A MODIFIED NELDER-MEAD BARRIER METHOD FOR CONSTRANDED OPTIMIZATION

C. J. Price
Mathematics and Statistics
University of Canterbury
Christchurch, New Zealand

(Communicated by Zhong Wan)

Abstract. An interior point modified Nelder Mead method for nonlinearly constrained optimization is described. This method neither uses nor estimates objective function or constraint gradients. A modified logarithmic barrier function is used. The method generates a sequence of points which converges to KKT point(s) under mild conditions including existence of a Slater point. Numerical results are presented that show the algorithm performs well in practice.

1. Introduction. This paper describes a direct search method for inequality constrained nonlinear optimization. The proposed method is a synthesis of the ideas behind barrier methods [10] and a variant [26] of the Nelder Mead algorithm [22]. The latter means only function values are needed, and the use of barrier techniques allows the objective to be undefined or infinite at points which violate constraints. Hence the method can be applied to problems which lack available derivatives. More recent developments in direct search methods [6, 15, 23, 25, 31, 32] allow an asymptotic convergence theory to be constructed. An instance of the method is also numerically tested on a variety of both smooth and non-smooth test problems.

The form of the nonlinearly constrained optimization problem considered here is

\[
\min f(x) \quad \text{subject to} \quad c_i(x) \leq 0 \quad \forall i = 1, \ldots, q
\]

where \( f \) is the objective function, and \( c = (c_1, c_2, \ldots, c_q)^T \) is the vector of constraint functions. It is assumed the objective and constraint functions are ‘black-box,’ with only function values able to be calculated at selected points.

In essence, barrier methods [10] construct a function \( B(x, w) \) with a minimizer near a solution of (1). This minimizer is dependent on the barrier parameter \( w > 0 \) and strictly satisfies the constraints (i.e. \( c_i < 0, \ i = 1, \ldots, q \)). Two well known examples of barrier functions for (1) are the inverse and logarithmic barrier functions

\[
B_{\text{inv}} = f(x) + w \sum_{i=1}^{q} \frac{-1}{c_i(x)} \quad \text{and} \quad B_{\text{log}} = f(x) + w \sum_{i=1}^{q} -\log(-c_i(x))
\]

These two barrier functions both fit the standard form \( B = f + w \sum_i P(c_i(x)) \) for a barrier function, where \( P \) is an increasing function of \( c_i \) such that \( P(c_i) \to \infty \) as \( c_i \to 0 \) from below. Each \( P(c_i(x)) \) term applies a penalty to \( B \) which increases to

2010 Mathematics Subject Classification. Primary: 65K05.

Key words and phrases. Derivative free, positive basis, convergence analysis, frames, direct search, numerical results.
infinity as any point where \( c_i(x) = 0 \) is approached. An immediate consequence of this is that any minimizers of \( B \) must have \( c_i < 0 \) for all \( i = 1, \ldots, q \), and so are unconstrained.

A barrier method typically minimizes \( B(x, w) \) for each of a sequence \( \{w_m\} \) of values of \( w \), where \( w_m \to 0 \) as \( m \to \infty \). The reason a sequence of \( w \) values is needed is that the minimizer of each \( B(x, w_m) \) with respect to \( x \) serves as the starting point for the minimization of \( B(x, w_{m+1}) \). This sequential process is much more robust than simply minimizing \( B(x, w) \) once with a very small \( w \) value because \( B \) almost always becomes increasingly ill conditioned as \( w \to 0 \). Since each minimizer of \( B(x, w) \) has no active constraints, the process of minimizing \( B \) can, in principle, be done by an unconstrained optimization method such as a variant of Nelder Mead.

The Nelder Mead method was first introduced in 1965 [22], and was initially popular. Its fortunes later waned somewhat due to competition from gradient based methods, before enjoying a resurgence driven in part by an increasing interest in problems which lacked available gradients [30]. This resurgence included a deeper examination of Nelder Mead’s characteristics [14, 16, 17], as well as a counter-example showing that the original Nelder Mead method is not guaranteed to converge even on twice continuously differentiable (\( C^2 \)) problems in 2 dimensions [19]. In response to [19], several convergent variants of Nelder Mead for unconstrained optimization [21, 26, 29] have been developed. These methods provide motivation for a Nelder Mead variant adapted to the role of minimizing barrier functions.

Nelder Mead begins with a polytope (or simplex) of \( n+1 \) points in \( \mathbb{R}^n \), where this initial polytope is typically large. At each iteration it seeks an improved polytope by modifying the current polytope. Preferentially, it tries to replace the highest point in the polytope with a lower point along the line through this highest point and the centroid of the remaining polytope points. If successful, this yields a new polytope, and completes an iteration. Otherwise, all polytope points except the lowest are shifted partway toward the lowest, yielding a new smaller polytope and completing an iteration. The strategy of replacing the highest polytope point is then resumed. The method terminates when the polytopes become sufficiently small.

A jejune approach to constructing a barrier based Nelder Mead method for solving (1) would be to use a variant of Nelder Mead to minimize \( B(x, w) \) for each value of \( w \) in the sequence \( \{w_m\} \). As such, a complete execution of the Nelder Mead variant is embedded inside each iteration of the barrier method. Much of each Nelder Mead execution might be wasted if Nelder Mead uses an initial polytope that is far larger than necessary for the current \( B(x, w) \), or if \( B \) is minimized to an unnecessarily high accuracy, or both. Moreover, how to transfer information from the very small polytope at the end of one minimization of \( B \) to the large initial polytope at the start of the next minimization is far from clear.

A more nuanced approach is to interlace changes in \( w \) with iterations of a single execution of the Nelder Mead variant. This provides scope for increased efficiency via a reduction in the total number of Nelder Mead iterations. It also retains the current polytope across changes in \( w \), thus preserving valuable information.

This approach is the subject of this paper, which is organized as follows. The next section describes the barrier function and gives a template for the method. Section 3 details the modified Nelder Mead sub-algorithm. Section 4 establishes asymptotic convergence theory under somewhat more restrictive conditions than are needed to execute the method. These include continuity of \( f \), that the constraint functions \( c_i \) are locally Lipschitz, and strict differentiability of the objective and all active
constraint functions at relevant limit point(s). Continuous differentiability is not required. Numerical results and a discussion are given in Section 5, with concluding remarks appearing in Section 6. The new method is named **penmeco**, which is short for ‘Pseudo-Expand Nelder Mead for Constrained Optimization.’

2. **The Nelder Mead Barrier Algorithm.** The method uses a modified logarithmic barrier function which is finite on the set of points $\Omega$ which strictly satisfy all constraints. That is to say $\Omega$ is the set

$$\Omega = \{ x \in \mathbb{R}^n : c_i(x) < 0 \ \forall i = 1, \ldots, q \}$$

This set $\Omega$ is usually a strict subset of the feasible region $F$, which is defined as

$$F = \{ x \in \mathbb{R}^n : c_i(x) \leq 0 \ \forall i = 1, \ldots, q \}.$$

The fact that the constraints are black-box means it is non-trivial to distinguish between interior and boundary points of $F$ at which one or more constraints equal zero. Hence $\Omega$ is used as a proxy for the interior $F^\circ$ of $F$, and the closure $\overline{\Omega}$ in turn is used as a proxy for the feasible region $F$. An example where $\Omega \neq F^\circ$ and $\overline{\Omega} \neq F$ is when $c = -[(x) +]^2$ with $n = q = 1$, where $[x]_+ = \max(x, 0)$.

The modified logarithmic barrier function $B(x, w)$ used herein is given by

$$B(x, w) = \begin{cases} f(x) + w \sum_{i=1}^q P(c_i(x)) & x \in \Omega \\ \infty & \text{otherwise.} \end{cases} \quad (2)$$

Here $w$ is the barrier parameter, and $P$ is the modified (natural) logarithm function

$$P(c_i) = -\log_e \left( \frac{-c_i}{r - c_i} \right) \quad (3)$$

where $r > 0$ is a fixed parameter. A different value of $r$ can be used for each constraint. The $P$ in (3) is an increasing smooth function mapping $(-\infty, 0)$ into $(0, \infty)$, where $P \uparrow \infty$ as $c_i \uparrow 0$. The first two derivatives of $P$ are

$$P'(c_i) = \frac{-r}{c_i(r - c_i)} \quad \text{and} \quad P''(c_i) = \frac{1}{c_i^2} - \frac{1}{(r - c_i)^2}$$

which shows that $P''$ is positive and bounded above by $1/c_i^2$ when $c_i < 0$. The non-negativity of $P$ means $B$ has two important properties needed to show convergence:

1. $B(x, w) \geq f(x)$ for all $x$ and for all $r > 0$ and $w \geq 0$; and
2. $u \leq w$ implies $B(x, u) \leq B(x, w)$ for all $x$.

The barrier function in (2) is defined at all points in $\mathbb{R}^n$, allowing an unconstrained optimization algorithm such as a modified Nelder Mead algorithm to be applied to it.

The barrier algorithm requires an initial point $z_0 \in \Omega$. Each $B(x, w_m)$ is approximately minimized with respect to $x$ for each member of a non-strictly decreasing sequence $\{w_m\}$ of positive barrier parameter values. The initial point for the first minimization is $z_0$, and for each subsequent iteration it is the approximate minimizer from the previous minimization. Since $w$ is held fixed during each minimization, the notation $B_m(x) = B(x, w_m)$ is used frequently.

A modified Nelder Mead algorithm is used to approximately minimize each $B_m$ by generating a sequence of polytopes until descent is poor. The final polytope is reshaped if needed, and one extra point added to form a structure called a frame of $n + 1$ points around the polytope’s lowest point. If completing this frame does not yield sufficient descent the minimization of $B_m$ ends, otherwise the Nelder Mead
iterations recommence. The frames that do not yield sufficient descent are central to the convergence theory.

2.1. Frames. Frames have been discussed in depth before in [6, 7, 8], and so the treatment here will be brief. Herein all frames are defined using minimal positive bases [9], where a minimal positive basis is simply a set \( V_+ = \{ v_1, \ldots, v_{n+1} \} \subset \mathbb{R}^n \) with the property that any vector in \( \mathbb{R}^n \) can be written as a linear combination of \( v_1, \ldots, v_{n+1} \) with non-negative coefficients. It can be shown [9] that all vectors in a minimal positive basis must be non-zero. Minimal positive bases have the useful property that, given \( g \in \mathbb{R}^n \), if \( v^T g \geq 0 \) for all \( v \in V_+ \), then \( g = 0 \). This is easily seen by writing

\[
-g = \sum_{i=1}^{n+1} \eta_i v_i \quad \text{with} \quad \eta_i \geq 0 \quad \text{for all} \quad i = 1, \ldots, n+1,
\]

\[
\Rightarrow \quad g = 0.
\]

This result is the crux of frame based direct search methods. A minimal positive basis \( V_+ \) is used to form a frame \( \Phi(z, h, V_+) \) around \( z \), which is the set of points

\[
\Phi(z, h, V_+) = \{ z + hv : v \in V_+ \}.
\]

The vector \( z \in \mathbb{R}^n \) and scalar \( h > 0 \) are called the frame centre and frame size respectively, where \( z \notin \Phi \). Each such frame used by the algorithm is constructed from the current Nelder-Mead polytope. The point in the polytope with the lowest \( B_m \) value is the frame’s centre \( z \). All remaining polytope points are included in the frame \( \Phi \). One extra point is added to complete the frame.

In mesh or grid based unconstrained optimization [1, 28] a sequence of frames is generated, where each frame satisfies

\[
B(x, w) \geq B(z, w) \quad \forall x \in \Phi(z, h, V_+).
\]

Such frames are called minimal. When \( B \) is smooth, a minimal frame yields finite difference approximations of the directional derivatives at \( z \) along each \( v \) of the form

\[
v^T \nabla_v B(z, w) \approx \frac{B(z + hv, w) - B(z, w)}{h} \geq 0 \quad \forall v \in V_+.
\]

Loosely speaking, minimal frame centres are a discrete equivalent of local minimizers, and by taking limits [25] it can be shown that these minimal frame centres converge to stationary points of \( B \) under appropriate conditions.

Herein (5) is relaxed slightly, giving a quasi-minimal frame [6], viz.

\[
B(x, w) \geq B(z, w) - \epsilon \quad \forall x \in \Phi(z, h, V_+)
\]

where \( \epsilon > 0 \) is the maximum acceptable deviation from minimality. A frame is either quasi-minimal, or contains a point \( x \) satisfying \( B(x, w) < B(z, w) - \epsilon \). Any such frame point \( x \) is called a point of sufficient descent. The quasi-minimal frames (indexed by \( m \)) are identified by the symbol \( \Phi^{(m)}(z_m, h_m, V_+^{(m)}) \). Each \( \Phi^{(m)} \) is generated by approximately minimizing \( B_m(x) = B(x, w_m) \) with respect to \( x \), halting when a frame within \( \epsilon_m \) of minimality is found. The resultant sequences of quasi-minimal frame centres \( \{ z_m \} \), sizes \( \{ h_m \} \), positive bases \( \{ V_+^{(m)} \} \), and maximum acceptable deviations from minimality \( \{ \epsilon_m \} \) are central to establishing convergence.

Conditions must also be imposed on the sequence \( \{ V_+^{(m)} \}_{m=1}^{\infty} \) to ensure that its limits are also positive bases. One convenient approach is to assume the members of each \( V_+^{(m)} \) have a specific order. This means limit(s) of a sequence of ordered positive bases can be simply defined as follows.
Definition 2.1. An ordered set $V^*_+ = \{v_1^*, \ldots, v_{n+1}^*\}$ is a limit of the sequence of ordered sets $\{V_{+}^{(m)}\}_{m=1}^{\infty}$ if and only if
\[
\lim_{m \to \infty} v_j^{(m)} = v_j^* \quad \forall j = 1, \ldots, n + 1
\]
where $m$ is restricted to some infinite subset of the natural numbers $\mathbb{N}$.

Provided the members of all $V_{+}^{(m)} = \{v_1^{(m)}, \ldots, v_{n+1}^{(m)}\}$ are uniformly bounded, the sequence $\{V_{+}^{(m)}\}$ must have at least one limit point. Hence it is assumed that
\[
\exists K \text{ such that } \forall m \in \mathbb{N}, \quad \forall v_j^{(m)} \in V_{+}^{(m)}, \quad \|v_j^{(m)}\| \leq K
\] (7)
Finally, it is necessary that the positive bases do not collapse in the limit $m \to \infty$.

This motivates the following assumption.

Assumption 1. All limits of the sequence $\{V_{+}^{(m)}\}_{m=1}^{\infty}$ of positive bases are also positive bases.

This assumption can be enforced by resetting collapsed or nearly collapsed polytopes whenever sufficient descent is not obtained [26], via any of the options given in [6]. Here sufficient descent means a reduction in $B$ of more than $\epsilon$.

Assumption 1 and the fact that no member of a minimal positive basis can be zero means $\|v_j^{(m)}\|$ is not only bounded via (7), it is also bounded away from zero for all $m \in \mathbb{N}$ and $j = 1, \ldots, n + 1$.

2.2. The Barrier Method. A framework for a Nelder Mead based barrier algorithm is as follows.

Algorithm 1. The Constrained Barrier Framework

1. Set $m = 1$, choose the initial point $z_0 \in \Omega$. Pick $h_m > 0$, $\epsilon_m > 0$ and $w_m > 0$.

   Form the initial polytope.

2. Using the modified Nelder Mead sub-algorithm, generate a quasi-minimal frame $\Phi^{(m)} = \Phi(z_m, h_m, V_{+}^{(m)})$ for which $B_m(z_m) \leq B_m(z_{m-1})$.

3. Increment $m$ and calculate $w_m > 0$, $h_m > 0$, and $\epsilon_m > 0$. Form the new polytope. If stopping conditions do not hold, go to step 2.

Step 1 is an initialization step that requires an initial point $z_0 \in \Omega$. Step 2 performs iterations of the modified Nelder Mead method until a quasi-minimal frame is found. The condition $B_m(z_m) \leq B_m(z_{m-1})$ in step 2 can be ensured simply by including $z_{m-1}$ in the new polytope in step 3 before minimizing $B_m$ is commenced. Step 3 adjusts the frame size $h$, measure of sufficient descent $\epsilon$, and barrier parameter $w$. Adjusting the former two must satisfy

Assumption 2. Both $h_m \to 0$ and $\epsilon_m/h_m \to 0$ as $m \to \infty$.

These can easily be achieved by using, for example, $h_m = h_{m-1}/2$ and $\epsilon_m = Gh_m^{3/2}$, for a fixed $G > 0$. Simple choices also exist for the initial and new polytopes in steps 1 and 3. Herein the initial polytope is $z_0$, together with points obtained from taking steps of length $h_1$ along each coordinate axis from $z_0$. A similar construction can be used in step 3 with $h_m$ and $z_{m-1}$ in place of $h_1$ and $z_0$. A slightly more sophisticated approach is to then align one edge of this new polytope along a favourable direction by reflecting it through a plane containing $z_{m-1}$. 
2.3. Adjusting \( w \). The basic strategy behind adjusting \( w \) is to reduce it towards zero sufficiently slowly so that the frames’ sizes shrink to zero faster than the frames’ centres approach the boundary of \( \Omega \). This means that the logarithmic singularities become distant on the scale of a frame, which is needed to show convergence. To this end the maximum difference \( D_i^{(m)} \) of each constraint’s values between the frame and its centre is specified via
\[
D_i^{(m)} = \max \left\{ \left| c_i(x) - c_i(z_m) \right| : x \in \Phi^{(m)} \right\}.
\]

Herein \( w \) is only reduced in step 3 of Algorithm 1 when
\[
w_m \sum_{i=1}^{q} \left( \frac{D_i^{(m)}}{c_i(z_m)} \right)^2 \leq W (h_m)^{3/2}
\]
for a fixed positive constant \( W \).

**Assumption 3.** For any \( w_1 > 0 \) the updating process for \( w \) ensures
(a) \( w_m \to 0 \) as \( m \to \infty \) provided \( w \) is updated infinitely often; and
(b) \( \exists K_w > 0 \) such that \( w_m \geq K_w h_m \) for all \( m \in \mathbb{N} \).

Part (b) of this assumption means the variation in \( c_i \) values across a frame becomes small relative to the constraint values at the frame’s centre for some sub-sequence of quasi-minimal frames. Specifically, for any \( m \) at which (8) holds, for all \( x \in \Phi^{(m)} \) and for all \( i = 1, \ldots, q \)
\[
\left| \frac{c_i(x) - c_i(z_m)}{c_i(z_m)} \right|^2 \leq \sum_{j=1}^{q} \left( \frac{D_j^{(m)}}{c_j(z_m)} \right)^2 \leq \frac{W (h_m)^{3/2}}{w_m} \leq \frac{w \sqrt{h_m}}{K_w}
\]
and the right hand side converges to zero as \( m \to \infty \) by Assumption 2. When \( W \sqrt{h_m} < K_w \), (9) implies \( c_i(x) < 0 \) \( \forall i = 1, \ldots, q \) and \( \forall x \in \Phi^{(m)} \), giving \( \Phi^{(m)} \subset \Omega \).

3. **The Nelder Mead sub-algorithm.** The modified Nelder Mead sub-algorithm uses a polytope in \( \mathbb{R}^n \) consisting of \( n+1 \) points \( x_0, \ldots, x_n \) which have been sorted in increasing order of their \( B_m \) values, with \( b_i = B_m(x_i) \) for \( i = 0, \ldots, n \). At each iteration the worst point \( x_n \) is shifted along a line through it and the centroid of the remaining \( n \) points. When this fails to yield sufficient descent, a step called the pseudo-expand step [26] is attempted, which might require reshaping the polytope first. This pseudo-expand step either yields sufficient descent, or gives a quasi-minimal frame. In the former case iterations of Nelder Mead are recommenced. In the latter, the sub-algorithm halts and returns the quasi-minimal frame.

In the following description of the modified Nelder Mead sub-algorithm, iteration numbers on various quantities are omitted in order to minimize clutter. An illustration of the points which could occur in one iteration of the modified Nelder Mead sub-algorithm is given in Figure 1.

**Algorithm 2. The modified Nelder Mead sub-algorithm**
1. Re-order the current polytope’s vertices \( x_0, \ldots, x_n \) so that \( b_0 \leq b_1 \leq \cdots \leq b_n \).
   
   \( b_{\text{new}} = \infty \).
2. Find the centroid \( \bar{x} \) of \( x_0, \ldots, x_{n-1} \), set \( p = \bar{x} - x_n \) and set \( x_r = \bar{x} + \alpha p \).
3. Calculate \( b_r = B_m(x_r) \), and then do one of the four sub-steps:
   (a) If \( b_r < b_0 \), set \( x_{\text{new}} = \bar{x} + \gamma p \) and calculate \( b_{\text{new}} = B_m(x_{\text{new}}) \). If \( b_e < b_r \) set \( x_{\text{new}} = x_e \) and \( b_{\text{new}} = b_e \), otherwise set \( x_{\text{new}} = x_r \) and \( b_{\text{new}} = b_r \).
   (b) If \( b_0 \leq b_r < b_{n-1} \) set \( x_{\text{new}} = x_r \) and \( b_{\text{new}} = b_r \).
α no longer occurs. The parameters method (without shrinks) until sufficient descent from the polytope’s worst point expansion coefficients of standard Nelder Mead, and must satisfy
cient reduction below $x_{\text{new}} = x_{\text{ic}}$ and $b_{\text{new}} = b_{\text{ic}}$.
(d) If $b_n \leq b_r$, set $x_{\text{ic}} = \bar{x} + \beta p$ and calculate $b_{\text{ic}} = B_m(x_{\text{ic}})$. If $b_{\text{ic}} < b_n$ set $x_{\text{new}} = x_{\text{ic}}$, set $b_{\text{new}} = b_{\text{ic}}$.

4. If $b_{\text{new}} < b_n - \epsilon$ replace $x_n$ with $x_{\text{new}}$ to get a new polytope and go to step 1.
5. If necessary, reshape the current polytope $x_0, \ldots, x_n$, to get a new polytope $x_0, y_1, \ldots, y_n$. Otherwise set $y_i = x_i$ for $i = 1, \ldots, n$. If $B_m(y_i) < b_0 - \epsilon$ for any $i = 1, \ldots, n$, go to step 1. Calculate the centroid $\bar{y}$ of $y_1, \ldots, y_n$.
6. Set $x_{\text{pe}} = \bar{y} + (\gamma / \alpha)(x_0 - \bar{y})$, and calculate $b_{\text{pe}} = B_m(x_{\text{pe}})$.
7. If $b_{\text{pe}} < b_0 - \epsilon$, replace $x_0$ with $x_{\text{pe}}$ to get a new polytope, and go to step 1. Otherwise return the quasi-minimal frame $\{y_1, \ldots, y_n, x_{\text{pe}}\}$ about $x_0$ and halt.

Steps 1 to 4 form a loop which executes iterations of the original Nelder Mead method (without shrinks) until sufficient descent from the polytope’s worst point no longer occurs. The parameters $\alpha$, $\beta$, and $\gamma$ are the reflection, contraction, and expansion coefficients of standard Nelder Mead, and must satisfy
\[ 0 < \beta < \min\{\alpha, 1\} \quad \text{and} \quad \gamma > \max\{\alpha, 1\}. \quad (10) \]

Step 5 reshapes the polytope if necessary, so that the frame generated in step 6 via the pseudo-expand step satisfies the necessary conditions for convergence. Specifically, this reshaping together with condition (7) must ensure the underlying sequence $\{V_{+}^{(m)}\}$ of positive bases satisfies Assumption 1. A strategy for identifying which polytopes require reshaping is given in [26]. A simpler approach is to reshape all polytopes in step 5 whether they need it or not. This uses more function evaluations, but these extra evaluations either provide additional evidence an approximate minimizer of $B_m$ has been found, or yield a point of sufficient descent.

Step 6 attempts the pseudo-expand step by assuming $x_0$ is the reflection point from a fictitious previous polytope, where the $n$ best points in that fictitious polytope are $y_1, \ldots, y_n$. The expansion point for this fictitious polytope is the pseudo-expand point $x_{\text{pe}}$ for the actual polytope. Continuing to treat $x_{\text{pe}}$ as an expansion point from a fictitious polytope, step 7 replaces $x_0$ with $x_{\text{pe}}$ if the latter gives sufficient reduction below $x_0$. Otherwise $\Phi^{(m)} = \{y_1, \ldots, y_n, x_{\text{pe}}\}$ forms a quasi-minimal frame around the frame centre $x_0 = z_m$.

4. Convergence Results. In this section the convergence properties of methods conforming to the template given in Algorithm 1 are examined. The convergence results are asymptotic, and assume the method is performed without stopping in exact arithmetic. Assumptions 1, 2, and 3 can be guaranteed directly by the way an algorithm is constructed. The remaining assumptions are directly or indirectly problem dependent, and are listed here:

Assumption 4.

(a) The sequence of quasi-minimal frame centres $\{z_m\}$ is bounded;
(b) The objective function is continuous and bounded below on $F$ by $f_{\text{min}} \in \mathbb{R}$;
(c) All constraints are locally Lipschitz; and
(d) All functions in problem (1) are strictly differentiable on $F$ at each limit point $z^*$ of the sequence of quasi-minimal frame centres $\{z_m\}$.

First we show that the subsequence of quasi-minimal frames is infinite. The approach taken is to show that, if the subsequence of quasi-minimal frames is finite, then Assumption 4(b) fails.
The current polytope \( x_0, x_1, \) and \( x_2 \) is shown in solid lines. The modified Nelder Mead sub-algorithm first tries to replace the worst polytope point \( x_2 \) with one of \( x_{ic}, x_{oc}, x_r \) or \( x_e \). These four points lie on the dashed line through \( x_2 \) and the centroid \( \bar{x} \) of the remaining polytope points. If this fails to give sufficient descent, the fictitious polytope \( y_1, y_2, H \) is used, where the fictitious polytope’s worst point \( H \) is reflected through the centroid \( \bar{y} \) of that polytope’s remaining points, giving a fictitious new low point \( x_0 \). The expand step is then tried from this fictitious polytope, giving the pseudo-expand point \( x_{pe} \). The points \( y_1, \ldots, y_n \) (\( y_1 \) and \( y_2 \) here) and \( x_{pe} \) form a frame around \( x_0 \).

**Theorem 4.1.** The sequence of quasi-minimal frames is infinite.

**Proof.** Assume the sequence of quasi-minimal frames is finite, and let there be \( J \) such frames in total. Once \( m = J \) occurs, Algorithm 2 never halts, and the positive parameters \( \epsilon_m \) and \( w_m \) are constant from then on. At each non-terminating iteration of Algorithm 2, either

(a) step 4 replaces the worst polytope point \( x_n \) with a new point \( x_{new} \) satisfying \( B_m(x_{new}) < B_m(x_n) - \epsilon_m \); or

(b) The best polytope point \( x_0 \) is replaced with either one of \( y_1, \ldots, y_n \) in step 5 or \( x_{pe} \) in step 7. In either case the value of \( B_m \) at this replacement point is less than \( B_m(x_0) - \epsilon_m \).

Since the lowest polytope point is always retained even during resets, if case (b) occurs infinitely often, then the sequence of minimum \( B_m \) values on the polytopes is unbounded below. Since \( f(x) \leq B_m(x) \) everywhere, this contradicts Assumption 4(b). Hence case (b) can only occur a finite number of times.

Once case (b) has occurred for the final time, case (a) occurs in all future iterations. Case (a) does not reset the polytope because step 5 is not performed. Hence all polytope points other than \( x_n \) are retained from one iteration to the next, and the quantity \( \sum_{i=0}^{n} B_m(x_i) \) is reduced by more than \( \epsilon_m \) in each such iteration. This also shows that the sequence of \( B_m \) values on the polytopes are unbounded below, again yielding a contradiction. Hence \( \{z_m\} \) must be an infinite sequence. \( \square \)
Corollary 1. Assumption 4(a) implies the sequence of quasi-minimal frame centres \( \{z_m\} \) has one or more limit points.

Theorem 4.1 also implies \( h \) is reduced infinitely often, which is important in satisfying Assumption 2. The next two results show \( w \) is also reduced infinitely often, which is needed if Assumption 3(a) is to hold.

**Proposition 1.** The sequence \( \{B(z_m, w_m)\} \) is non-strictly decreasing.

**Proof.** Via

\[
B(z_m, w_m) \leq B(z_{m-1}, w_m) \leq B(z_{m-1}, w_{m-1})
\]

where the left hand inequality is because \( B_m(z_m) \leq B_m(z_{m-1}) \) from step 2 of Algorithm 1. The right hand inequality is because \( B \) is a non-strictly increasing function of \( w \), and \( w_m \leq w_{m-1} \).

**Proposition 2.** Inequality (8) holds for an infinite number of \( m \in \mathbb{N} \), and the sequence \( \{w_m\} \) of barrier parameters satisfies \( w_m \to 0 \) as \( m \to \infty \).

**Proof.** The second part is a direct consequence of the first, by Assumption 3. The proof that inequality (8) holds infinitely often is by contradiction. Assume the final time inequality (8) holds is in iteration \( J \). Then \( w_m = w^* > 0 \) for all \( m \geq J \).

Now the first iterate \( z_0 \in \Omega \) which means \( B(z_0, w_0) \) is finite. Since \( B(z, w) \geq f(z) \) for all \( w \geq 0 \) and \( z \in \Omega \), and \( f \) is bounded below on \( F \) by \( f_{\min} \), we have

\[
B(z_0, w_0) \geq B(z_m, w_m) \geq f_{\min} + w^* \sum_{j=1}^{q} P(c_j(z_m)) \geq f_{\min} + w^* \sum_{j=1}^{q} P(c_j(z_m))
\]

for each \( m \geq J \) and \( i = 1, \ldots, q \) via Proposition 1. This implies

\[
P(c_i(z_m)) \leq \frac{B_0(z_m) - f_{\min}}{w^*} \quad \forall m \geq J \quad \text{and} \quad \forall i = 1, \ldots, q
\]

where the right hand side is finite. Hence the sequences \( \{c_i(z_m)\}_{m \geq J} \) are bounded away from zero for all \( i = 1, \ldots, q \).

Assumption 2 and the fact that \( c_i \) is locally Lipschitz (with a Lipschitz constant \( \lambda_i \) say) imply \( D_i^{(m)} \) in (8) satisfies

\[
D_i^{(m)} \leq K \lambda_i h_m = O(h_m)
\]

as \( m \to \infty \). Hence the left hand side of (8) is \( O(h_m^2) \) while the right hand side is \( W(h_m)^{3/2} \) with \( W > 0 \). Since \( h_m \to 0 \) as \( m \to \infty \), this is a contradiction.

4.1. **The Lagrangian.** In order to show convergence to one or more Karush-Kuhn-Tucker (KKT) points, the Lagrangian

\[
L(x, \zeta) = f(x) - \sum_{i=1}^{q} \zeta_i c_i(x)
\]

is used, where the \( \zeta_i \) play the role of Lagrange multiplier estimates. The barrier function \( B \) and the Lagrangian can be connected as follows. Using

\[
\Delta f = f(x + hv) - f(x) \quad \text{and} \quad \Delta c_i = c_i(x + hv) - c_i(x)
\]

we have

\[
\Delta L = L(x + hv, \zeta) - L(x, \zeta) = \Delta f - \sum_{i=1}^{q} \zeta_i \Delta c_i
\]
Similarly
\[ \Delta B = B(x+hv,w) - B(x,w) = \Delta f + w \sum_{i=1}^{q} [P(c_i(x) + \Delta c_i) - P(c_i(x))] \]
and so
\[ \Delta B = \Delta f + w \sum_{i=1}^{q} \left[ P'(c(x))\Delta c_i + \frac{1}{2}P''(c(x) + \theta_i \Delta c_i)\Delta c_i^2 \right] \]
for some \( \theta_i \in [0,1] \) for each \( i = 1, \ldots, q \) by the intermediate value theorem. Setting \( x = z_m \), if the Lagrange multiplier estimates are defined via
\[ \zeta_i^{(m)} = -w_m P'(c_i(z_m)) \]
then \( \Delta B \) and \( \Delta L \) agree to the first order at \( z_m \). Consequently when a frame centre appears to be a discrete stationary point of \( B \), it is also appears to be a discrete stationary point of \( L \).

This fact is used to establish the KKT conditions for selected limit points \( z^* \) of the sequence of quasi-minimal frame centres \( \{z_m\} \) under mild conditions. These conditions include that the sequence \( \{\zeta^{(m)}\} \) of Lagrange multiplier estimates remains bounded, where the notation \( \zeta^{(m)} = (\zeta_1^{(m)}, \zeta_2^{(m)}, \ldots, \zeta_q^{(m)})^T \) has been used. Further remarks on the boundedness of \( \zeta \) are deferred until subsection 4.4. A result connecting the sequence of quasi-minimal frames to function gradients via strict differentiability is given next, followed by the main convergence result on the KKT conditions.

4.2. Strict Differentiability. A function \( \psi \) is said to be strictly differentiable on the set \( \Omega \) (see e.g. [5]) at a point \( z \in \Omega \) if and only if
\[ \exists g \in \mathbb{R}^n \text{ such that } \psi(y) - \psi(x) - (y-x)^T g = o(||y-x||) \]
as \( (x,y) \to (z,z) \) over \( (x,y) \in \Omega \times \Omega \). The notation \( \nabla \psi^* \) will be used for \( g \) when \( \psi \) is strictly differentiable at the point \( z^* \).

**Theorem 4.2.** Let \( \psi \) be strictly differentiable over the set \( \Omega \) at \( z^* \in \Omega \). Let \( \{\Phi^{(m)}\} \subset \Omega \) be a sequence of quasi-minimal frames for \( \psi \), with the corresponding sequences of frame centres \( \{z_m\} \), sizes \( \{h_m\} \), and positive bases \( \{V_+^{(m)}\} \) converging to \( z^*, 0, \) and the positive basis \( V_+^* \) respectively. Then \( \nabla \psi(z^*) = 0 \).

**Proof.** Denoting the \( j \)th member of \( V_+^{(m)} \) by \( v_j^{(m)} \), the uniform bound
\[ \|v_j^{(m)}\| \leq K \quad \forall j = 1, \ldots, n+1 \quad \text{and} \quad \forall m = 1, 2, \ldots \]
in (7) implies
\[ z_m + h_m v_j^{(m)} \to z^* \quad \text{as} \quad m \to \infty. \]
Quasi-minimality implies
\[ \psi\left(z_m + h_m v_j^{(m)}\right) - \psi\left(z_m\right) \geq -\epsilon_m \] (13)
Strict differentiability of \( \psi \) at \( z^* \) implies there exists a \( g \in \mathbb{R}^n \) such that
\[ \psi\left(z_m + h_m v_j^{(m)}\right) - \psi\left(z_m\right) - h_m g^T v_j^{(m)} = o(h_m) \]
Hence from (13),
\[ h_m g^T v_j^{(m)} + o(h_m) \geq -\epsilon_m \]
Dividing by \( h_m \) and allowing \( m \to \infty \), this gives
\[
g^T v_j^* \geq 0 \quad \forall v_j^* \in V^*_m
\]
where \( v_j^{(m)} \to v_j^* \) for all \( j = 1, \ldots, n + 1 \) as \( m \to \infty \). Hence \( g = 0 \) by, for example, Theorem 2.1 of [7], or via (4).

4.3. The Main Convergence Result. The main convergence result is now stated.

**Theorem 4.3.** Let Assumptions 1, 2, 3 and 4 hold. Also let \( \mathcal{M} \subseteq \mathbb{N} \) be such that (8) holds for all \( m \in \mathcal{M} \), and the sequences \( \{z_m\}, \{\zeta^{(m)}\} \) and \( \{V^+_m\} \) converge to \( z^*, \zeta^* \), and the positive basis \( V^*_m \) respectively as \( m \to \infty \) over \( \mathcal{M} \). Let \( f \) and all constraints satisfying \( c_i(z^*) = 0 \) be strictly differentiable at \( z^* \). Then \( (z^*, \zeta^*) \) is a KKT point of problem (1).

**Proof.** First, for feasibility, \( z_m \in \Omega \) for all \( m \) implies \( z^* \in \overline{\Omega} \). However \( \overline{\Omega} \subseteq F \), and so \( z^* \) is feasible.

Second, the form of the Lagrange multiplier estimates (12) means \( \zeta^* \leq 0 \).

Third is complementarity. If constraint \( i \) is inactive at \( z^* \), then \( c_i(z^*) < 0 \) and \( P(c_i(z^*)) \) is finite. Hence \( P(c_i(z_m)) \to P(c_i(z^*)) \) as \( m \to \infty \) over \( \mathcal{M} \). So \( w_m \to 0 \) implies \( \zeta_i^{(m)} = -w_mP(c_i(z_m)) \to 0 \) as \( m \to \infty \), giving \( \zeta_i^* = 0 \), as required.

Since \( \zeta_i^* = 0 \) for all inactive constraints at \( z^* \), the \( \zeta_i^* c_i \) terms in \( L(z, \zeta^*) \) for the inactive constraints all vanish. Hence \( L(z, \zeta^*) \) is strictly differentiable at \( z = z^* \) because \( f \) and all constraints with \( c_i(z^*) = 0 \) are strictly differentiable at \( z^* \).

Finally, each \( V^+_m \) and the limiting positive basis \( V^*_m \) are all ordered sets. For an arbitrary fixed \( j \), consider the \( j \)th member \( v_j^{(m)} \) of \( V^+_m \). Using the notation
\[
\Delta f^{(m)} = f\left(z_m + h_m v_j^{(m)}\right) - f(z_m) \quad \text{and} \quad \Delta c_i^{(m)} = c_i\left(z_m + h_m v_j^{(m)}\right) - c_i(z_m),
\]
quasi-minimality of the frame \( \Phi^{(m)} \) means
\[
B\left(z_m + h_m v_j^{(m)}, w_m\right) - B(z_m, w_m) \geq -\epsilon_m \quad \forall m \in \mathcal{M}
\]
This implies
\[
f\left(z_m + h_m v_j^{(m)}\right) - f(z_m) + w_m \sum_{i=1}^q \left[P\left(c_i\left(z_m + h_m v_j^{(m)}\right)\right) - P\left(c_i(z_m)\right)\right] \geq -\epsilon_m
\]
for all \( m \in \mathcal{M} \). This in turn implies for all \( m \in \mathcal{M} \) and for all \( i = 1, \ldots, q \), there exist \( \theta_i^{(m)} \in [0, 1] \) such that
\[
\Delta f^{(m)} + w_m \sum_{i=1}^q \left[P'\left(c_i(z_m)\right) \Delta c_i + \frac{1}{2} P''\left(c_i(z_m) + \theta_i^{(m)} \Delta c_i\right) \Delta c_i^2\right] \geq -\epsilon_m \quad (14)
\]
where \( \Delta c_i \) has been used for \( \Delta c_i^{(m)} \). Hence the definition of \( \zeta^{(m)} \) and the Lagrangian \( L \) mean the previous line implies
\[
L(z_m + h_m v_j^{(m)}, \zeta^{(m)}) - L(z_m, \zeta^{(m)}) \geq -\epsilon_m - \frac{w_m}{2} \sum_{i=1}^q P''\left(c_i(z_m) + \theta_i^{(m)} \Delta c_i\right) \Delta c_i^2 \quad (15)
\]
Noting that
\[
L(z_m + h_m v_j^{(m)}, \zeta^{(m)}) - L(z_m, \zeta^{(m)}) = L(z_m + h_m v_j^{(m)}, \zeta^*) - L(z_m, \zeta^*) + \sum_{i=1}^q \Delta \zeta_i^{(m)} \Delta c_i
\]
where $\Delta \zeta_i^{(m)} = \zeta_i^{(m)} - \zeta^*$, inequality (15) becomes

$$L(z_m + h_{m} v_j^{(m)}, \zeta^*) - L(z_m, \zeta^*) \geq -\epsilon_m - \frac{\|w_m\|}{2} \sum_{i=1}^{q} \frac{P''(c_i(z_m))}{c_i} \Delta c_i^{(m)} + \sum_{i=1}^{q} \Delta \zeta_i^{(m)} \Delta c_i$$

For all $m \in \mathbb{M}$ sufficiently large, inequality (9) implies $|\Delta c_i| < |c_i|/2$ for all $i$ since $h_{m} \to 0$ as $m \to \infty$. This, together with the bound $P''(c_i) < 1/c_i^2$ implies $P''(c_i(z_m)) + \theta_i^{(m)} \Delta c_i) < 4/c_i^2$, which in turn implies

$$L(z_m + h_{m} v_j^{(m)}, \zeta^*) - L(z_m, \zeta^*) \geq -\epsilon_m - \sum_{i=1}^{q} 2w_m \frac{\Delta c_i^{2}}{c_i^2} - \sum_{i=1}^{q} \Delta \zeta_i^{(m)} \Delta c_i$$

and hence inequality (8) and Assumption 2 yield

$$L(z_m + h_{m} v_j^{(m)}, \zeta^*) - L(z_m, \zeta^*) \geq -o(h_{m}) - 2W(h_{m})^{3/2} - \sum_{i=1}^{q} \Delta \zeta_i^{(m)} \Delta c_i$$

The Lipschitz continuity of $c_i$ and the fact that $\zeta_i^{(m)} \to \zeta^*$ as $m \to \infty$ mean that the right hand term is $o(h_{m})$. Hence

$$L(z_m + h_{m} v_j^{(m)}, \zeta^*) - L(z_m, \zeta^*) \geq o(h_{m}) \quad \forall m \in \mathbb{M}, \quad \forall j = 1, \ldots, n + 1$$

which, in the limit $m \to \infty$, implies $\nabla_z L(z^*, \zeta^*) = 0$ via Theorem 4.2.

4.4. Remarks on the Boundedness of $\zeta$. The assumption that the sequence $\{\zeta^{(m)}\}$ of Lagrange multiplier estimates remains bounded is used in Theorem 4.3 to show asymptotic convergence. It can be shown that the boundedness of $\{\zeta^{(m)}\}$ is guaranteed when converging to a limit point $z^*$ at which $f$ and all active constraints are strictly differentiable, and the active constraints’ normals are linearly independent. For convenience let $A \subseteq \{1, \ldots, q\}$ be the indices of the set of active constraints at $z^*$.

**Theorem 4.4.** If all the conditions required by Theorem 4.3 apart from the condition on $\{\zeta^{(m)}\}$ hold, and if the active constraint normals are linearly independent at the limit point $z^*$, then the sequence $\{\zeta^{(m)}\}$ is bounded.

**Proof.** To see this we start from inequality (14), which has been established without use of any property of $\{\zeta^{(m)}\}$. Additionally, inequality (9) implies $|\Delta c_i| < |c_i|/2$ for all $i$ when $m \in \mathbb{M}$ is sufficiently large. This, and the bound $P''(c_i) < 1/c_i^2$ imply $P''(c_i(z_m)) + \theta_i^{(m)} \Delta c_i) < 4/c_i^2$ for all $i$ when $m \in \mathbb{M}$ is sufficiently large. Applying this bound to the second derivative term in (14) yields

$$\Delta f^{(m)} + w_m \sum_{i=1}^{q} P'(c_i(z_m)) \Delta c_i^{(m)} + 2w_m \sum_{i=1}^{q} \left( \frac{\Delta c_i^{(m)}}{c_i(z_m)} \right)^2 \geq -\epsilon_m$$

which holds for each $v_j^{(m)} \in V_+^{(m)}$, where $\Delta f^{(m)} = f(z_m + h_{m} v_j^{(m)}) - f(z_m)$ and similarly for $\Delta c_i^{(m)}$. Assumption 2 and inequality (8) imply

$$\Delta f^{(m)} - \sum_{i=1}^{q} \zeta_i^{(m)} \Delta c_i^{(m)} + 2W h_m^{3/2} \geq o(h_m)$$

(16)

where $\zeta_i^{(m)} = -w_m P'(c_i(z_m))$ has been used without further assumption.
Now for inactive constraints \( c_i(z^*) < 0 \) holds, which, together with \( w_m \to 0 \) as \( m \to \infty \), implies \( \zeta_i^{(m)} \to 0 \) as \( m \to \infty \). All constraints are also locally Lipschitz, which means the \( \Delta c_i^{(m)} = O(h_m) \). Hence each inactive constraint’s \( \zeta_i^{(m)} \Delta c_i^{(m)} \) term is \( o(h_m) \), and can be included in the \( o(h_m) \) term on the right hand side.

Strict differentiability at \( z^* \) means \( \Delta f^{(m)} = h_m \left( v_j^{(m)} \right)^T \nabla f(z^*) + o(h_m) \) and similarly for \( \Delta c_i^{(m)} \) for each active constraint. Hence (16) becomes

\[
\left( h_m \nabla f(z^*) - \sum_{i \in A} \zeta_i^{(m)} (h_m \nabla c_i(z^*) + o(h_m)) \right)^T v_j^{(m)} + 2Wh_m^{3/2} \geq o(h_m) \tag{17}
\]

The proof that \( \{\zeta^{(m)}\} \) remains bounded is by contradiction. Let \( \zeta_{\text{max}}^{(m)} = \|\zeta^{(m)}\|_\infty \). Replacing \( \mathbb{M} \) with a subset of itself if necessary, assume that \( \zeta_{\text{max}}^{(m)} > 0 \) and \( \zeta_{\text{max}}^{(m)} \to \infty \) as \( m \to \infty \) over \( m \in \mathbb{M} \). Define \( \eta^{(m)} = \zeta^{(m)}/\zeta_{\text{max}}^{(m)} \). Noting that \( \|\eta^{(m)}\|_\infty = 1 \) for all \( m \in \mathbb{M} \), it is clear that \( \{\eta^{(m)}\} \) must have limit(s), and these limit(s) must be non-zero. Dividing (17) through by \( h_m \zeta_{\text{max}}^{(m)} \) and allowing \( m \to \infty \) over a subset of \( \mathbb{M} \) for which \( \{\eta^{(m)}\} \) has a unique limit \( \eta^* \) gives

\[
\left( \sum_{i \in A} \eta_i^* \nabla c_i(z^*) \right)^T v_j^* \geq 0 \tag{18}
\]

where the \( \nabla f \) term has vanished because \( \zeta_{\text{max}}^{(m)} \to \infty \) as \( m \to \infty \). Since (18) holds for all \( v_j^* \) in a positive basis by Assumption 1, it follows that

\[
\sum_{i \in A} \eta_i^* \nabla c_i(z^*) = 0
\]

by Theorem 2.1 of [7], or via (4). Since \( \|\eta^*\|_\infty = 1 \) this contradicts the fact that the active constraint normals are linearly independent. \( \square \)

The convergence theory only applies to an ideal algorithm which is run forever in exact arithmetic. Realisable implementations necessarily deviate from this ideal on both counts, and hence might fail on some problems. The behaviour of one such implementation is examined in depth in the next section.

5. Implementation and Numerical Testing. The algorithm tested is as described above, with initial values \( G = h = w = r = 1 \) and \( W = n \). The values used for the Nelder Mead parameters were those recommended in [22]: the reflection, contraction, and expansion coefficients were \( \alpha = 1 \), \( \beta = 1/2 \) and \( \gamma = 2 \) respectively. The values recommended by Gao and Han [11] were tried, but gave a slightly worse performance than those from [22]. The shrink coefficient \( \sigma \in (0,1) \) for \( h \) was set at \( \sigma = 1/2 \), in line with [22]. Specifically \( h_{m+1} = \sigma h_m \) was used, along with \( w_{m+1} = \sigma w_m \) when condition (8) holds.

Algorithm 3. The Penmeco Implementation

1. Choose the initial point \( z_0 \). Set \( m = G = r = h_m = w_m = 1 \) and \( W = n \). Set \( h_{\text{min}} = 10^{-10} \). The initial polytope is \( \{z_0\} \cup \{z_0 + h_1 c_i : i = 1, \ldots, n\} \).
2. Execute Algorithm 2, returning the quasi-minimal frame \( \Phi(z_m, h_m, V_+^{(m)}) \).
3. Set \( h_{m+1} = \sigma h_m \). If inequality (8) holds, set \( w_{m+1} = \sigma w_m \) else use \( w_{m+1} = w_m \). Set \( \epsilon_{m+1} = Gh_{m+1}^{5/2} \).
4. If \( h_{m+1} < h_{\text{min}} \) halt. Otherwise increment \( m \), form the new polytope using the reshaping process given below, and go to step 2.

Here \( e_i \) is the \( i \)th unit coordinate vector. The algorithm also halted if a budget of 80,000 function evaluations was reached. In addition, a record of the best feasible point was kept, and this point returned as the solution found by PENMECO.

5.1. **Reshaping the Polytope.** In order to reduce algorithmic complexity in the implementation, every polytope \( x_0, \ldots, x_n \) is automatically reshaped in step 5 of Algorithm 2 and in step 4 of Algorithm 3 as follows. Assuming the points \( x_0, \ldots, x_n \) in the polytope are in increasing order of \( B_m \) value, the vector \( s = x_1 - x_0 \) is calculated. A Householder matrix \( H = I - 2uu^T \) is constructed, where \( u \) is the unit vector parallel to \( s \mp \|s\|e_1 \). Here \( e_1 \) is the first column of the identity matrix \( I \) and the \( \mp \) sign is chosen to match the sign of the first element of \( s \). This \( u \) gives \( Hs = \|s\|e_1 \). Noting \( H^{-1} = HT = H \), this also gives \( s = \|s\|He_1 \). The points in the reshaped polytope \( x_0, y_1, y_2, \ldots, y_n \) are given by

\[
y_i = x_0 + (-1)^\kappa h_m He_i \quad \forall i = 1, \ldots, n
\]

where \( e_i \) is the \( i \)th unit coordinate vector, and \( \kappa \) is the number of times the reshaping process has been performed. Hence each reshaped polytope has orthogonal edges from the vertex \( x_0 \) of equal length. This ensures both (7) and Assumption 1 hold. The \((-1)^\kappa\) factor means if the direction of the vector \( s = x_1 - x_0 \) remains fixed for several iterations, successive polytopes look along both each \( He_1 \) direction and its negative. This slightly increases the reliability of the method.

5.2. **Numerical Results and Discussion.** The algorithm was numerically tested on a variety of smooth and nonsmooth constrained problems. Problems 1–8 are from [12] and problems 12–15, 17, 18, and 20 are from [18]. One of these (LV4.7) was modified because its original form in [18] has one equality constraint, which means barrier methods are not applicable to it. This equality constraint was used to eliminate \( x_1 \) from LV4.7 yielding the test problem used herein. Problem 9 is the crescent problem from [3]. Problems 10, 11, and 16 are modifications of those in [20]. The objectives for problems 21–26 are as given in [13]. Problems 21 and 22 have the Broyden set of constraints as given in [13, 24]. P23 and P24 have the single constraint \( 1 - \|x\|_\infty \leq 0 \), and P25 and P26 have the sole constraint \( \|x\|_2 \leq \sqrt{n} \). The initial points for P21-P24 are as listed in [13], and the initial points for P25 and P26 are \( x_i = e \) for all \( i \).

The modified Zakharov problem (P19) is

\[
f = x_7 + x_8^2 + x_9^3 \quad \text{subject to} \quad \sum_{i=1}^{6} x_i^2 \leq x_7 \quad \text{and} \quad \frac{1}{2} \sum_{i=1}^{6} ix_i \leq x_8
\]

with the initial point \( x_i = i \) for \( i = 1, \ldots, 8 \), and optimal point \( x^* = 0 \). The modified Beale problem (P16) is

\[
f = \max\{x_3, x_4, x_5\} \quad \text{subject to} \quad (y_i - x_1 (1 - x_i^4))^2 \leq x_{2+i}, \quad i = 1, 2, 3
\]

where \( y_1 = 1.5, y_2 = 2.25 \) and \( y_3 = 2.625 \). The initial point was \( x_i = 1 \) for \( i = 1, \ldots, 5 \). The modified Variably Dimensioned problem (P11) is

\[
f = x_n + \sum_{i=1}^{n-1} (x_i - 1)^2 \quad \text{subject to} \quad \left( \sum_{i=1}^{n-1} i(x_i - 1) \right)^2 + \left( \sum_{i=1}^{n-1} i(x_i - 1) \right)^4 \leq x_n
\]
Table 1. Numerical results for the 26 test problems. Problems 1-11 are smooth, the rest are nonsmooth.

| function         | n | q | PENMECO | ORTHOMADS | PATTERNSEARCH |
|------------------|---|---|---------|-----------|---------------|
|                  |   |   | nf | nc(all) | E_f | max(c) | nf | nc | E_f | nf | E_f |
| 1 HS43 Rosen-Suzuki | 4 | 3 | 8186 | 8209 | – | 4e-9 | – | 4e-9 | 1328 | 2116 | 4e-4 | 2725 | 0.5 |
| 2 HS44            | 4 | 10 | 6556 | 6587 | 17 | 4e-9 | – | 1e-9 | 26 | 365 | 0 | 4577 | 2e-7 |
| 3 HS45            | 5 | 10 | 7896 | 7949 | 41 | 2e-9 | – | 4e-10 | 56 | 494 | 0 | 5359 | 0.43 |
| 4 HS72            | 4 | 10 | 3802 | 4401 | – | 2e-8 | – | 2e-15 | 5074 | 6411 | 3e-4 | 15760 | 5e-4 |
| 5 HS76            | 4 | 7 | 3194 | 3458 | 140 | 1e-10 | – | 2e-11 | 411 | 1156 | 0.01 | 76909 | 0.014 |
| 6 HS93            | 6 | 1 | 14709 | 22001 | – | 6e-3 | – | 1e-13 | 4643 | 6827 | 3e-3 | 1105 | 2e-3 |
| 7 HS100           | 7 | 4 | 45073 | 45163 | – | 4e-6 | – | 1e-3 | 2694 | 4792 | 5e-4 | 5205 | 5e-3 |
| 8 HS108           | 9 | 14 | 24982 | 26198 | 1058 | 2e-8 | – | 2e-8 | 5226 | 14443 | 0.4 | 13307 | 0.26 |
| 9 Audet-Dennis cres. | 10 | 2 | 52532 | 52921 | – | 3e-7 | – | 4e-9 | 22267 | 29561 | 1e-4 | 26004 | 0.84 |
| 10 mod. Jenrich-Sam. | 12 | 10 | 71702 | 80014 | – | 3.4 | – | 0.10 | 6336 | 33694 | 28.5 | 17145 | 29.4 |
| 11 mod. vardim    | 21 | 1 | 79667 | 80468 | 468 | 1e-7 | – | 1e-7 | 17997 | 80002 | 5e-7 | – | – |
| 12 LV4.1 madI     | 2 | 1 | 2421 | 2427 | – | 4e-9 | – | 3e-11 | 369 | 587 | 4e-9 | 635 | 3e-4 |
| 13 LV4.2 mad2     | 2 | 1 | 1924 | 1932 | – | 2e-9 | – | 2e-9 | 323 | 504 | 8e-6 | 1827 | 1e-7 |
| 14 LV4.3 mad4     | 2 | 1 | 3070 | 3076 | – | 4e-9 | – | 3e-11 | 4034 | 4639 | 4e-9 | 1995 | 1e-6 |
| 15 LV4.4 mad5     | 2 | 1 | 3363 | 3410 | – | 5e-9 | – | 9e-11 | 147 | 265 | 5e-5 | 695 | 8e-5 |
| 16 mod. Beale     | 5 | 3 | 27525 | 27591 | 15 | 1e-9 | – | 8e-10 | 5022 | 9087 | 0.03 | 37462 | 0.24 |
| 17 LV4.5 pentagon | 6 | 15 | 29246 | 29373 | 108 | 5e-6 | – | 1e-9 | 6948 | 12452 | 0.04 | 13472 | 0.16 |
| 18 LV4.7 mod. equil | 7 | 8 | 23739 | 23821 | – | 4e-5 | – | 0.03 | 53633 | 53701 | 1.9 | 12322 | 3e-3 |
| 19 mod. Zakharov  | 8 | 2 | 32469 | 32921 | 86 | 1e-10 | – | 5e-11 | 10982 | 22115 | 1e-4 | 47017 | 8e-10 |
| 20 LV4.8 Wong2    | 10 | 3 | 79589 | 80000 | – | 8e-5 | – | 3e-9 | 18436 | 24656 | 0.17 | 9420 | 0.2 |
| 21 maxQ10        | 10 | 8 | 79777 | 80001 | – | 2e-4 | – | 3e-5 | 11916 | 22834 | 1.8 | 14195 | 3.4 |
| 22 chainLQ       | 10 | 8 | 79861 | 80955 | 995 | 2e-7 | – | 5e-7 | 0 | 821 | infeas. | 59740 | 15 |
| 23 crescentI     | 20 | 1 | 79720 | 80000 | – | 8e-5 | – | 9e-9 | 23788 | 33759 | 0.78 | 52699 | 0.013 |
| 24 crescentII    | 20 | 1 | 79603 | 80029 | 29 | 8e-10 | – | 8e-10 | 41328 | 50365 | 2e-9 | 29710 | 6e-9 |
| 25 chainCB3I     | 20 | 1 | 79712 | 80000 | – | 9e-11 | – | 3e-10 | 44512 | 80007 | 1e-3 | 55562 | 0.032 |
| 26 chainCB3II    | 20 | 1 | 66735 | 66752 | – | 9e-6 | – | 2e-3 | 69767 | 70523 | 4e-4 | 71610 | 0.14 |
with the initial point \( x_i = (n - 1 - i)/(n - 1) \) for \( i = 1, \ldots, n - 1 \) and \( x_n = 0 \).

The modified Jennrich–Sampson problem (P10) has

\[
f = \sum_{i=3}^{12} x_i \quad \text{subject to} \quad \left( 2(1 + i) - e^{ix_1} - e^{ix_2} \right)^2 \leq x_{i+2} \quad i = 1, \ldots, 10
\]

with the initial point \( x_1 = 0.3, x_2 = 0.4 \) and

\[
x_{i+2} = 1.1 \left( 2(1 + i) - e^{0.3i} - e^{0.4i} \right)^2 \quad i = 1, \ldots, 10.
\]

Numerical comparisons are made with a variant \( \text{V-ORTHOMADS} \) of \( \text{ORTHOMADS} \) [4] programmed by this author. \( \text{V-ORTHOMADS} \) is an extreme barrier feasible point direct search method which asymptotically looks in all directions, and as such is a good comparator for \( \text{PENMECO} \). Several of the test problems have infeasible starting points. This was dealt with by both methods via a phase I which minimizes \( \max(c) \) until a point in \( F \) (for \( \text{V-ORTHOMADS} \)) or \( \Omega \) is found. This phase I solution was then used as the initial point when solving the actual problem.

Numerical results for the 26 test problems appear in Table 1, where the first 11 problems listed are smooth, and the remaining 15 have either a nonsmooth objective or nonsmooth constraints. Table 1’s legend is as follows. Columns headed 'nf' and 'nc' list the number of function and constraint evaluations used to solve the problem. For \( \text{PENMECO} \) the total number of constraint evaluations ('nc(all)') and the number used by the first phase to find a feasible point ('nc(I)') are listed separately. Columns headed 'Ef' list the relative error in \( f \), which is \( (f - f^*)/\max(1, |f^*|) \), where \( f^* \) is the optimal value of \( f \). The column headed 'max(c)' lists the maximum constraint value at the solution. This gives an indication of how close to the boundary of the feasible region \( \text{PENMECO} \) was able to approach.

Numerical tests show \( \text{PENMECO} \) is significantly more robust than \( \text{V-ORTHOMADS} \), at the expense of significantly more function evaluations. The differences in function counts can be partially attributed to the fact that \( \text{V-ORTHOMADS} \) has a line search whereas \( \text{PENMECO} \) does not. The discrepancy in robustness is partly due to the differences between the extreme and modified logarithmic barrier functions. The ability of the logarithmic barrier to initially keep iterates away from the feasible region’s boundary reduces the risk of being trapped on the boundary and unable to find a feasible descent direction. The effect of this can be seen in the relative numbers of objective and constraint function evaluations for each method. For \( \text{PENMECO} \), the number of objective and constraint evaluations is similar on most problems, showing that \( \text{PENMECO} \) largely moves through the interior towards the solution, only rarely encountering infeasible points. In contrast \( \text{V-ORTHOMADS} \) typically uses significantly fewer objective evaluations than constraint evaluations.

There were eight problems (P2, 3, 11–15, and 24) which both methods solved to an acceptable accuracy of \( E_f < 10^{-4} \), and two further problems (P9 and P19) on which \( \text{V-ORTHOMADS} \) almost achieved \( E_f < 10^{-4} \). The spectacular performance of \( \text{V-ORTHOMADS} \) on P2 and P3 is due to the fact that the initial points and solutions both lie on one of the coarse grids used by \( \text{V-ORTHOMADS} \) in its early iterations, allowing that method to step exactly to the solution quickly. As measured by each algorithm’s stopping conditions (see Table 1), \( \text{V-ORTHOMADS} \) was significantly faster on all but one of these ten problems. If the methods are stopped when \( E_f < 10^{-4} \) is first reached, the situation becomes somewhat more balanced, but
with v-ORTHOMADS still having the edge. Nevertheless, there are many problems which PENMECO can solve, but V-ORTHOMADS can not.

Table 1 also lists results for MATLAB’s PATTERNSEARCH algorithm. The default settings are used apart from the mesh and step tolerances were set at $10^{-10}$, and the constraint tolerance was set at $10^{-18}$. The former lines up with $h_{\text{min}} = 10^{-10}$ for PENMECO. The latter puts all three algorithms on the same footing with regard to constraint violations. The polling method used was ‘MADSPositiveBasis2N’, which is a Mesh Adaptive Direct Search [2, 4] method that uses an augmented Lagrangian to handle the non-linear constraints [27]. In addition PATTERNSEARCH used the inbuilt Nelder Mead search step ‘searchneldermead’. Since PATTERNSEARCH is a stochastic method, all results for PATTERNSEARCH that are presented in Table 1 are 30 run averages.

The use of an augmented Lagrangian by PATTERNSEARCH means that it can use infeasible iterates, in contrast to the other two methods. The nature of many test problems change depending on whether or not $f$ can be evaluated at infeasible points, which makes comparison between PATTERNSEARCH and the other two methods less straightforward. Disregarding this issue, Table 1 shows PENMECO solves more problems than PATTERNSEARCH, but is slower.

Overall, the performance of PENMECO on the smooth and nonsmooth test problems is similar, with only two smooth and one nonsmooth not solved to an acceptable accuracy. Although the convergence theory does not extend to the nonsmooth problems, it shows the method has a significant capability on such problems.

5.3. Comparison on the Cones Test Functions. PENMECO and V-ORTHOMADS were also compared on the cones set of randomly generated test functions from [24]. Constructing a cones test problem starts with a nonsmooth convex objective and one convex hyperspherical constraint which is active at the solution. All other constraints are cone shaped with the solution at each cone’s apex. The region which violates a conical constraint has a semi-angle of $\pi/3$, and randomly oriented axis. The objective and constraints are then twisted using a random cubic transformation which destroys convexity, but does not create any other local minimizers. In high dimensions the region excluded by the conical constraints is relatively small, and V-ORTHOMADS is largely unaffected by them because it ignores constraints until it generates a point which actually violates a constraint. In contrast, any logarithmic barrier method will be affected by the proximity of the conical constraints when near the solution, even if no point violating these constraints is ever generated.

The results for the cones test set appear in Figure 2, which plots the number of randomly generated problems solved by each method against the number of simplex gradients $N$, for problems of dimension $n = 12, 18, \text{and} 24$, each with 11 constraints. Initially V-ORTHOMADS finds solutions in approximately half the number of function evaluations as PENMECO, but is eventually overtaken as $N$ increases. On most problems V-ORTHOMADS does not halt, but rather makes negligible progress until the maximum value of $N$ is reached. In contrast, PENMECO has a sufficient descent condition, which limits this behaviour. If the maximum value of $N$ is raised to 25,000 then PENMECO halts on all problems, with 0, 2, and 13 fails for $n = 12, 18, \text{and} 24$ respectively. V-ORTHOMADS was still going on 4, 15, and 22 problems, and for the problems on which it halted, it chalked up 13, 27, and 50 fails for $n = 12, 18, \text{and} 24$ respectively.
Figure 2. Data profiles for penmeco and v-orthomads on 100 randomly generated cones test functions from [24], in 12, 18, and 24 dimensions with 11 constraints. Each curve gives the number of problems solved to an accuracy $E_f < 10^{-4}$ within $N$ simplex gradients, which is $N(n+1)$ evaluations of $f$.

6. Conclusion. A barrier function based variant of the Nelder Mead method for nonlinearly constrained optimization has been described. Asymptotic convergence is guaranteed under mild conditions which include strict differentiability at each solution found, but do not include continuous differentiability. Numerical tests show the method performs well in low to moderate dimensions, and that it can be effective on some nonsmooth problems outside the scope of its convergence theory.

REFERENCES
[1] C. Audet and J. E. Dennis Jr., Analysis of generalized pattern searches, SIAM J. Optim., 13 (2003), 889–903.
[2] C. Audet and J. E. Dennis Jr., Mesh adaptive direct search algorithms for constrained optimization, SIAM J. Optim., 17 (2006), 188–217.
[3] C. Audet and J. E. Dennis Jr., A progressive barrier for derivative free nonlinear programming, SIAM J. Optim., 20 (2009), 445–472.
[4] M. A. Abramson, C. Audet, J. E. Dennis Jr. and S. Le Digabel, orthoMADS: a deterministic MADS instance with orthogonal directions, SIAM J. Optim., 20 (2009), 948-966.
[5] F. H. Clarke, Optimization and Non-smooth Analysis, SIAM classics in applied mathematics, New York, 1990.
[6] I. D. Coope and C. J. Price, Frame based methods for unconstrained optimization, J. Optim. Theory and Appl., 107 (2000), 261–274.
[7] I. D. Coope and C. J. Price, On the convergence of grid based methods for unconstrained optimization, SIAM J. Optim., 11 (2001), 859–869.
[8] I. D. Coope and C. J. Price, Positive bases in numerical optimization, *Comput. Optim. and Appl.*, 21 (2002), 169–175.

[9] C. Davis, Theory of positive linear dependence, *Amer. J. Math.*, 76 (1954), 733–746.

[10] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Wiley, New York, 1968.

[11] F. Gao and L. Han, Implementing the Nelder Mead simplex algorithm with adaptive parameters, *Comput. Optim. Appl.*, 51 (2012), 259–277.

[12] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Springer, Berlin, 1981.

[13] N. Karmitsa, *Test Problems for Large-Scale Nonsmooth Minimization*, Department of Mathematical Information Technology Report B. 4/2007, University of Jyväskylä, Finland, 2007.

[14] C. T. Kelley, Detection and remediation of stagnation in the Nelder-Mead algorithm using a sufficient decrease condition, *SIAM J. Optim.*, 10 (1999), 43–55.

[15] T. G. Kolda, R. M. Lewis and V. Torczon, Optimization by direct search: New perspectives on some classical and modern methods, *SIAM Rev.*, 45 (2003), 385–482.

[16] J. C. Lagarias, J. A. Reeds, M. H. Wright and P. E. Wright, Convergence properties of the Nelder-Mead simplex method in low dimensions, *SIAM J. Optim.*, 9 (1999), 112–147.

[17] J. C. Lagarias, B. Poonen and M. H. Wright, Convergence of the restricted Nelder Mead algorithm in two dimensions, *SIAM J. Optim.*, 22 (2012), 501–532.

[18] L. Lukšan and J. Vlček, *Test problems for nonsmooth unconstrained and linearly constrained optimization*, Tech. Report 798, Prague: Institute of Computer Science, Academy of Sciences of the Czech Republic, 2000.

[19] K. I. M. McKinnon, A simplex method for function minimization, *SIAM J. Optim.*, 9 (1999), 148–158.

[20] J. J. More, B. S. Garbow and K. E. Hillstrom, Testing unconstrained optimization software, *ACM Trans. Math. Software*, 7 (1981), 17–41.

[21] L. Nazareth and P. Tseng, Gilding the lily: a variant of the Nelder Mead algorithm based on golden-section search, *Comput. Optim. Appl.*, 22 (2002), 133–144.

[22] J. A. Nelder and R. Mead, A simplex method for function minimization, *Computer J.*, 7 (1965), 308–313.

[23] M. J. D. Powell, Direct search algorithms for optimization calculations, *Acta Numerica*, 7 (1998), 287–336.

[24] C. J. Price. Direct search nonsmooth constrained optimization via rounded ℓ1 penalty functions, *Optim. Methods & Software*, (2020), DOI: 10.1080/10556788.2020.1746961

[25] C. J. Price and I. D. Coope, Frames and grids in unconstrained and linearly constrained optimization: a non-smooth approach, *SIAM J. Optim.*, 14 (2003), 415–438.

[26] C. J. Price, I. D. Coope and D. Byatt, A convergent variant of the Nelder Mead algorithm, *J. Optim. Theory and Appl.*, 113 (2002), 5–19.

[27] The MathWorks Inc., *Global Optimization User’s Toolbox Guide R2020b*, www.mathworks.com.

[28] V. Torczon, On the convergence of pattern search algorithms, *SIAM J. Optim.*, 7 (1997), 1–25.

[29] P. Tseng, Fortified-descent simplicial search method: a general approach, *SIAM J. Optim.*, 10 (1999), 269–288.

[30] M. H. Wright, Direct search methods: once scorned now respectable, in Proceedings of the 1995 Dundee Biennial Conference in Numerical Analysis Addison-Wesley, Reading, MA and Longman, Harlow, UK, 1996.

[31] W.-C. Yu, Positive basis and a class of direct search techniques, *Scientia Sinica (special issue)*, 1 (1979).

[32] W. C. Yu and Y.-X. Li, A direct search method by the local positive basis for the nonlinearly constrained optimization, *Chinese Annals of Math.*, 2 (1981), 269–280.

Received October 2020; 1st revision November 2020; Final revision November 2020.

E-mail address: chrisj.price@canterbury.ac.nz