nabla h(x)'\nabla_x L(x, \lambda)$, which are adequate in practice; hence, the algorithm does not need to adjust the finite-difference stepsize.

7. Conclusions

In this paper, we have proposed a recursive quadratic programming algorithm for nonlinear programming problems with equality constraints which uses as merit function a differentiable exact augmented Lagrangian. An attractive feature of this approach is that the unconstrained minimization of the exact augmented Lagrangian is performed without the calculation of any second derivatives. We have also proved that the proposed algorithm possesses global and superlinear properties.

In order to obtain some feel for the effectiveness of our method, we have tested a preliminary implementation of the algorithm with a few test problems. The obtained numerical results are very encouraging and seem to indicate that our approach appears quite promising.

References

1. WILSON, R. B., *A Simplicial Algorithm for Concave Programming*, PhD Dissertation, Graduate School of Business Administration, Harvard University, Boston, Massachusetts, 1963.
2. HAN, S. P., *Superlinearly Convergent Variable Metric Algorithm for General Nonlinear Programming Problems*, Mathematical Programming, Vol. 11, pp. 263-282, 1976.
3. HAN, S. P., *A Globally Convergent Method for Nonlinear Programming*, Journal of Optimization Theory and Application, Vol. 22, pp. 297-309, 1977.
4. BERTSEKAS, D. P., *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, New York, 1982.
5. BIGGS, M. C., *On the Convergence of Some Constrained Minimization Algorithms Based on Recursive Quadratic Programming*, Journal of the Institute of Mathematics and Its Applications, Vol. 21, pp. 67-82, 1978.
6. BARTHOLOMEW-BIGGS, M. C., *A Recursive Quadratic Programming Algorithm Based on the Augmented Lagrangian Function*, Report No. NOC 139, The Hatfield Polytechnic, 1983.
7. POWELL, M. J. D., *A Fast Algorithm for Nonlinearly Constrained Optimization Calculations*, Numerical Analysis Dundee 1977, Edited by G. A. Watson, Springer-Verlag, Berlin, Germany, pp. 144-157, 1978.
8. POWELL, M. J. D., *The Convergence of Variable Metric Methods for Nonlinear Constrained Optimization Calculations*, Nonlinear Programming 3, Edited by O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, Academic Press, New York, New York, pp. 27-63, 1978.
9. POWELL, M. J. D., *Variable Metric Methods for Constrained Optimization*, Mathematical Programming, The State of the Art, Edited by A. Bachem, M. Grötschel, and B. Korte, Springer-Verlag, Berlin, Germany, pp. 288-311, 1983.
10. POWELL, M. J. D., *Methods for Nonlinear Constraints in Optimization Calculations*, Report No. DAMTP-1986/NA5, University of Cambridge, 1983.
11. SCHITTKOWSKI, K., *The Nonlinear Programming Method of Wilson, Han, and Powell with Augmented Lagrangian Type Line Search Function*, Numerische Mathematik, Vol. 38, pp. 83-114, 1981.
12. SCHITTKOWSKI, K., *On the Convergence of a Sequential Quadratic Programming Method with an Augmented Line Search Function*, Mathematische Operationsforschung und Statistik, Ser. Optimization, Vol. 14, pp. 197-216, 1983.

13. BOGGS, P. T., TOLLE, J. W., and WANG, P., *On the Local Convergence of Quasi-Newton Methods for Constrained Optimization*, SIAM Journal on Control and Optimization, Vol. 20, pp. 161-171, 1982.
14. CHAMBERLAIN, R. M., POWELL, M. J. D., LEMARECHAL, C., and PEDERSEN, H. C., *The Watchdog Technique for Forcing Convergence in Algorithms for Constrained Optimization*, Mathematical Programming Study, Vol. 16, pp. 1-17, 1982.
15. FLETCHER, R. E., *Second-Order Corrections for Nondifferentiable Optimization*, Numerical Analysis Dundee 1981, Edited by G. A. Watson, Springer-Verlag, Berlin, Germany, pp. 144-157, 1982.
16. MAYNE, D. Q., and POLAK, E., *A Superlinearly Convergent Algorithm for Constrained Optimization Problems*, Mathematical Programming Study, Vol. 16, pp. 45-61, 1982.
17. FUKUSHIMA, M., *A Successive Quadratic Programming Algorithm with Global and Superlinear Convergence Properties*, Mathematical Programming, Vol. 35, pp. 253-264, 1986.
18. GABAY, D., *Reduced Quasi-Newton Methods with Feasibility Improvement for Nonlinearly Constrained Optimization*, Mathematical Programming Study, Vol. 16, pp. 18-44, 1982.
19. DIXON, L. C. W., *Exact Penalty Function Methods in Nonlinear Programming*, Report No. NOC 103, The Hatfield Polytechnic, 1979.
20. DIXON, L. C. W., *On the Convergence Properties of Variable Metric Recursive Quacratic Programming Methods*, Report No. NOC 110, The Hatfield Polytechnic, 1980.
21. POWELL, M. J. D., and YUAN, Y., *A Recursive Quadratic Programming Algorithm That Uses Differentiable Exact Penalty Functions*, Mathematical Programming, Vol. 3, pp. 265-278, 1986.
22. FLETCHER, R. E., *An Exact Penalty Function for Nonlinear Programming with Inequalities*, Mathematical Programming, Vol. 5, pp. 129-239, 1973.
23. DEW, M. C., *An Algorithm for Nonlinear Equality Constrained Optimization Using the Di Pillo-Grippo Exact Penalty Function*, Report No. NOC 159, The Hatfield Polytechnic, 1985.
24. DI PILLO, G., and GRIPPO, L., *A New Class of Augmented Lagrangians in Nonlinear Programming*, SIAM Journal on Control and Optimization, Vol. 17, pp. 618-628, 1979.
25. LUCIDI, S., *A RQP Algorithm That Uses an Exact Augmented Lagrangian Function*, IASI-CNR, Report No. 252, 1989.
26. HOCK, W., and SCHITTKOWSKI, K., *Text Examples for Nonlinear Programming Codes*, Springer-Verlag, Berlin, Germany, 1981.
27. SCHITTKOWSKI, K., *More Text Examples for Nonlinear Programming Codes*, Springer-Verlag, Berlin, Germany, 1987.
28. MARATOS, N., *Exact Penalty Function Algorithms for Finite-Dimensional and Control Optimization Problems*, PhD Thesis, Imperial College, London, 1978.