Mathematical Programming 24 (1982) 137–161
North-Holland Publishing Company

NONLINEAR PROGRAMMING VIA AN EXACT PENALTY FUNCTION: GLOBAL ANALYSIS*

T.F. COLEMAN**
Applied Mathematics Division, Argonne National Laboratory, Argonne, IL, U.S.A.

A.R. CONN
Department of Computer Science, University of Waterloo, Waterloo, Ont., Canada

Received 21 July 1980
Revised manuscript received 1 March 1981

In this paper we motivate and describe an algorithm to solve the nonlinear programming problem. The method is based on an exact penalty function and possesses both global and superlinear convergence properties. We establish the global qualities here (the superlinear nature is proven in [7]). The numerical implementation techniques are briefly discussed and preliminary numerical results are given.

Key words: Nonlinear Programming, Exact Penalty Methods, Successive Quadratic Programming.

1. Introduction

The nonlinear programming problem can be described as

$$\begin{aligned}&\text{minimize} \quad f(x),\\ &\text{subject to} \quad \phi_i(x) \geq 0, \quad i = 1, \dots, m, \end{aligned} \tag{1.1}$$

where m is a positive integer, and f, ϕ_i, $i = 1, \dots, m$ are twice continuously differentiable and map $\mathbf{R}^n$ to $\mathbf{R}^1$. (Our method and the theoretical results are *not* limited to the inequality constrained problem: we omit the equality constraints in this paper, to simplify the presentation.) The major purpose of this paper is to describe and motivate a procedure to solve (1.1), and to establish that this method possesses global convergence properties (regardless of starting point). The method has a fast (2-step superlinear) asymptotic convergence rate: this is established in [7].

In Section 4 we describe a sound numerical procedure to implement the conceptual algorithm, and we present some preliminary numerical results in Section 6.

2. Motivation

The method is based on the minimization of an exact penalty function p, given below in (2.1). This unconstrained function p is not readily minimized by

* This work is supported in part by NSERC Grant No. A8639 and the U.S. Dept. of Energy.
** Now at Department of Computer Science, Cornell University, Ithaca, New York.

conventional gradient techniques because it is not everywhere differentiable. Nevertheless, it was demonstrated in [9] that conventional techniques can be adapted to enable the numerical minimization of p. In broad terms, the approach in [9] is as follows: At the current point separate p into two parts—the differentiable part (including the objective function and the *clearly inactive* constraints), and the nondifferentiable part (including the *nearly active* constraints). A useable descent direction is then derived by attempting to decrease the first (differentiable) part of p while trying to maintain the value of the second (nondifferentiable) part of p. In [9], a numerical method was given to obtain these two objectives up to first-order changes. (Thus, a projected gradient method was described.) Here we obtain the two objectives up to second-order changes. (Thus we describe a projected Newton method.) We note that in both cases well-established numerical techniques can be used.

2.1. The horizontal direction, h

We suggest transforming (1.1) into

$$\underset{x}{\text{minimize}}\ p(x,\mu)=f(x)-\frac{1}{\mu}\sum_{i=1}^{m}\min(0,\phi_i(x)), \tag{2.1}$$

where μ is a positive scalar.

Since, under certain conditions, local minima to (2.1) are also solutions to (1.1), the unconstrained minimization of this exact penalty function p may yield a solution to (1.1) [4, 5, 6]. The major difficulty in designing an algorithm to minimize p is: how do we overcome the nondifferentiable nature of p?

Let us consider this problem in detail. Let ϵ be a 'small' positive number used to identify the near-active (or, ϵ-active) constraint set and suppose that the first t constraints are ϵ-active at x^1. That is, suppose

$$|\phi_i(x^1)|\le\epsilon,\quad i=1,\dots,t, \tag{2.2}$$

and

$$\phi_i(x^1)<-\epsilon,\quad i=t+1,\dots,m. \tag{2.3}$$

(The clearly feasible constraints do not affect this argument, so we assume, for the present, that there are none.)

Let

$$p_1(x)=f(x)-\frac{1}{\mu}\sum_{i=t+1}^{m}\phi_i(x),$$

and thus, in a neighbourhood of x^1,

$$p(x)=p_1(x)-\frac{1}{\mu}\sum_{i=1}^{t}\min(0,\phi_i(x)). \tag{2.4}$$

Clearly p_1 is differentiable over $\mathbf{R}^n$; we can view p_1 as the differentiable portion of p in a neighbourhood of x^1. To develop a *first-order method* to minimize p, we should consider minimizing the first-order change in p. That is, if we are at the point x^1, and β is some positive scalar, we should consider solving

$$\underset{h,\|h\|\leq\beta}{\text{minimize}}\ \nabla p_1(x^1)^{\mathrm{T}}h - \frac{1}{\mu}\sum_{i=1}^{t} {}''\nabla\phi_i^{\mathrm{T}}h'', \tag{2.5}$$

where the quotations are used here to indicate terms of that form. That is, 'a' is either 0 or $-|a|$. (We use 'h' instead of the more usual 'd' since our development will eventually lead to a direction termed the *h*orizontal direction, to contrast with a *v*ertical direction v, introduced later and satisfying $h^{\mathrm{T}}v = 0$.) Here, and throughout this paper, we use the convention that the expression $\|\cdot\|$ will denote the Euclidean norm unless it is specifically indicated otherwise.

Clearly the computing of a direction to solve (2.5) will be nontrivial due to the awkward term in quotations. The occurrence of this term is due, of course, to the nondifferentiable nature of p at constraint boundaries. If we could restrict h so that

$$\nabla\phi_i^{\mathrm{T}}h = 0, \quad i = 1, \dots, t, \tag{2.6}$$

then we would have a computable problem. That is, let us attempt to find a direction h which *minimizes the* **change** *in the penalty function p* (up to *first-order terms*) subject to the **change** *in the active constraints being zero* (up to *first-order terms*). Thus our constrained direction finding problem is,

$$\begin{aligned}&\underset{h,\|h\|\leq\beta}{\text{minimize}} \quad \nabla p_1(x^1)^{\mathrm{T}}h,\\ &\text{subject to} \quad \nabla\phi_i^{\mathrm{T}}h = 0, \quad i = 1, \dots, t.\end{aligned} \tag{2.7}$$

The solution to (2.7) is

$$h^* = -\alpha P\nabla p_1(x^1), \tag{2.8}$$

where P is the orthogonal projector onto the space orthogonal to $\nabla\phi_i$, $i = 1, \dots, t$. (We assume for the moment, that $P\nabla p_1 \neq 0$, and α is chosen so that $\|h^*\| = \beta$.)

The method of Conn and Pietrykowski [9] is, in large, based on the above observations. This method has global convergence properties but possesses (in general) only a linear convergence rate. Clearly we can expect no more since h^* is obtained by minimizing only up to first-order terms.

Let us consider attempting to find a direction which *minimizes the* **change** *in the penalty function p* (up to *second-order terms*) subject to the **change** *in the active constraints being zero* (up to ***second-order*** *terms*). That is, consider

$$\begin{aligned}&\underset{h}{\text{minimize}} \quad \nabla p_1^{\mathrm{T}}h + \tfrac{1}{2}h^{\mathrm{T}}\nabla^2 p_1 h - \frac{1}{\mu}\sum_{i=1}^{t} {}''\,\nabla\phi_i^{\mathrm{T}}h + \tfrac{1}{2}h^{\mathrm{T}}\nabla^2\phi_i h'',\\ &\text{subject to} \quad \nabla\phi_i^{\mathrm{T}}h + \tfrac{1}{2}h^{\mathrm{T}}\nabla^2\phi_i h = 0, \quad i = 1, \dots, t.\end{aligned} \tag{2.9}$$

Clearly (2.9) is equivalent to

$$\underset{h}{\text{minimize}} \quad \nabla p_1^{\mathrm{T}} h + \tfrac{1}{2} h^{\mathrm{T}} \nabla^2 p_1, h, \tag{2.10}$$
$$\text{subject to} \quad \nabla \phi_i^{\mathrm{T}} h + \tfrac{1}{2} h^{\mathrm{T}} \nabla^2 \phi_i h = 0, \quad i = 1, \dots, t.$$

Unfortunately, a solution to (2.10) cannot be explicitly computed. We can, however, obtain a computable and useable approximation to problem (2.10) in the following way. Problem (2.10) is equivalent to

$$\begin{aligned} \min_h \max_\lambda L(h, \lambda) = {} & \nabla p_1^{\mathrm{T}} h + \tfrac{1}{2} h^{\mathrm{T}} \nabla^2 p_1 h \\ & - \sum_{i=1}^{t} \lambda_i (\nabla \phi_i^{\mathrm{T}} h + \tfrac{1}{2} h^{\mathrm{T}} \nabla^2 \phi_i h), \end{aligned} \tag{2.11}$$

which we can approximate by

$$\min_h \max_\lambda \tilde{L}(h, \lambda) = \nabla p_1^{\mathrm{T}} h + \tfrac{1}{2} h^{\mathrm{T}} \nabla^2 p_1 h - \frac{1}{2} \sum_{i=1}^{t} \tilde{\lambda}_i h^{\mathrm{T}} \nabla^2 \phi_i h - \sum_{i=1}^{t} \lambda_i \nabla \phi_i^{\mathrm{T}} h,$$

or

$$\begin{aligned} \min_h \max_\lambda \tilde{L}(h, \lambda) = {} & \tfrac{1}{2} h^{\mathrm{T}} \Big(\nabla^2 p_1 - \sum_{i=1}^{t} \tilde{\lambda}_i \nabla^2 \phi_i \Big) h \\ & + h^{\mathrm{T}} \Big(\nabla p_1 - \sum_{i=1}^{t} \lambda_i \nabla \phi_i \Big), \end{aligned} \tag{2.12}$$

where $\tilde{\lambda}$ is a computable approximation to λ. (We discuss this approximation later.) Differentiating $\tilde{L}$ with respect to λ gives

$$\nabla \phi_i^{\mathrm{T}} h = 0, \quad i = 1, \dots, t. \tag{2.13}$$

Let us define a matrix $A = (\nabla \phi_1(x^1), \dots, \nabla \phi_t(x^1))$ and an $n \times (n-t)$ matrix Z satisfying

$$A^{\mathrm{T}} Z = 0, \tag{2.14}$$
$$Z^{\mathrm{T}} Z = I_{(n-t)}. \tag{2.15}$$

Thus by (2.13) we can use the transformation $h \to Zw$, and (2.12) becomes

$$\min_w \tfrac{1}{2} w^{\mathrm{T}} Z^{\mathrm{T}} \Big(\nabla^2 p_1 - \sum_{i=1}^{t} \tilde{\lambda}_i \nabla^2 \phi_i \Big) Z w + \nabla p_1^{\mathrm{T}} Z w. \tag{2.16}$$

If $Z^{\mathrm{T}}(\nabla^2 p_i - \sum_{i=l}^{t} \tilde{\lambda}_i \nabla^2 \phi_i) Z$ is positive definite, then the solution to (2.16) can be obtained by solving

$$\Big(Z^{\mathrm{T}} \Big(\nabla^2 p_1 - \sum_{i=1}^{t} \tilde{\lambda}_i \nabla^2 \phi_i \Big) Z \Big) w = - Z^{\mathrm{T}} \nabla p_1. \tag{2.17}$$

We gain an approximate solution to our original system (2.10) by setting

$h^* \leftarrow Zw^*$, where w^* solves (2.17).

Provided $Z^{\mathrm{T}}\nabla p_1 \neq 0$, the direction h^* is a descent direction for p, at x^1. In addition, as our development suggests, h^* is the second-order analog of the projected first-order direction (2.8), used by Conn and Pietrzykowski [9]. In practise, the projected Hessian need not be computed but rather approximated by a positive definite matrix, $Z^{\mathrm{T}}BZ$. Since the sufficiency conditions for p ensure that the true projected Hessian is positive definite at a local minimum of p[6], it is reasonable to restrict the projected Hessian approximating matrices to this class.

This is not the full story, however. In particular, when $\|Z^{\mathrm{T}}\nabla p_1\|$ is 'large' (and thus we are likely far from a stationary point of p([6]; this paper, Section 5), dual estimates $\{\lambda_i\}$ have little meaning. In addition, when far from a stationary point of p, the objective function and the violated constraints should dominate the penalty function changes: it is reasonable then, in this region, to ignore the *second-order* changes in the current activities. Thus, when far from a stationary point, $Z^{\mathrm{T}}BZ$ is a positive definite approximation to $Z^{\mathrm{T}}\nabla^2 p_1 Z$, and $\tilde{\lambda}$ is not computed. (Alternatively, one can view $\tilde{\lambda}$ as being approximated by the zero vector.)

2.2. The dual estimates, $\tilde{\lambda}$

Let us suppose that $\{x^k\}\to\bar{x}$, where $\bar{x}$ is a stationary point for p (see [6]). By definition there exists a vector $\bar{\lambda}$ such that

$$\nabla f(\bar{x}) - \frac{1}{\mu}\sum_{i\in I_{\bar{v}}} \nabla\phi_i(\bar{x}) = \sum_{i\in I_{\bar{A}}} \bar{\lambda}_i \nabla\phi_i(\bar{x}), \tag{2.18}$$

where $I_{\bar{v}}$ and $I_{\bar{A}}$ are the *v*iolated and the *a*ctive constraint sets of $\bar{x}$. That is,

$$I_{\bar{v}} = \{i \mid \phi_i(\bar{x}) < 0\},$$

and

$$I_{\bar{A}} = \{i \mid \phi_i(\bar{x}) = 0\}.$$

Similarly, we define

$$I_A^\epsilon(x^k) = \{i \mid |\phi_i(x^k)| \leq \epsilon\},$$

and

$$I_v^\epsilon(x^k) = \{i \mid \phi_i(x^k) < -\epsilon\}, \quad \text{where } \epsilon > 0.$$

Clearly, if for all k sufficiently large, $I_A^\epsilon(x^k) = I_{\bar{A}}$, and $I_v^\epsilon(x^k) = I_{\bar{v}}$, then $\lambda^k \to \bar{\lambda}$, where λ^k is a *least-squares* solution to

$$A_k\lambda = \nabla f(x^k) - \frac{1}{\mu}\sum_{i\in I_v^\epsilon(x^k)} \nabla\phi_i^k, \tag{2.19}$$

where $\phi_i^k = \phi_i(x^k)$, $A_k = (\nabla\phi_1^k, \ldots, \nabla\phi_{t_k}^k)$, and $t_k = |I_A^\epsilon(x^k)|$. Define an $n \times (n - t_k)$

matrix Z_k so that Z_k satisfies

$$A_k^{\mathrm{T}} Z_k = 0, \tag{2.20}$$

and

$$Z_k^{\mathrm{T}} Z_k = I_{(n-t_k)}. \tag{2.21}$$

It follows that if the active and violated constraints have been correctly identified then $\|Z_k^{\mathrm{T}} \bar{\nabla} p\| \to 0$, as $x^k \to \bar{x}$, where

$$\bar{\nabla} p = \nabla f(x^k) - \frac{1}{\mu} \sum_{i \in I_v^{\epsilon}(x^k)} \nabla \phi_i^k.$$

(We call $\bar{\nabla} p$ the pseudo-gradient of p.) Thus, when $\|Z_k^{\mathrm{T}} \bar{\nabla} p\|$ becomes sufficiently small the least-squares solution to (2.19) will usually give reasonable estimates to the multipliers $\{\bar{\lambda}_i\}$ used in (2.18). Conversely, if $\|Z_k^{\mathrm{T}} \nabla p\|$ is large the least-squares solution to (2.19) will likely bear little similarity to $\bar{\lambda}$ (particularly if $I_A^{\epsilon}(x^k) \neq I_{\bar{A}}$).

The actual computation of the least squares solution to (2.19) will be discussed in Section 4.

2.3. *The vertical step, v*

Let us suppose that x^* satisfies the second-order sufficiency conditions for problem (2.1) [6]. Further, we suppose that ϵ and $\|x^k - x^*\|$ are sufficiently small so that $I_A^{\epsilon}(x^k) = I_A^*$ and $I_v^{\epsilon}(x^k) = I_v^*$. The horizontal direction h^k (obtained by solving a system similar to (2.17)) decreases p, while *attempting* to keep the activities constant. But, at x^k the activities are not precisely zero (they are ϵ-active). In addition, the step in the direction h^k *will* change the active constraint values to some degree. Therefore, it seems reasonable to try and satisfy the ϵ-active constraints more precisely, when x^k is close to x^*. We do this by means of a vertical step, v^k:

$$v^k = -A_k (A_k^{\mathrm{T}} A_k)^{-1} \Phi(x^k + \alpha_k h^k), \tag{2.22}$$

where Φ is the vector of active constraint values, ordered in the same fashion as A_k, A_k is the $n \times t_k$ matrix of active constraint gradients, *evaluated at* x^k, and α_k is the stepsize. (We assume, for the purposes of description, that the columns of A_k are linearly independent; v^k is *not* computed in this fashion—we describe the computation in Section 4.)

Note that with the use of a Taylor expansion, it can be seen that

$$\begin{aligned} \Phi(x^k + \alpha_k h^k + v^k) &\approx \Phi(x^k + \alpha_k h^k) - A_k^{\mathrm{T}} v^k \\ &= 0. \end{aligned}$$

2.4. *Dropping a constraint*

Again let us suppose that $\{x^k\} \to \bar{x}$, where $\bar{x}$ is a stationary point for p. Thus (2.18) will be satisfied (by definition). But suppose that $\bar{\lambda}_j \notin [0, 1/\mu]$. Consider the direction

$$h = \sigma_j P^j \nabla\phi_j(\bar{x}), \tag{2.23}$$

where $\sigma_j = -\operatorname{sgn}(\bar{\lambda}_j)$, and P^j projects orthogonal to $\{\nabla\phi_i(\bar{x}) \mid i \in I_{\bar{A}} - \{j\}\}$. Assuming that the gradients of the active constraints at $\bar{x}$ are linearly independent, it is easy to see that (2.23) gives a descent direction for p, at $\bar{x}$ (part 2 of Theorem 1).

If x^k is sufficiently close to $\bar{x}$, and $I_A^\epsilon(x^k) = I_{\bar{A}}$, then the multiplier *estimate*, λ^k, will satisfy

$$\lambda_j^k \notin [0, 1/\mu]. \tag{2.24}$$

It is then easy to show that there is a neighbourhood of $\bar{x}$ in which

$$h^k = \sigma_j^k P_k^j \nabla\phi_j(x^k) = \sigma_j^k Z_{k\setminus j} Z_{k\setminus j}^{\mathrm{T}} \nabla\phi_j(x^k) \tag{2.25}$$

is a descent direction for p at x^k. The matrix $Z_{k\setminus j}$ satisfies $Z_{k\setminus j}^{\mathrm{T}} Z_{k\setminus j} = I_{n-t_k+1}$ and $\nabla\phi_i(x^k)^{\mathrm{T}} Z_{k\setminus j} = 0$ if $i \in I_A^\epsilon(x^k) - \{j\}$.

We note that dropping a constraint and attempting a vertical step are complementary activities. When $\|Z_k^{\mathrm{T}}\bar{\nabla} p\|$ is 'small' the dual variables $\{\lambda_i^k\}$ are computed: a vertical step is attempted if $\lambda_i^k \in [0, 1/\mu]$ for all i in $I_A^\epsilon(x^k)$, otherwise we *attempt* to drop a constraint. (The size of $\|Z_k^{\mathrm{T}}\bar{\nabla} p\|$ and the vector $\tilde{\lambda}$ indicate whether or not to *attempt* these steps—the indicated step is usually taken *only* if a sufficient decrease in p can be guaranteed.)

2.5. *Direction choice strategy*

We have defined 3 directions:

(i) $$h^k = -Z_k(Z_k^{\mathrm{T}} B_k Z_k)^{-1} Z_k^{\mathrm{T}} \bar{\nabla} p(x^k), \tag{2.26a}$$

(ii) $$v^k = -A_k(A_k^{\mathrm{T}} A_k)^{-1} \Phi(x^k + \alpha_k h^k), \tag{2.26b}$$

(iii) $$\tilde{h}^k = \sigma_j^k Z_{k\setminus j} Z_{k\setminus j}^{\mathrm{T}} \nabla\phi_j(x^k). \tag{2.26c}$$

Determining the optimal combination of directions to use in any given circumstance (in combination with the best modification strategy for ϵ) goes beyond the scope of this paper. Here we suggest a strategy which is both simple and reasonable for the well-scaled problem. The theoretical results will be limited to this simple algorithm, however, it is clear that more complicated strategies could be used without violating the convergence properties.

When $\|Z_k^{\mathrm{T}}\bar{\nabla} p\|$ is 'large', that is when $\|Z_k^{\mathrm{T}}\bar{\nabla} p\| > \Lambda$, we choose to use only the direction h^k, where $Z_k^{\mathrm{T}} B_k Z_k$ approximates

$$Z_k^{\mathrm{T}}\Big(\nabla^2 f(x^k) - \frac{1}{\mu} \sum_{i \in I_v^\epsilon(x^k)} \nabla^2 \phi_i(x^k)\Big) Z_k.$$

When $\|Z_k^{\mathrm{T}}\bar{\nabla} p\| \le \Lambda$, then the multiplier estimates become important. A direction (iii) is attempted if there exists a multiplier estimate which is not in the range

$[0, 1/\mu]$. If this is the case, then direction (iii) is accepted only if a sufficient decrease can be guaranteed: that is we use $\tilde{h}^k$ if

$$(\bar{\nabla} p + \min(0, \sigma_j^k)\nabla\phi_j)^{\mathrm{T}}\tilde{h}^k < -\delta, \quad \text{for some } \delta > 0, \tag{2.27}$$

and

$$\sigma_j^k = -\operatorname{sgn}(\lambda_j^k), \quad \lambda_j^k \notin [0, 1/\mu]. \tag{2.28}$$

If either of the above situations occur, then a stepsize α_k must be determined, and

$$x^{k+1} \leftarrow x^k + \alpha_k h^k \tag{2.29}$$

or

$$x^{k+1} \leftarrow x^k + \alpha_k \tilde{h}^k.$$

Since p is a piecewise differentiable function, we use a special line search which we describe in the next section.

The second use of the multiplier estimate is to properly include constraint curvature information in the Hesian approximation. That is, when $\|Z_k^{\mathrm{T}}\bar{\nabla} p\| \leq \Lambda$, $Z_k^{\mathrm{T}} B_k Z_k$ approximates

$$Z_k^{\mathrm{T}}\Big(\nabla^2 f(x^k) - \frac{1}{\mu}\sum_{i\in I_v^\epsilon(x^k)} \nabla^2\phi_i(x^k) - \sum_{i\in I_A^\epsilon(x^k)} \lambda_i \nabla^2\phi_i(x^k)\Big)Z_k. \tag{2.30}$$

In addition, provided the multiplier estimates are in the range $[0, 1/\mu]$, our search direction becomes

$$d^k = h^k + v^k.$$

Here we attempt a stepsize of one: if a sufficient decrease is not observed then a line search is performed along the direction h^k.

We have chosen to use a very simple strategy to modify ϵ and Λ. If the dropping step (iii) or the 'Newton step' $h^k + v^k$ are unsuccessfully attempted, then we reduce both parameters by a factor of 2. The initial choice of ϵ, Λ is arbitrary provided ϵ, $\Lambda > 0$.

We realize that the direction choice mechanism described above is somewhat arbitrary. As the form of the convergence proof will demonstrate, there are any number of selection rules which will supply global convergence properties. The particular process we have described is based on limited numerical testing. (It is quite possible that further and more specialized numerical experimentation will suggest an alternative selection scheme.)

In the next section we present statements of the basic algorithm and a sub-algorithm (the special line search). The implementation of the numerical procedures is duscussed in the following section.

3. Algorithms

In this section we describe three algorithms. Algorithm 1 is the line search procedure designed to exploit the piecewise nature of p. Algorithm 2 is our basic method designed to minimize p with any given but *fixed* μ. Finally, we describe Algorithm 3: this procedure is wrapped around Algorithm 2 and automatically reduces μ when it appears that μ is too large (that is, when it appears we are not converging to a point feasible to (1.1)).

3.1. Line search procedure

Since a stepsize of unity is always taken when we are in a neighbourhood of x^* (a point satisfying second-order sufficiency conditions) it is expected that this procedure will be used only when 'far' from a solution.

Suppose that we are at a point $x^1 \in \mathbf{R}^n$ and that we have determined a direction h as in (2.26). If the functions f, ϕ_i, $i = 1, \dots, m$ are all *linear*, it is easy to see that either p is unbounded below in the direction h, or a minimum (along h) occurs at a 'breakpoint' of p. ($\bar{x}$ is a breakpoint along h, if for at least one $j \notin I_A(x^1)$, $\phi_j(\bar{x}) = 0$. Since we are describing the linear situation here, we can assume that $\epsilon = 0$.) Thus an algorithm to determine the minimum of p, in the direction h is straightforward: determine all breakpoints and find the minimum penalty function value at these breakpoints. Let us consider this linear case in more detail.

Suppose that y^1 is the *first* breakpoint along the (positive) direction h from x^1, corresponding to the function ϕ_1. Thus, $\phi_1(y^1) = 0$, and $\phi_1(x^1) \neq 0$. Since h is a descent direction for p at x^1, it follows that $\bar{\nabla} p(x^1)^{\mathrm{T}} h < 0$. But the pseudo-gradient of p, $\bar{\nabla} p$, will remain unchanged (and thus p will continue to decrease) until y^1 is reached ($y_1 = x^1 + \alpha_1 h$, where $\alpha_1 = -\phi_1(x^1)/\nabla\phi_1^{\mathrm{T}} h$). As we move *past* y^1 the pseudo-gradient of p becomes

$$\bar{\nabla} p(y^1 + \Delta h) = \bar{\nabla} p(x^1) + \frac{1}{\mu} \operatorname{sgn}(\nabla\phi_1^{\mathrm{T}} h) \nabla\phi_1,$$

where 'Δ' is any small but positive quantity. (We assume, for descriptive purposes, that the breakpoints are distinct.) Clearly if $\bar{\nabla} p(y^1 + \Delta h)^{\mathrm{T}} h \geq 0$, then y^1 is a minimum of p in the direction h. If $\bar{\nabla} p(y^1 + \Delta h)^{\mathrm{T}} h < 0$, p continues to decrease along h—thus the next breakpoint is determined and the argument is repeated. If we progress through the entire sorted list of breakpoints and p continues to decrease past the last breakpoint then the procedure returns with a pointer indicating unboundedness.

Define $I_M = \{1, \dots, m\}$, and $I_A = I_A(x^1) = \{i \in I_M \mid \phi_i(x^1) = 0\}$.

Algorithm 1. (the line search—linear case).

Step 1: $a_0 \leftarrow \bar{\nabla} p^{\mathrm{T}} h < 0$ (by assumption),

```
        $\gamma_i \leftarrow -\phi_i(x^1)/\nabla\phi_i^T h, \quad i \in I_M \setminus I_A,$
        $I_0 \leftarrow \{i \in I_M \setminus I_A \mid \gamma_i > 0\},$
        $k \leftarrow 0, \qquad l_0 \leftarrow 0, \qquad \gamma_0 \leftarrow 0.$
Step 2: If $I_k = \phi$, return with a message that $p$ is unbounded below.
Step 3: Determine $l_k$ such that
        $\gamma_{l_k} \le \gamma_i, \quad \forall i \in I_k,$
        $a_{k+1} \leftarrow a_k + \sigma_{l_k} \cdot \nabla\phi_{l_k}^T h, \quad (\sigma_{l_k} = \mathrm{sgn}(\nabla\phi_{l_k}^T h)).$
Step 4: If $a_{k+1} \ge 0$, go to Step 5
        else $I_{k+1} \leftarrow I_k - \{l_k\}$
        $k \leftarrow k+1$, go to Step 2.
Step 5: $x^2 \leftarrow x^1 + \gamma_{l_k} h$ return.
```

The extension of this procedure is not difficult if we are content with *estimating* the location of a possible minimum breakpoint along h. (Note that, in the nonlinear case, the minimum of p along h need not be a breakpoint.) Our line search makes *linear approximation* to all functions and, with two exceptions, performs the version of Algorithm 1 listed above. In the nonlinear case, $I_k = \phi$ does not imply p is unbounded. Thus Step 2 is replaced with

```
Step 2': If $I_k = \phi$ and $k = 0$, $\gamma_{l_k} \leftarrow 0$, go to Step 5.
         If $I_k = \phi$ and $k > 0$, $\gamma_{l_k} \leftarrow \gamma_{l_{k-1}}$, go to Step 5.
```

The procedure we have described thus far is clearly not sufficient since the point that is returned, x^2, does *not* (due to nonlinearity) necessarily satisfy

$$p(x^2) < p(x^1).$$

We rectify this situation in the following way. Let δ be a 'small' positive constant. Step 5 is replaced with

```
Step 5: $x^2 \leftarrow x^1 + \gamma_{l_k} h$
        If $(p(x^2) < p(x^1) - \delta)$ return,
        else
        perform a cubic interpolation minimization between $x^1$ and
        $x^1 + \tau h$, $\tau > 0$;
        return.
```

(Since $\bar{\nabla} p^T h < 0$ and p (excluding the 'active' constraints) is continuously differentiable on some interval $(x^1, x^1 + \tau h)$, $\tau > 0$, the cubic routine can be used successfully [9].)

It should be noted that the cost of attempting to choose a breakpoint is at most one penalty function evaluation.

In general, line searches are dispensed with altogether when in a neighbourhood of a solution, as we expect to take stepsizes of one.

(The reader is referred to Murray and Overton [12] for an alternative piecewise line search.)

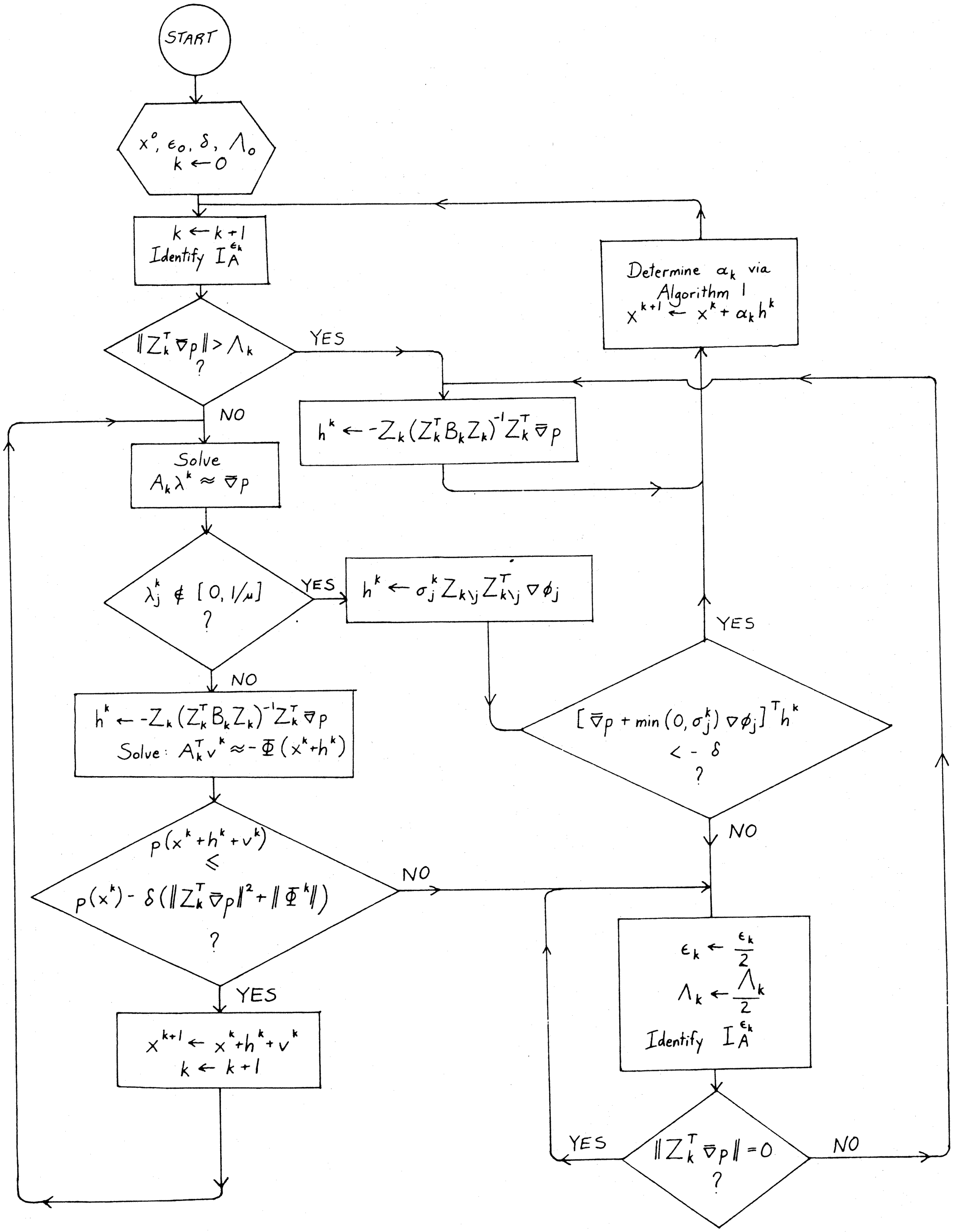
START
$x^0, \epsilon_0, \delta, \Lambda_0$
$k \leftarrow 0$
$k \leftarrow k+1$
Identify $I_A^{\epsilon_k}$
$\|Z_k^T \nabla p\| > \Lambda_k$?
YES
NO
$h^k \leftarrow -Z_k(Z_k^T B_k Z_k)^{-1} Z_k^T \nabla p$
Determine α_k via Algorithm 1
$x^{k+1} \leftarrow x^k + \alpha_k h^k$
Solve $A_k \lambda^k \approx \nabla p$
$\lambda_j^k \notin [0, 1/\mu]$?
YES
$h^k \leftarrow \sigma_j^k Z_{k\setminus j} Z_{k\setminus j}^T \nabla \phi_j$
NO
$h^k \leftarrow -Z_k(Z_k^T B_k Z_k)^{-1} Z_k^T \nabla p$
Solve: $A_k^T v^k \approx -\Phi(x^k + h^k)$
YES
$[\nabla p + \min(0, \sigma_j^k) \nabla \phi_j]^T h^k < -\delta$?
NO
$p(x^k + h^k + v^k) \leqslant p(x^k) - \delta(\|Z_k^T \nabla p\|^2 + \|\Phi^k\|)$?
NO
YES
$x^{k+1} \leftarrow x^k + h^k + v^k$
$k \leftarrow k+1$
$\epsilon_k \leftarrow \frac{\epsilon_k}{2}$
$\Lambda_k \leftarrow \frac{\Lambda_k}{2}$
Identify $I_A^{\epsilon_k}$
YES
$\|Z_k^T \nabla p\| = 0$?
NO

Fig. 1. Flowchart of Algorithm 2.

3.2. Minimizing p (with fixed μ)

In Fig. 1 we present a flowchart of our algorithm designed to minimize the exact penalty function for a fixed $\mu > 0$. The user must supply an initial guess for x^0. In addition, the positive parameters Λ and ϵ are initially arbitrary. The global convergence properties are unaffected by this initial assignment, however, the efficiency of the program can be adversely affected by inappropriate choices. What is, and is not, an appropriate choice is a difficult question to answer a priori: we have not attempted to address this question or the closely related scaling problem in this paper. In the numerical results section we give the testing interval which was used for each parameter on the test problems (most of which are well scaled). The parameter δ is assumed to be sufficiently small and is not modified by Algorithm 2.

In the flow-chart (Fig. 1), as well as in the proofs, we use the following notational rule: if a quantity is sub- or super-scripted, then it may change from iteration to iteration in Algorithm 2—if a quantity is not sub- or super-scripted the quantities remain constant.

The algorithm, as it stands, generates an infinite sequence of points (or, if $x^k = x^*$, it cycles). In practice, Algorithm 2 is terminated when all of the following conditions hold:

(1) $\lambda_j^k \in [0, 1/\mu]$ for all $j \in I_A^\epsilon(x^k)$,

(2) $\|Z_k^{\mathrm{T}}\bar{\nabla} p(x^k)\| \leq \mathrm{TOL}$,

(3) $\|\Phi(x^k)\| \leq \mathrm{TOL}$.

Note that after the 'Newton step' has been successful the ϵ-active set is not re-identified until a 'Newton step' is not taken.

There are a number of possible variants of Algorithm 2. They include:

(1) attempting vertical steps in conjunction with non-unit steps along h^k,

(2) performing line-searches along v^k, or $h^k + v^k$,

(3) including second-order information in the dropping step.

Algorithm 2 then, is a procedure to minimize p for a given and fixed μ. If μ is less than a threshold value then the solution will usually also solve the nonlinear programming problem (1.1). Since our objective is to solve (1.1) we may wish to interrupt Algorithm 2 (and restart with a smaller μ) if it appears μ is too large.

3.3. Algorithm 3 (reducing μ)

In some cases there exists a threshold value for μ, say μ_0, such that if $\mu \leq \mu_0$, local minima of $p(x, \mu)$ are also local solutions to (1.1). This value is unknown a priori (it is a function of the Kuhn–Tucker multipliers [6]), therefore an initial choice of μ may be too large, in which case optima of p may be infeasible or indeed p may be unbounded below. Nevertheless, experience suggests that recognizing when μ is too large (during the p-minimization) is usually not a difficult task and a simple reduction of the form $\mu \leftarrow \mu/10$ seems quite adequate.

In particular, when μ_0 does exist, the number of reductions of μ will be finite and thus the asymptotic convergence rate is dependent only on the convergence rate of Algorithm 2, which is independent of μ.

We must periodically check for 2 possibilities:

(i) $\{x^k\}$ is converging to an infeasible point,

(ii) $\{\phi_j(x^k)\} \to -\infty$ for some j.

Our response to possibility (i) is to reduce μ but continue decreasing p from the current x^k. Possibility (ii) is handled by reducing μ and restarting the procedure from the original starting point. (An alternate strategy to handle (ii) is to first find a feasible point by applying Algorithm 2 to problem (1.1) with $f(x)$ replaced by a constant function. If a feasible point is found, then Algorithm 2 can be restarted with this feasible starting value.) We note that the above strategies are not guaranteed to work however they are reasonable and have been successfully used by the authors.

4. Implementation techniques

Many of the numerical techniques used by Gill and Murray [10] for the linearly constrainted problems are easily adapted to this method for the nonlinearly constrainted problem. We sketch the basic ideas here: the reader is referred to the works of Gill and Murray (see also Murray and Wright [11]) for further details.

4.1. The vertical step and estimating duals

Suppose that A is $n \times t$ with linearly independent columns. Then, there exists an $n \times n$ orthogonal matrix Q such that

$$A = Q \cdot R$$

where

$$R = \begin{pmatrix} \bar{R} \\ 0 \end{pmatrix} \quad \text{and} \quad \bar{R} \text{ is } t \times t \text{ and upper triangular.}$$

Clearly then, the dual estimates can be computed by solving

$$\bar{R}\lambda = \bar{Q}^{\mathrm{T}}\bar{\nabla}p \tag{4.1}$$

where $\bar{Q}$ represents the first t columns of Q. Similarly, the vertical step can be computed by solving

$$\bar{R}^{\mathrm{T}}\bar{v} = -\Phi(x^k + h^k) \tag{4.2}$$

and then setting $v \leftarrow Q\bar{v}$.

Consider a single iteration: $x^k \to x^{k+1}$. In general, at x^k there will be a number of active linear and *non*linear constraints at x^k. At x^{k+1} the linear constraints will remain active and thus the columns of A corresponding to these constraints do not change. The columns of A corresponding to the *non*linear constraints will change entirely (in general) whether the *non*linear constraint set changes or not. Therefore we maintain a partition of A so that linear constraints always precede nonlinear constraints. (It is easy to see that this is always possible.)

The mechanisms for adding and deleting constraints (and thus modifying the QR factors) are essentially those of Gill and Murray [10]—we omit the details here. Linear dependence is detected in the following way: if a vector 'a' is to be 'added' to A but is dependent (numerically) on the current columns (detection of linear dependence follows automatically from the QR updating process) then 'a' is *not* added. From Theorem 1, part 1 it follows that if we are far from a stationary point, then 'a' can be ignored with impunity. If we are in a region in which the dual estimates must be computed, then the existence of dependencies can be troublesome. We follow the perturbution strategy of Bartels et al. [2]: all dependencies are perturbed by a small amount and we then solve this new perturbed problem. (However, ultimately the problem solved is the original problem.)

4.2. The horizontal step

The horizontal step is obtained by solving a positive definite system of the form

$$(Z^{\mathrm{T}}BZ)w = -Z^{\mathrm{T}}\bar{\nabla}p, \tag{4.3}$$

and then setting $h \leftarrow Zw$. Since $Z^{\mathrm{T}}BZ$ is positive definite, the LDL^{T} decomposition exists, and if $LDL^{\mathrm{T}} = Z^{\mathrm{T}}BZ$, then we obtain the solution to (4.3) by solving

$$L\bar{w} = -Z^{\mathrm{T}}\bar{\nabla}p, \tag{4.4}$$

$$L^{\mathrm{T}}w = D^{-1}\bar{w}. \tag{4.5}$$

Note that the matrix Z can easily be obtained from the last $n-t$ columns of Q (Gill and Murray [10]).

As we move from x^k to x^{k+1} there are essentially 2 problems with regard to the projected Hessian factors: modifying the factors to include new information about the change in gradients, and adjusting the factors to reflect changes in Z.

Since the asymptotic convergence rate results obtained so far [7] necessitate that the projected Hessian approximations approach the true projected Hessian, our implementation uses a gradient difference approximation technique in the *final* stages. That is, when we are sufficiently close to x^* so that unit stepsizes are being attempted, then we approximate

$$Z_k^{\mathrm{T}}\Big(\nabla^2 f - \frac{1}{\mu}\sum_{i\in I_v}\nabla^2\phi_i - \sum_{i\in I_A}\lambda_i^k\nabla^2\phi_i\Big)Z_k$$

by the method of gradient differences along the columns of Z_k (Gill and Murray [10]).

When we are not in this neighbourhood of x^* we feel the above procedure is probably unjustifiably expensive and furthermore appears unnecessary. Therefore, in this region we apply the rank-2 updating procedures (suggested by Gill and Murray for linear constraints) in a straightforward way.

5. Convergence results

This section is organized as follows:

5.1. A number of definitions and assumptions are formally stated.

5.2. The results are stated and proved.

5.1. *Definitions and assumptions*

(i) The pseudo-value function of p is defined to be

$$\bar{p}(x,\epsilon)=f(x)-\frac{1}{\mu}\sum_{i\in I_v^\epsilon(x)}\phi_i(x).$$

The *pseudo-gradient* of p is

$$\bar{\nabla}p(x,\epsilon)=\nabla\bar{p}=\nabla f(x)-\frac{1}{\mu}\sum_{i\in I_v^\epsilon(x)}\nabla\phi_i(x).$$

(ii) Let A_k denote an $n\times t_k$ matrix whose columns are the gradients of the ϵ-active constraints. We will always assume A_k to be of full rank $(=t_k)$. The matrix Z_k satisfies $Z_k^{\mathrm{T}}Z_k=I_{(n-t_k)}$, and $A_k^{\mathrm{T}}Z_k=0$. The matrix $Z_{k\setminus j}$ satisfies $Z_{k\setminus j}^{\mathrm{T}}Z_{k\setminus j}=I_{n-t_k+1}$, and $\nabla\phi_i(x^k)^{\mathrm{T}}Z_{k\setminus j}=0$ if $i\in I_A^\epsilon(x^k)-\{j\}$, for some $j\in I_A^\epsilon(x^k)$. Define

$$G_L(x^k)=\nabla^2\bar{p}(x^k)-\sum_{i\in I_A^\epsilon}\lambda_i^k\nabla^2\phi_i(x^k),$$

where λ^k is the least squares solution to $A_k\lambda=\bar{\nabla}p(x^k)$.

(iii) At times we will consider an arbitrary (usually) but fixed point $\bar{x}$. At such a point, $\bar{A}$ will denote a matrix whose columns are the gradients of the *precisely active* constraints. That is, the columns of $\bar{A}$ belong to $\{\nabla\phi_i(\bar{x})\mid i\in I_A^0(\bar{x})\}$. The matrix $\bar{Z}$ satisfies $\bar{A}^{\mathrm{T}}\bar{Z}=0$, $\bar{Z}^{\mathrm{T}}\bar{Z}=I_{n-\bar{t}}$, where $\bar{A}$ is $n\times\bar{t}$ and of full rank. If $\bar{x}$ is a stationary point, then $\bar{\lambda}$ satisfies

$$\nabla f(\bar{x})-\sum_{i\in I_v^0(\bar{x})}\nabla\phi_i(\bar{x})=\bar{A}\bar{\lambda}.$$

(iv) A vector $\bar{x}$ is termed a *stationary point* of p if there exists a vector $\bar{\lambda}$ satisfying

$$\nabla f(\bar{x})-\frac{1}{\mu}\sum_{i\in I_v^0(\bar{x})}\nabla\phi_i(\bar{x})=\sum_{i\in I_A^0(\bar{x})}\bar{\lambda}_i\nabla\phi_i(\bar{x}).$$

If $\bar{\lambda}$ satisfies the above equation and $0 \leq \bar{\lambda} \leq 1/\mu$, then we call $\bar{x}$ a *first-order point* of p. If $0 < \bar{\lambda} < 1/\mu$, then $\bar{x}$ is a *strict first-order* point. If $0 < \bar{\lambda} < 1/\mu$ and

$$y^{\mathrm{T}}\Big(\nabla^2 f(\bar{x}) - \frac{1}{\mu} \sum_{i \in I_v^0(\bar{x})} \nabla^2 \phi_i(\bar{x}) - \sum_{i \in I_A^0(\bar{x})} \bar{\lambda}_i \nabla^2 \phi_i(\bar{x})\Big) y > 0$$

for all y satisfying $\bar{A}^{\mathrm{T}} y = 0$, $y \neq 0$, then we term $\bar{x}$ a *strict second-order point* of p. [The reader is referred to Coleman and Conn [6] for justification of this terminology.]

(v) We make the following line-search assumption: If h^k is a descent direction for p at x^k, then α_k is determined so that

$$p(x^k) - p(x^k + \alpha_k h^k) \geq \gamma_1((h^k)^{\mathrm{T}} g^k)^2, \quad \gamma_1 > 0,$$

where

$$g^k = \bar{\nabla} p(x^k) - \frac{1}{\mu} \sum_{i \in \tilde{I}_v^{\epsilon}(x^k)} \nabla \phi_i(x^k)$$

and

$$\tilde{I}_v^{\epsilon}(x^k) = \{i \in I_A^{\epsilon}(x^k) \mid \nabla \phi_i^{\mathrm{T}} h^k < 0\}.$$

(It is a direct extension of a result of Conn and Pietryzkowski (Proposition 1, [9]) to show that the above condition be satisfied.)

(vi) Let W denote a compact set, where $\{x^k\} \in W$.

(vii) Let S, $\bar{S}$, $\hat{S}$ denote the set of stationary points of p in W, the set of stationary but *not* first-order points and the set of first-order points respectively. Note that $S = \bar{S} \cup \hat{S}$, $\bar{S} \cap \hat{S} = \emptyset$.

Lemma 1. *We assume that*

(i) *f, ϕ_i, $i = 1, \ldots, m$ are twice continuously differentiable on a compact set W, where $\{x^k\} \in W$,*

(ii) *$\bar{x}$ is any strict second-order point of p in W,*

(iii) *the gradients of the active constraints (at $\bar{x}$) are linearly independent on W,*

(iv) *$I_A^{\epsilon_k}(x^k) = I_A^0(\bar{x})$, for all k,*

(v) *there exist positive constants b_1, b_2 such that for all vectors $w \in \mathbf{R}^{n-t_k}$, $w \neq 0$,*

$$b_1 \|w\|^2 \leq w^{\mathrm{T}}(Z_k^{\mathrm{T}} B_k Z_k) w \leq b_2 \|w\|^2.$$

Then there exist positive constants Δ_1, Δ_2, and δ such that

(1) $\|x^k - \bar{x}\| \leq \Delta_1$,

(2) $\|Z_k^{\mathrm{T}} B_k Z_k - Z_k^{\mathrm{T}} G_L(x^k) Z_k\| \leq \Delta_2$

implies that

$$p(x^k + h^k + v^k) - p(x^k) \leq -\delta(\|Z_k^{\mathrm{T}} \bar{\nabla} p(x^k)\|_2^2 + \|\Phi(x^k)\|_1),$$

where

$$h^k = -Z_k(Z_k^{\mathrm{T}}B_kZ_k)^{-1}Z_k^{\mathrm{T}}\bar{\nabla}p(x^k),$$
$$v^k = -A_k(A_k^{\mathrm{T}}A_k)^{-1}\Phi(x^k + h^k).$$

Proof. (To simplify notation, we drop the 'k' superscripts and subscripts. In addition, I_A and I_v denote $I_A^0(\bar{x})$, and $I_v^0(\bar{x})$ respectively and we assume that $I_A = \{1, \ldots, t\}$. Finally, if an argument is unspecified, it is assumed to be x.)

(a) *Changes in* $\bar{p}$. Recall

$$\bar{p}(x) = f(x) - \frac{1}{\mu}\sum_{i\in I_v} \phi_i(x).$$

By Taylor's theorem,

$$\bar{p}(x+h+v) = \bar{p}(x) + \nabla\bar{p}^{\mathrm{T}}[h+v] + \tfrac{1}{2}[h+v]^{\mathrm{T}}\nabla^2\bar{p}[h+v] + \mathrm{o}(\|h+v\|^2). \tag{5.1}$$

Recall

$$G_L(x) = \nabla^2\bar{p}(x) - \sum_{i=1}^{t} \lambda_i\nabla^2\phi_i(x).$$

Thus

$$h = -Z(Z^{\mathrm{T}}G_LZ)^{-1}Z^{\mathrm{T}}\nabla\bar{p} + ZEZ^{\mathrm{T}}\nabla\bar{p}, \tag{5.2}$$

where

$$E = (Z^{\mathrm{T}}G_LZ)^{-1} - (Z^{\mathrm{T}}BZ)^{-1}.$$

In addition,

$$\nabla\bar{p} = A\lambda + Zw, \quad \text{for some } w \in \mathbf{R}^{n-t}. \tag{5.3}$$

Hence, by (5.1), (5.2), and (5.3),

$$\begin{aligned}\bar{p}(x+h+v) - \bar{p}(x) = {} & -\nabla\bar{p}^{\mathrm{T}}Z(Z^{\mathrm{T}}G_LZ)^{-1}Z^{\mathrm{T}}\nabla\bar{p} + \lambda^{\mathrm{T}}A^{\mathrm{T}}v \\ & + \tfrac{1}{2}h^{\mathrm{T}}\nabla^2\bar{p}h + \tfrac{1}{2}v^{\mathrm{T}}\nabla^2\bar{p}v + h^{\mathrm{T}}\nabla^2\bar{p}v \\ & + \mathrm{o}(\|h+v\|^2) + \nabla\bar{p}Z^{\mathrm{T}}EZ^{\mathrm{T}}\nabla\bar{p}.\end{aligned} \tag{5.4}$$

But

$$v = -A(A^{\mathrm{T}}A)^{-1}\Phi(x+h), \tag{5.5}$$

where $\Phi = (\phi_1, \ldots, \phi_t)$. Since $\nabla\phi_i^{\mathrm{T}}h = 0$, $i \in I_A$, it follows that

$$\phi_i(x+h) = \phi_i(x) + \tfrac{1}{2}h^{\mathrm{T}}\nabla^2\phi_ih + \mathrm{o}(\|h\|^2). \tag{5.6}$$

Therefore, by (5.5) and (5.6),

$$\lambda^{\mathrm{T}}A^{\mathrm{T}}v = -\sum_{i=1}^{t} \lambda_i(\phi_i + \tfrac{1}{2}h^{\mathrm{T}}\nabla^2\phi_ih) + \mathrm{o}(\|h\|^2).$$

Hence (5.4) can be written

$$\begin{aligned}\bar{p}(x+h+v)-\bar{p}(x) = &-\nabla\bar{p}^{\mathrm{T}}(Z^{\mathrm{T}}G_LZ)^{-1}Z^{\mathrm{T}}\nabla\bar{p}+\tfrac{1}{2}h^{\mathrm{T}}\Big(\nabla^2\bar{p}-\sum_{i=1}^{t}\lambda_i\nabla^2\phi_i\Big)h\\ &-\sum_{i=1}^{t}\lambda_i\phi_i+\tfrac{1}{2}v^{\mathrm{T}}\nabla^2\bar{p}v+h^{\mathrm{T}}\nabla^2\bar{p}v\\ &+\mathrm{o}(\|h+v\|^2)+\mathrm{o}(\|h\|^2)+\nabla\bar{p}^{\mathrm{T}}ZEZ^{\mathrm{T}}\nabla\bar{p}. \qquad (5.7)\end{aligned}$$

But, by the definition of G_L,

$$\begin{aligned}h^{\mathrm{T}}\Big(\nabla^2\bar{p}-\sum_{i=1}^{t}\lambda_i\nabla^2\phi_i\Big)h &= h^{\mathrm{T}}G_Lh\\ &=\nabla\bar{p}^{\mathrm{T}}Z(Z^{\mathrm{T}}G_LZ)^{-1}Z^{\mathrm{T}}\nabla\bar{p}-2\nabla\bar{p}^{\mathrm{T}}ZEZ^{\mathrm{T}}\nabla\bar{p}\\ &\quad+\nabla\bar{p}^{\mathrm{T}}ZEZ^{\mathrm{T}}G_LZEZ^{\mathrm{T}}\nabla\bar{p}\end{aligned}$$

using (5.2). Substituting this expression into (5.7) results in

$$\begin{aligned}\bar{p}(x+h+v)-\bar{p}(x) = &-\tfrac{1}{2}\nabla\bar{p}^{\mathrm{T}}Z(Z^{\mathrm{T}}G_LZ)^{-1}Z^{\mathrm{T}}\nabla\bar{p}\\ &-\sum_{i=1}^{t}\lambda_i\phi_i+\tfrac{1}{2}v^{\mathrm{T}}\nabla^2\bar{p}v+h^{\mathrm{T}}\nabla^2\bar{p}v.\\ &+e(x), \qquad (5.8)\end{aligned}$$

where

$$\begin{aligned}e(x)=\mathrm{o}(\|h+v\|^2)+\mathrm{o}(\|h\|^2)-\nabla\bar{p}^{\mathrm{T}}ZEZ^{\mathrm{T}}\nabla\bar{p}\\ +\nabla\bar{p}^{\mathrm{T}}ZEZ^{\mathrm{T}}G_LZEZ^{\mathrm{T}}\nabla\bar{p}.\end{aligned}$$

(b) *Changes in* ϕ_i, $i=1,\dots,t$. Using Taylor's theorem, it is straightforward to verify that

$$\begin{aligned}&\sum_{i=1}^{t}\min(0,\phi_i(x))-\sum_{i=1}^{t}\min(0,\phi_i(x+h+v))\le\\ &\quad\le\eta^{\mathrm{T}}\Phi+\frac{1}{2}\sum_{i=1}^{t}|v^{\mathrm{T}}\nabla^2\phi_iv|-\sum_{i=1}^{t}|h^{\mathrm{T}}\nabla^2\phi_iv|+\mathrm{o}(\|h+v\|^2), \qquad (5.9)\end{aligned}$$

where $\eta_i=1$ if $\phi_i<0$, otherwise $\eta_i=0$.

(c) *Changes in p*. We now combine parts (a) and (b) to obtain

$$\begin{aligned}p(x+h+v)-p(x)\le &-\tfrac{1}{2}\nabla\bar{p}^{\mathrm{T}}Z(Z^{\mathrm{T}}G_LZ)^{-1}Z^{\mathrm{T}}\nabla\bar{p}+\Big(\frac{\eta}{\mu}-\lambda\Big)^{\mathrm{T}}\Phi\\ &+\tfrac{1}{2}|v^{\mathrm{T}}\nabla^2\bar{p}v|+|h^{\mathrm{T}}\nabla^2\bar{p}v|+\frac{1}{2\mu}\sum_{i=1}^{t}|v^{\mathrm{T}}\nabla^2\phi_iv|\\ &+\frac{1}{\mu}\sum_{i=1}^{t}|h^{\mathrm{T}}\nabla^2\phi_iv|+e(x). \qquad (5.10)\end{aligned}$$

But

$$\begin{aligned}v=&-A(A^{\mathrm{T}}A)^{-1}\Phi(x+h)\\ &-A(A^{\mathrm{T}}A)^{-1}[\Phi+\tfrac{1}{2}r]+\mathrm{o}(\|h\|^2), \qquad (5.11)\end{aligned}$$

where $r = (h^T\nabla^2\phi_1 h, \dots, h^T\nabla^2\phi_t h)^T$. Define

$$H_1 = \nabla^2\bar{p}A(A^TA)^{-1}, \qquad H_2 = (A^TA)^{-1}A^TH_1,$$

$$\left.\begin{aligned} \bar{H}_i &= \nabla^2\phi_i A(A^TA)^{-1} \\ \tilde{H}_i &= (A^TA)^{-1}A^T\bar{H}_i \end{aligned}\right\} \quad i = 1, \dots, t.$$

Thus

$$v^T\nabla^2\bar{p}v = \Phi(x)^Tw(x) + \tfrac{1}{4}r^TH_2r + o(\|h\|^2), \tag{5.12}$$

where $w(x) = H_2[\Phi + r(x)]$. But $r^TH_2r = O(\|h^4\|)$, and hence by (5.12)

$$\tfrac{1}{2}|v^T\nabla^2\bar{p}v| \le \tfrac{1}{2}\sum_{i=1}^{t} |\phi_i(x)| \cdot |w_i(x)| + o(\|h\|^2). \tag{5.13}$$

Similarly, we can define vectors

$$u^j(x) = \tilde{H}_j[\Phi(x) + r(x)], \quad j = 1, \dots, t.$$

If we let $\bar{u}_i = \sum_{j=1}^{t} |u_i^j(x)|$, for $i = 1, \dots, t$ we obtain

$$\tfrac{1}{2}\sum_{i=1}^{t} |v^T\nabla^2\phi_i v| \le \tfrac{1}{2}\sum_{i=1}^{t} |\phi_i(x)| \cdot |\bar{u}_i(x)| + o(\|h\|^2). \tag{5.14}$$

Consider now the terms $h^T\nabla^2\bar{p}v$, $h^T\nabla^2\phi_i$, $i = 1, \dots, t$. Following lines similar to that used above, it is straightforward to show

$$|h^T\nabla^2\bar{p}v| \le \sum_{i=1}^{t} |y_i(x)| \cdot |\phi_i(x)| + o(\|h\|^2) \tag{5.15}$$

and

$$\sum_{i=1}^{t} |v^T\nabla^2\phi_i h| \le \sum_{i=1}^{t} |\phi_i(x)| \cdot |\bar{s}_j(x)| + o(\|h\|^2), \tag{5.16}$$

where $y^T = h^TH_1$, $s^j(x) = h^T\bar{H}_j$, and $\bar{s}_j(x) = \sum_{i=1}^{t} |s_i^j(x)|$. Define

$$c_i(x) = \tfrac{1}{2}|w_i(x)| + \frac{1}{2\mu}|\bar{u}_i(x)| + \frac{1}{\mu}|\bar{s}_i(x)| + |y_i(x)|.$$

By (5.10), (5.13)–(5.16) (and noting that, for Δ_1 sufficiently small, $((\eta_i/\mu) - \lambda_i)\phi_i = -|\lambda_i - (\eta_i/\mu)| \cdot |\phi_i|$)

$$\begin{aligned} p(x + h + v) - p(x) \le{}& -\tfrac{1}{2}\nabla\bar{p}^TZ(Z^TG_LZ)^{-1}Z^T\nabla\bar{p} \\ &+ \sum_{i=1}^{t}\left(-\left|\lambda_i - \frac{\eta_i}{\mu}\right| + c_i(x)\right)|\phi_i(x)| + e(x). \end{aligned} \tag{5.17}$$

By assumption (ii), for Δ_1 sufficiently small there exists a positive constant $\bar{b}_2$ such that

$$\nabla\bar{p}^TZ(Z^TG_LZ)^{-1}Z^T\nabla\bar{p} \ge \frac{2}{\bar{b}_2}\|Z^T\nabla\bar{p}\|^2. \tag{5.18}$$

Moreover $\lambda_i - (\eta_i/\mu) \neq 0$, $i = 1, \dots, t$ and $c_i(x) \to 0$ as $x \to \bar{x}$ (by assumption (iv) and continuity of c_i). Therefore for Δ_1 sufficiently small, and for some $\hat{\delta} > 0$,

$$p(x+h+v) - p(x) \leq -\hat{\delta}(\|Z^{\mathrm{T}}\nabla\bar{p}\|_2^2 + \|\Phi\|_1) + e(x). \tag{5.19}$$

Consider,

$$\begin{aligned} e(x) &= \mathrm{o}(\|h+v\|^2) + \mathrm{o}(\|h\|^2) - \nabla\bar{p}^{\mathrm{T}} ZEZ^{\mathrm{T}}\nabla\bar{p} \\ &\quad + \nabla\bar{p}^{\mathrm{T}} ZEZ^{\mathrm{T}} G_L ZEZ^{\mathrm{T}}\nabla\bar{p}. \end{aligned}$$

But

$$\begin{aligned} \|h+v\|_2^2 &= \|h\|_2^2 + \|v\|_2^2 \\ &\leq L_1\|Z^{\mathrm{T}}\nabla\bar{p}\|_2^2 + L_2\|\Phi\|_2^2, \end{aligned}$$

$L_1, L_2 > 0$. (Since $A(x)$, $(Z^{\mathrm{T}}BZ)^{-1}$, and Z are bounded on W.) Therefore by (2), for Δ_1, Δ_2 sufficiently small,

$$e(x) \leq \tfrac{1}{2}\hat{\delta}(\|Z^{\mathrm{T}}\nabla\bar{p}\|_2^2 + \|\Phi\|_1). \tag{5.20}$$

Hence, if $\delta = \frac{1}{2}\hat{\delta}$, by (5.19), (5.20),

$$p(x+h+v) - p(x) \leq -\delta(\|Z^{\mathrm{T}}\nabla\bar{p}\|_2^2 + \|\Phi\|_1).$$

Theorem 1. *We assume that*

(1) *the functions f, ϕ_i, $1 = 1, \dots, m$ are twice continuously differentiable,*

(2) *$\{x^k\}$ is generated by Algorithm 2 starting from an arbitrary initial point, and $\{x^k\} \in W$, W is compact,*

(3) *the number of stationary points of p, in W, is finite,*

(4) *all first-order points of p in W (recall these are denoted $\hat{S}$) are strict second-order points of p,*

(5) *the vectors $\nabla\phi_i(x^k)$, $i \in I_A^{\epsilon_k}(x^k)$ are linearly independent,*

(6) *the line-search condition is satisfied by Algorithm 1,*

(7) *if $\{x^{k_i}\}$ is a subsequence and $\bar{x}$ is a first-order point such that*

$$x^{k_i} \to \bar{x}, \quad \textit{and} \quad I_A^{\epsilon_{k_i}}(x^{k_i}) = I_A^0(\bar{x}),$$

then second-order information is approximated so that

$$Z_{k_i}^{\mathrm{T}} B_{k_i} Z_{k_i} \to \bar{Z}^{\mathrm{T}}\Big(\nabla^2 f(\bar{x}) - \frac{1}{\mu}\sum_{i \in I_v^0(\bar{x})} \nabla^2\phi_i(\bar{x}) - \sum_{i \in I_A^0(\bar{x}_i)} \bar{\lambda}_i \nabla^2\phi_i(\bar{x}_i)\Big)\bar{Z}_i.$$

Then, for all δ sufficiently small,

(1) $\epsilon_k \not\to 0$,

(2) $x^k \to \bar{x} \in \hat{S}$,

(3) *for k sufficiently large, the 'Newton step' is executed.*

Proof. *Part* 1 [If $Z_k^{\mathrm{T}}\bar{\nabla}p(x^k) \neq 0$, then $-Z_k(Z_k^{\mathrm{T}}B_kZ_k)^{-1}Z_k^{\mathrm{T}}\bar{\nabla}p(x^k)$ $(= h^k)$ is a descent direction for p, at x^k.]

$$\text{Since } \nabla\phi_i^{\mathrm{T}} h^k = 0 \quad \text{for all } i \in I_A^{\epsilon}(x^k) \tag{5.21}$$

it follows, by a Taylor expansion,

$$\phi_i(x^k + \alpha h^k) = \phi_i(x^k) + \mathrm{O}(\alpha^2), \quad \text{for all } i \in I_A^{\epsilon}(x^k). \tag{5.22}$$

Therefore,

$$p(x^k + \alpha h^k) = p(x^k) + \alpha\bar{\nabla} p(x^k)^{\mathrm{T}} h^k + \mathrm{O}(\alpha^2). \tag{5.23}$$

Hence for all α sufficiently small,

$$p(x^k + \alpha h^k) < p(x^k). \tag{5.24}$$

Part 2 [If $\lambda_j^k \notin [0, 1/\mu]$ it follows that $h^k = \sigma_j^k Z_{k\setminus j} Z_{k\setminus j}^{\mathrm{T}} \nabla\phi_j(x^k)$ is a descent direction for p at x^k, where $\sigma_j^k = -\operatorname{sign}(\lambda_j^k)$.] First we note that

$$\nabla\phi_i(x^k)^{\mathrm{T}} h^k = 0 \quad \text{for all } i \in I_A^{\epsilon}(x^k) - \{j\}, \tag{5.25}$$

and therefore

$$\phi_i(x^k + \alpha h^k) = \phi_i(x^k) + \mathrm{O}(\alpha^2), \quad \text{for } i \in I_A^{\epsilon}(x^k) - \{j\}. \tag{5.26}$$

Considering (5.25), and (5.26)

$$p(x^k + \alpha h^k) \le p(x^k) + \alpha g(x^k)^{\mathrm{T}} h^k + \mathrm{O}(\alpha^2), \tag{5.27}$$

where

$$g(x^k) = \bar{\nabla} p(x^k) + \frac{1}{\mu}\min(0, \sigma_j^k)\nabla\phi_j(x^k). \tag{5.28}$$

Case 1: Suppose $\lambda_j^k < 0$. Then $\sigma_j = 1$ and

$$\begin{aligned} g(x^k)^{\mathrm{T}} h^k &= \bar{\nabla} p(x^k)^{\mathrm{T}} h^k && \text{(using (5.28))} \\ &= (A_k \lambda^k)^{\mathrm{T}} h^k && \text{(since } \bar{\nabla} p = A_k\lambda^k + u,\ u \in \mathbf{N}(A_k^{\mathrm{T}}) \\ &&& \text{where } \mathbf{N}(A) \text{ denotes the nullspace of } A.) \\ &= \lambda_j^k \nabla\phi_j(x^k)^{\mathrm{T}} h^k && \text{(by (5.25))} \\ &= \lambda_j^k \|Z_{k\setminus j}^{\mathrm{T}} \nabla\phi_j(x^k)\|^2 < 0. \end{aligned}$$

Case 2. Suppose $\lambda_j^k > 1/\mu$. Then $\sigma_j^k = -1$ and

$$\begin{aligned} g(x^k)^{\mathrm{T}} h^k &= (\bar{\nabla} p(x^k) - 1/\mu \nabla\phi_j(x^k))^{\mathrm{T}} h^k \\ &= (\lambda_j^k - 1/\mu)\nabla\phi_j^{k\mathrm{T}} h^k \\ &= (1/\mu - \lambda_j^k)\|Z_{k\setminus j}^{\mathrm{T}} \nabla\phi_j\|^2 < 0. \end{aligned}$$

Therefore, h^k is a descent direction for p at x^k.

Part 3 [There exist positive scalars Δ, δ such that $\|x^k - \bar{x}\| \le \Delta$, $\bar{x} \in \bar{S}$, and $I_A^{\epsilon}(x^k) = I_A^0(\bar{x})$ implies that

(i) $\lambda_j^k \notin [0, 1/\mu]$ for some $j \in I_A^0(\bar{x})$, and

(ii) $(h^k)^{\mathrm{T}} g^k < -\delta$, where

$$h^k = \sigma_j^k Z_{k\setminus j} Z_{k\setminus j}^{\mathrm{T}} \nabla\phi_j(x^k),$$

$$g^k = \bar{\nabla} p(x^k) + \frac{1}{\mu}\min(0, \sigma_j^k)\nabla\phi_j(x^k).]$$

Since $\bar{x} \in \bar{S}$ it follows that for some j,

$$\bar{\lambda} \notin [0, 1/\mu] \quad \text{where } \bar{A}\bar{\lambda} = \nabla f(\bar{x}) - \frac{1}{\mu}\sum_{i\in I_{\mathrm{r}}^0(\bar{x})} \nabla\phi_i(\bar{x}). \tag{5.29}$$

Recall from Section 5.1, part (iii) that quantities are 'barred' to indicate that they are evaluated at $\bar{x}$.

Let $\tilde{Z}$ be a matrix with $\bar{t}+1$ orthonormal columns which satisfies $\nabla\phi_i^{\mathrm{T}}\tilde{Z} = 0$, for all i in $I_A^0(\bar{x}) - \{j\}$. Define

$$\bar{h} = \bar{\sigma}_j \tilde{Z}\tilde{Z}^{\mathrm{T}} \nabla\phi_j(\bar{x}),$$

and

$$\bar{g} = \nabla f(\bar{x}) - \frac{1}{\mu}\Big\{\sum_{i\in I_{\mathrm{r}}^0(\bar{x})} \nabla\phi_i(\bar{x}) - \min(0, \bar{\sigma}_j)\nabla\phi_j(\bar{x})\Big\},$$

where $\bar{\sigma}_j = -\operatorname{sgn}(\bar{\lambda}_j)$.

Case 1: Suppose $\bar{\lambda}_j < 0$. Then $\bar{\sigma}_j = 1$, and

$$\bar{g} = \nabla f(\bar{x}) - \frac{1}{\mu}\sum_{i\in I_{\mathrm{r}}^0(\bar{x})} \nabla\phi_i(\bar{x}).$$

Since $\nabla\phi_i^{\mathrm{T}}\bar{h} = 0$ for $i \in I_A^0(\bar{x}) - \{j\}$, and considering (5.29) and the definition of $\bar{g}$,

$$-\bar{h}^{\mathrm{T}}\bar{g} = -\bar{\lambda}_j \nabla\phi_j(\bar{x})^{\mathrm{T}}\bar{h} = -\bar{\lambda}_j\|\tilde{Z}^{\mathrm{T}}\nabla\phi_j\|^2 =: \bar{\delta}_1,$$

Considering the linear independence assumption, $\bar{\delta}_1 > 0$.

Case 2: Suppose $\bar{\lambda}_j > 1/\mu$. Then $\bar{\sigma}_j = -1$ and

$$\bar{g} = \nabla f(\bar{x}) - \frac{1}{\mu}\sum_{i\in I_{\mathrm{r}}^0(\bar{x})} \nabla\phi_i(\bar{x}) - \frac{1}{\mu}\nabla\phi_j(\bar{x}).$$

Since $\nabla\phi_i^{\mathrm{T}}\bar{h} = 0$ for $i \in I_A^0(\bar{x}) - \{j\}$, and considering (5.25) and the definition of $\bar{g}$,

$$\begin{aligned} -\bar{h}^{\mathrm{T}}\bar{g} &= (1/\mu - \bar{\lambda}_j)\nabla\phi_j(\bar{x})^{\mathrm{T}}\bar{h} \\ &= (\bar{\lambda}_j - 1/\mu)\|\tilde{Z}^{\mathrm{T}}\nabla\phi_j\|^2 =: \bar{\delta}^2. \end{aligned}$$

Considering the linear independence assumption, $\bar{\delta}_2 > 0$.

Since $|\bar{S}|$ is finite, it is clear that there exists a positive scalar $\bar{\delta}$ such that if $\bar{x}$ is any member of $\bar{S}$, then $-\bar{h}^{\mathrm{T}}g \geq \bar{\delta}$. By continuity it follows that if $\|x^k - \bar{x}\| \leq \Delta$, $\bar{x} \in \bar{s}$, *and* $I_A^{\epsilon_k}(x^k) = I_A^0(\bar{x})$, then

(i) $\lambda_j^k \notin [0, 1/\mu]$,

(ii) $(-h^k)^{\mathrm{T}} g^k \geq \frac{1}{2}\bar{\delta} =: \delta$.

Part 4 [For all k sufficiently large, the 'dropping step' is not executed.]

Rather than consider a subsequence let us assume (without loss of generality) that the 'dropping step' is executed for all k. Then, for some $j \in I_A^{\epsilon_k}(x^k)$,

$$\lambda_j^k \notin [0, 1/\mu] \quad \text{and} \quad h^k = \sigma_j^k Z_{k\setminus j} Z_{k\setminus j}^{\mathrm{T}} \nabla \phi_j(x^k).$$

But Algorithm 2 requires that $-g(x^k)^{\mathrm{T}} h^k > \delta$, where $g(x^k) = \bar{\nabla} p + (1/\mu) \min(0, \sigma_j^k) \nabla \phi_j$, and therefore, by the line search condition,

$$p(x^k) - p(x^k + \alpha_k h^k) \geq \gamma_1 \delta^2. \tag{5.30}$$

But (5.30) implies that $p(x^k) \to -\infty$, which is contradictory.

Part 5 [If $x^{k_i} \to \bar{x}$ then for k_i sufficiently large, $I_A^{\epsilon_{k_i}}(x^{k_i}) \subseteq I_A^0(\bar{x})$.] Suppose that $I_A^{\epsilon_{k_i}}(x^{k_i}) \supset I_A^0(\bar{x})$. We can assume (without loss of generality) that $I_A^{\epsilon_{k_i}}(x^{k_i}) = I_A^{\epsilon_{k_i+1}}(x^{k_i+1})$, $i = 1, \ldots, \infty$.

(i) Suppose $Z_{k_i}^{\mathrm{T}} \bar{\nabla} p(x^{k_i}) \not\to 0$. Then we can assume that $h^{k_i} \to \bar{h} \neq 0$. Let I_A denote $I_A^{\epsilon_{k_i}}(x^{k_i})$ and I_v denote $I_v^{\epsilon_{k_i}}(x^{k_i})$. Clearly $-\bar{g}^{\mathrm{T}} \bar{h} > 0$, where

$$\bar{g} = \nabla f(\bar{x}) - \frac{1}{\mu} \sum_{i \in I_v} \nabla \phi_i(\bar{x}).$$

Let $\bar{\beta} = -\bar{g}^{\mathrm{T}} \bar{h}$. By continuity, $g^{k_i} \to \bar{g}$ where

$$g_{k_i} = \nabla f(x^{k_i}) - \frac{1}{\mu} \sum_{i \in I_v} \nabla \phi_i(x^{k_i}).$$

Hence for k_i sufficiently large $-(h^{k_i})^{\mathrm{T}} g^{k_i} \geq \frac{1}{2}\bar{\beta}$. By Part 4, h^{k_i} is a descent direction for p and therefore, applying the line search condition,

$$p(x^k) - p(x^k + \alpha_k h^k) \geq \delta_1 (\bar{\beta}/2)^2.$$

It follows that $p(x^k) \to -\infty$, a contradiction.

(ii) Suppose $Z_{k_i}^{\mathrm{T}} \bar{\nabla} p(x^{k_i}) \to 0$. Since $I_A^{\epsilon_{k_i}} \not\subseteq I_A^0(\bar{x})$, it follows that for some j, $c_j(x^{k_i}) \not\to 0$. A consequence of this is that $\Lambda_k \to 0$. (Suppose $\Lambda_k \not\to 0$. By part 4, the Newton step is attempted for all k sufficiently large. Since p is bounded below and $c_j(x^{k_i}) \not\to 0$ it follows that for k sufficiently large the Newton step is *unsuccessfully* attempted. This implies $\Lambda_k \to 0$ and $\epsilon_k \to 0$.) But $\epsilon_k \to 0$, and $x^{k_i} \to \bar{x}$ implies $I_A^{\epsilon_{k_i}}(x^{k_i}) \subseteq I_A^0(\bar{x})$.

Part 6 [The 'Newton-step' is successful for all k sufficiently large, $\epsilon_k \not\to 0$, and $x^k \to \bar{x} \in \hat{S}$.]

(i) Suppose $\epsilon_k \to 0$. Then $\Lambda_k \to 0$ and therefore $Z_{k_i}^{\mathrm{T}} \bar{\nabla} p(x^{k_i}) \to 0$ for some subsequence $\{x^{k_i}\}$. (Suppose $Z_{k_i}^{\mathrm{T}} \bar{\nabla} p(x^{k_i}) \not\to 0$, for any subsequence $\{x^{k_i}\}$. Then, for all k sufficiently large $Z_k^{\mathrm{T}} \bar{\nabla} p(x^k) > \Lambda_k$. By Algorithm 2, it follows that Λ_k is not reduced, for all k sufficiently large, and thus $\Lambda_k \not\to 0$.) Therefore we have a convergent subsequence $\{x^{k_i}\} \to \bar{x}$ and $Z_{k_i}^{\mathrm{T}} \bar{\nabla} p(x^{k_i}) \to 0$. By Part 5, for k sufficiently large, $I_A^{\epsilon_{k_i}}(x^{k_i}) \subseteq I_A^0(\bar{x})$. However, the linear independence assumption and $Z_{k_i}^{\mathrm{T}} \bar{\nabla} p(x^{k_i}) \to 0$, forces $I_A^{\epsilon_{k_i}}(x^{k_i}) = I_A^0(\bar{x})$. Considering Parts 3 and 4, it follows that for at least one subsequence $\{x^{k_i}\}$, $x^{k_i} \to \bar{x} \in \hat{S}$, and $I_A^{\epsilon_{k_i}}(x^{k_i}) = I_A^0(\bar{x})$. By Lemma 1, for k_i sufficiently large, iterations k_i+1, $k_i+2, \ldots$ will be 'Newton-steps'. It follows that $\epsilon_k \not\to 0$.

(ii) Suppose $\epsilon_k \not\to 0$. Thus $\epsilon_k = \epsilon > 0$, for k sufficiently large. By Part 5 it follows that there exists a subsequence $\{x^{k_i}\}$ such that $x^{k_i} \to \bar{x}$, and $I^{\epsilon}_A(x^{k_i}) = I^0_A(\bar{x})$. Using an argument identical to that used in Part 5(i), we can establish that $\bar{x} \in S$. By Parts 3 and 4, $\bar{x} \in \hat{S}$. By Lemma 1 and the boundedness of p, for k_i sufficiently large, iterations $k := k_i+1$, $k_i+2, \ldots$ are Newton steps and $x^k \to \bar{x} \in \hat{S}$.

6. Numerical results

We present here a brief summary of numerical results obtained on a number of test problems. We feel that these results indicate that the proposed algorithm has a promising future.

In Table 1 is a chart of the best and worst results achieved by our method for various initial parameter choices in the indicated parameter choices in the indicated range. We list our starting point in every case. The 'accuracy' column indicates the number of significant digits achieved (x-values)

Table 1

ϵ : 1–5.
μ : 0.001–1.
TOL : 10^{-8}
δ : 10^{-6}
Λ : 0.001–0.1

Problem	Initial point	accuracy	Penalty function evaluations	
			(Best)	(Worst)
Rosen-Suzuki [14]	$(0, 0, 0, 0)$	8	19	37
Wong in [1]	$(3, 3, 0, 4, 1, 3, 0)$	5	69	75
	$(1, 2, 0, 4, 1, 1)$	5	50	64
Powell [13]	$(-2, 2, 2, -1, -1)$	7	5	15
	$(-1.5, 1.5, 2, -1, -1)$	7	5	15
Colville 1 [8]	$(0, 0, 0, 0, 1)$	4	11	11
†Chamberlain 1 [3]	$(0, 0)$	8	6	6
†Chamberlain 2	$(1, 5)$	8	4	4
Colville 3 [8]	$(78, 62, 33, 44, 31.07, 44.18, 35.22)$	5	14	19
	$(78, 33, 27, 27, 27)$	5	6	18
Colville 2 [8]	$x_i = 0.001, i \neq 7, x_7 = 60$	6	87	179

† These problems were designed by Chamberlain [3] to demonstrate cycling behaviour of a successive quadratic programming algorithm. We note that our method converges rapidly.

7. Concluding remarks

The numerical and theoretical results given here (see also [7]) suggest that the proposed algorithm is an efficient and reliable way to solve the well-scaled nonlinear programming problem. The method possesses both global and superlinear convergence properties: it is not necessary that the full ($n \times n$) Lagrangian Hessian be positive definite at the solution. The method is computational efficient: it is not required that the projected Hessian be approximated by the expensive gradient difference method except in the final stages of convergence.

Further developments are expected to include a full projected Quasi-Newton implementation, and scaling considerations.

References

[1] J. Asaadi, "A computational comparison of some nonlinear programs", *Mathematical Programming* 4 (1973) 144–154.

[2] R.H. Bartels, A.R. Conn and J.W. Sinclair, "Minimization techniques for piecewise differentiable functions: The l_1 solution to an overdetermined linear system", *SIAM Journal on Numerical Analysis* 15 (1978) 224–241.

[3] R.M. Chamberlain, "Some examples of cycling in variable metric methods for constrained optimization", *Mathematical Programming* 16 (1979) 378–383.

[4] C. Charalambous, "A lower bound for the controlling parameter of the exact penalty function", *Mathematical Programming* 15 (1978) 278–290.

[5] C. Charalambous, "On the conditions for optimality for the nonlinear l_1 problem", *Mathematical Programming* 17 (1979) 123–135.

[6] T.F. Coleman and A.R. Conn, "Second-order conditions for an exact penalty function", *Mathematical Programming* 19 (1980) 178–185.

[7] T.F. Coleman and A.R. Conn, "Nonlinear programming via an exact penalty function: Asymptotic analysis", *Mathematical Programming* 24 (1982) 123–136. [This issue.].

[8] A.R. Colville, "A comparative study of non-linear programming codes", IBM New York Scientific Center Report 320-2949 (1968).

[9] A.R. Conn and T. Pietrzykowski, "A penalty function method converging directly to a constrained optimum", *SIAM Journal on Numerical Analysis* 14 (1977) 348–375.

[10] P. Gill and W. Murray, *Numerical methods for constrained optimization* (Academic Press, London, 1974).

[11] W. Murray and M. Wright, "Projected Lagrangian methods based on the trajectories of penalty and barrier functions", Technical Report SOL 78-23 Department of Operations Research, Stanford University (Stanford, CA, 1978).

[12] W. Murray and M. Overton, "Steplength algorithms for minimizing a class of non-differentiable functions", *Computing* 23 (1979) 309–331.

[13] M.J.D. Powell, "A method for non-linear constraints in minimization problems" in: R. Fletcher, ed., *Optimization* (Academic Press, London, 1969).

[14] J.B. Rosen and S. Suzuki, "Construction of non-linear programming test problems", *Communications of the ACM* 8 (1965) 113.

[15] W. Zangwill, "Nonlinear programming via penalty functions", *Management Science* 13 (1967) 344–350.