MNH: A Derivative-Free Optimization Algorithm Using Minimal Norm Hessians

Stefan M. Wild[*]

January 13, 2008

Abstract

We introduce MNH, a new algorithm for unconstrained optimization when derivatives are unavailable, primarily targeting applications that require running computationally expensive deterministic simulations. MNH relies on a trust-region framework with an underdetermined quadratic model that interpolates the function at a set of data points. We show how to construct this interpolation set to yield computationally stable parameters for the model and, in doing so, obtain an algorithm which converges to first-order critical points. Preliminary results are encouraging and show that MNH makes effective use of the points evaluated in the course of the optimization.

1 Introduction

In this paper we address unconstrained optimization,

$$\min \{f(x) : x \in \mathbb{R}^n\}, \tag{1.1}$$

of a function whose derivatives are unavailable. Our work is motivated by functions that are computationally expensive to evaluate, usually as a result of the need to run some underlying complex simulation model. These simulations often provide the user solely with the simulation output, creating the need for a *derivative-free* optimization algorithm. Examples of derivative-free optimization applied to these types of problems in electrical, environmental, and biomedical engineering can be found in [6, 8, 13].

When f is computationally expensive, a user is typically constrained by a computational budget that limits the number of function evaluations available to the optimization algorithm. We view the data gained from each function evaluation as contributing to a *bank* of insight into the function. As the optimization is carried out, more points are evaluated and this bank will grow. How to most effectively manage the data contained in the bank is a central driving force behind this paper.

Our approach is inspired by the recent work of Powell [10, 11] using quadratic models interpolating fewer than a quadratic (in the dimension n) number of points. This strategy allows the underlying optimization to begin sooner and make more rapid progress in fewer function evaluations. These models are assumed to locally approximate the function while being computationally inexpensive to evaluate and optimize over.

In this paper we introduce a new algorithm, MNH, that contributes two new features. First, unlike previous algorithms [8, 11], which were driven by a desire to keep linear algebraic overhead to $\mathcal{O}(n^3)$ operations per iteration, our algorithm views overhead as negligible relative to the expense of function evaluation. This allows greater flexibility in using points from the bank.

Second, our models are formed from interpolation sets in a computationally stable manner which guarantees that the models are well-behaved. In fact, both our model and its gradient are able to approximate the function and its gradient arbitrarily well. Consequently, the recent convergence

[*] School of Operations Research & Information Engineering, Cornell University, Ithaca, New York 14853. Research supported by a DOE Computational Science Graduate Fellowship under grant number DE-FG02-97ER25308

result of Conn, Scheinberg and Vicente [3] guarantees that our algorithm will converge to first-order critical points.

Encouraged by preliminary results, we hope that this convergence result and our way of using points from the bank will yield a theoretically sound algorithm that is both relatively simple and works well in practice.

This paper is organized as follows. In Section 2 we review derivative-free trust-region algorithms. Section 3 introduces the special quadratic models employed by our algorithm. The MNH algorithm is discussed in Section 4 and preliminary numerical findings are presented in Section 5.

2 Derivative-Free Trust-Region Methods

Our algorithm is built upon a trust-region framework that we now review. A trust-region method is an iterative method that optimizes over a surrogate model m_k assumed to approximate f within a neighborhood of the current iterate x_k, the *trust-region*

$$\mathcal{B}_k = \{x \in \mathbb{R}^n : \|x - x_k\| \leq \Delta_k\},$$

for a radius $\Delta_k > 0$. New candidate points are obtained by solving the subproblem

$$\min \{m_k(x_k + s) : x_k + s \in \mathcal{B}_k\}. \tag{2.1}$$

In fact, it suffices to only solve (2.1) approximately, provided that the resulting step s_k satisfies a sufficient decrease condition. After the function is evaluated at $x_k + s_k$, the pair (x_k, Δ_k) is updated according to the ratio of actual to predicted decrease,

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)},$$

ρ_k values close to 1 corresponding to good model prediction.

Given an initial point x_0 and a maximum radius $\Delta_{\max}$, the design of the trust-region algorithm ensures that f is only sampled within the relaxed level set

$$\mathcal{L}(x_0) = \{y \in \mathbb{R}^n : \|x - y\| \leq \Delta_{\max} \text{ for some } x \text{ with } f(x) \leq f(x_0)\}.$$

A quadratic model,

$$m_k(x_k + s) = f(x_k) + g_k^T s + \frac{1}{2} s^T H_k s, \tag{2.2}$$

is typically employed, with $g_k = \nabla f(x_k)$ and $H_k = \nabla^2 f(x_k)$ when these derivatives are available. The quadratic model in (2.2) is attractive because global solutions to the subproblem in (2.1) can then be efficiently computed. When the gradient ∇f is exactly available, global convergence to local minima is possible under mild assumptions. Full treatment is given in [1].

When only function values are available, the model m_k can be obtained by interpolating the function at a set of distinct data points $\mathcal{Y} = \{y_1 = 0, y_2, \ldots, y_{|\mathcal{Y}|}\} \subset \mathbb{R}^n$:

$$m_k(x_k + y_j) = f(x_k + y_j) \qquad \text{for all } y_j \in \mathcal{Y}. \tag{2.3}$$

This approach was taken with both quadratic [2, 9] and radial basis function (RBF) models [8, 13].

A primary concern in the study of interpolation model-based derivative-free methods is the quality of the model within $\mathcal{B}_k$. In [4], Taylor-like error bounds are established based on the geometry of the interpolation set $\mathcal{Y}$. These results motivate a class of so-called *fully linear* models for approximating functions that are reasonably smooth. In particular, we will assume that $f \in C^1[\Omega]$ for some open $\Omega \supset \mathcal{L}(x_0)$, ∇f is Lipschitz continuous on $\mathcal{L}(x_0)$, and f is bounded on $\mathcal{L}(x_0)$.

Definition 1. *For fixed $\kappa_f, \kappa_g > 0$ and $\mathcal{B} = \{x \in \mathbb{R}^n : \|x - x_k\| \leq \Delta\}$, a model $m \in C^1[\Omega]$ is said to be* <u>*fully linear (f.l.) on $\mathcal{B}$*</u> *if for all $x \in \mathcal{B}$:*

$$\begin{aligned} |f(x) - m(x)| &\leq \kappa_f \Delta^2, && (2.4) \\ \|\nabla f(x) - \nabla m(x)\| &\leq \kappa_g \Delta. && (2.5) \end{aligned}$$

The two conditions in Definition 1 ensure that approximations to the function and its gradient can achieve any desired degree of precision within a small enough neighborhood of x_k. Provided that m_k can be made fully linear (for fixed κ_f and κ_g) in finitely many steps, Algorithm 2.1 was recently shown to be globally convergent to a stationary point $\nabla f(x_*) = 0$, given an appropriate termination test [3].

```
Input $x_0 \in \mathbb{R}^n$, $0 < \Delta_0 \leq \Delta_{\max}$, $m_0$, $0 \leq \eta_0 \leq \eta_1 < 1$ $(\eta_1 \neq 0)$, $0 < \gamma_0 < 1 < \gamma_1$, $\epsilon_g > 0$.
Iteration $k \geq 0$:
    1. If $\|\nabla m_k\| \leq \epsilon_g$, test for termination.
    2. Solve $\min\{m_k(x_k + s) : \|s\| \leq \Delta_k\}$ for $s_k$ and set $x_+ = x_k + s_k$.
    3. Evaluate $f(x_+)$ and $\rho_k = \frac{f(x_k) - f(x_+)}{m_k(x_k) - m_k(x_+)}$ and update the center:
           $x_{k+1} = \begin{cases} x_+ & \text{if } \rho_k \geq \eta_1 \\ x_+ & \text{if } \eta_1 > \rho_k > \eta_0 \text{ and } m_k \text{ f.l. on } \mathcal{B}_k \\ x_k & \text{else.} \end{cases}$
    4. If $\rho_k < \eta_1$ and $m_k$ not f.l. on $\mathcal{B}_k$, improve the model by evaluating at a model-improving
       point. Hence or otherwise update the model to $m_{k+1}$.
    5. Update the trust-region radius
           $\Delta_{k+1} = \begin{cases} \min\{\gamma_1 \Delta_k, \Delta_{\max}\} & \text{if } \rho_k \geq \eta_1 \\ \Delta_k & \text{if } \rho_k < \eta_1 \text{ and } m_k \text{ not f.l. on } \mathcal{B}_k \\ \gamma_0 \Delta_k & \text{if } \rho_k < \eta_1 \text{ and } m_k \text{ f.l. on } \mathcal{B}_k. \end{cases}$
```

Algorithm 2.1: Basic first-order derivative-free trust-region algorithm.

3 Minimum Norm Quadratic Interpolation Models

In this paper we are interested in quadratic models of the form (2.2) with the parameters g_k and H_k such that m_k satisfies the interpolation conditions (2.3). To this end, we define

$$\begin{aligned} \mu(x) &= [1, \chi_1, \cdots, \chi_n], \\ \nu(x) &= \left[\frac{\chi_1^2}{2}, \cdots, \frac{\chi_n^2}{2}, \frac{\chi_1 \chi_2}{\sqrt{2}}, \cdots, \frac{\chi_{n-1}\chi_n}{\sqrt{2}}\right], \end{aligned}$$

where χ_i denotes the ith component of the argument $x \in \mathbb{R}^n$. When taken together, $[\mu(x), \nu(x)]$ forms a basis for the linear space of quadratics in n variables, $\mathcal{Q}^n$. Thus any quadratic $m_k \in \mathcal{Q}^n$ can be written as

$$m_k(x - x_k) = \alpha^T \mu(x - x_k) + \beta^T \nu(x - x_k), \tag{3.1}$$

for coefficients $\alpha \in \mathbb{R}^{n+1}$ and $\beta \in \mathbb{R}^{n(n+1)/2}$. We note that any bijection of this basis would also yield a quadratic and so the form of the quadratic model in (3.1) may seem unusual at first glance. We propose to use this particular form of model because it lends itself well to our solution procedure.

Abusing notation, we let f denote the vector of function values so that (2.3) can be written as

$$\left[\begin{array}{c} M_{\mathcal{Y}} \\ N_{\mathcal{Y}} \end{array} \right]^T \left[\begin{array}{c} \alpha \\ \beta \end{array} \right] = f, \tag{3.2}$$

where we define $M_{\mathcal{Y}} \in \mathbb{R}^{n+1\times|\mathcal{Y}|}$ and $N_{\mathcal{Y}} \in \mathbb{R}^{n(n+1)/2\times|\mathcal{Y}|}$, by $M_{i,j} = \mu_i(y_j)$ and $N_{i,j} = \nu_i(y_j)$, respectively. We explicitly note the dependence of these matrices on the interpolation set $\mathcal{Y}$.

The interpolation problem in (2.3) for multivariate quadratics is significantly more difficult than its univariate counterpart [12]. These points must satisfy additional geometric conditions that are summarized in the following Lemma, which follows immediately from the fact that $[\mu(x), \nu(x)]$ form a basis for $\mathcal{Q}^n$.

Lemma 3.1. *The following are equivalent:*

1. *For any $f \in \mathbb{R}^{|\mathcal{Y}|}$, there exists $m_k \in \mathcal{Q}^n$ satisfying (2.3).*
2. *$\{[\mu(y_j),\, \nu(y_j)]\}_{j=1}^{|\mathcal{Y}|}$ is linearly independent.*
3. *$dim\{q \in Q^n : q(x_k + y_i) = 0\, \forall y_j \in \mathcal{Y}\} = \frac{(n+1)(n+2)}{2} - |\mathcal{Y}|$.*

The third condition in Lemma 3.1 reveals that these conditions are geometric, requiring that the subspace of quadratics disappearing at all of the data points be of sufficiently low dimension. For example, this prevents interpolation of arbitrary f values using 6 points lying on a circle in $\mathbb{R}^2$.

Lemma 3.1 implies that quadratic interpolation is only feasible for arbitrary right hand side values if $[M_{\mathcal{Y}}^T,\, N_{\mathcal{Y}}^T]$ is full row rank. Further, this interpolation is only unique if $|\mathcal{Y}| = \frac{(n+1)(n+2)}{2}$ (the dimension of quadratics in $\mathbb{R}^n$) and $[M_{\mathcal{Y}}^T,\, N_{\mathcal{Y}}^T]$ is nonsingular.

When $|\mathcal{Y}| < \frac{(n+1)(n+2)}{2}$, and $[M_{\mathcal{Y}}^T,\, N_{\mathcal{Y}}^T]$ is full rank, the interpolation problem (3.2) will have an infinite number of solutions. In this paper we will focus on solutions to (3.2) that are of minimum norm with respect to the vector β. Hence we require the solution (α, β) of

$$\min \left\{ \frac{1}{2}\|\beta\|^2 : M_{\mathcal{Y}}^T \alpha + N_{\mathcal{Y}}^T \beta = f \right\}. \tag{3.3}$$

This solution is of interest because it represents the quadratic whose Hessian matrix is of minimum Frobenius norm since $\|\beta\| = \|\nabla^2_{x,x} m(x)\|_F$. While other “minimal norm” quadratics could be found, we are drawn to those with Hessians of minimal norm because the resulting solution procedure will have a natural tie-in to fully linear models.

The KKT conditions for (3.3) can be written as

$$\left[\begin{array}{cc} N_{\mathcal{Y}}^T N_{\mathcal{Y}} & M_{\mathcal{Y}}^T \\ M_{\mathcal{Y}} & 0 \end{array} \right] \left[\begin{array}{c} \lambda \\ \alpha \end{array} \right] = \left[\begin{array}{c} f \\ 0 \end{array} \right], \tag{3.4}$$

with $\beta = N_{\mathcal{Y}}\lambda$. We solve this saddle point problem with a null space method by letting Z be an orthogonal basis for the null space $\mathcal{N}(M_{\mathcal{Y}})$ and $QR = M_{\mathcal{Y}}^T$ be a QR factorization. Since λ must belong to $\mathcal{N}(M_{\mathcal{Y}})$, we write $\lambda = Z\omega$ for $\omega \in \mathbb{R}^{|\mathcal{Y}|-n-1}$ so that (3.4) reduces to the $|\mathcal{Y}|$ equations:

$$\begin{aligned} Z^T N_{\mathcal{Y}}^T N_{\mathcal{Y}} Z\omega &= Z^T f & (3.5)\\ R\alpha &= Q^T(f - N_{\mathcal{Y}}^T N_{\mathcal{Y}} Z\omega), & (3.6) \end{aligned}$$

with $\beta = N_{\mathcal{Y}} Z\omega$.

The following Theorem establishes that the quadratic program (3.3) will yield a unique solution given geometric conditions on $\mathcal{Y}$.

Theorem 3.2. *For $n \geq 2$, if:*

(Y1) rank($M_{\mathcal{Y}}$) $= n+1$, and

(Y2) $Z^T N_{\mathcal{Y}}^T N_{\mathcal{Y}} Z$ is positive definite,

then, for any $f \in \mathbb{R}^{|\mathcal{Y}|}$, there exists a unique solution (α, β) to the quadratic program (3.3).

Proof. $Z^T N_{\mathcal{Y}}^T N_{\mathcal{Y}} Z$ is positive definite if and only if $N_{\mathcal{Y}} Z$ is full rank. Since $n \geq 2$, $N_{\mathcal{Y}} Z$ is full rank if and only if $\mathcal{N}(N_{\mathcal{Y}} Z) = \{0\}$. Lastly, since Z is a basis for $\mathcal{N}(M_{\mathcal{Y}})$, this is equivalent to $\mathcal{N}(N_{\mathcal{Y}}) \cap \mathcal{N}(M_{\mathcal{Y}}) = \{0\}$, which says that $[M_{\mathcal{Y}}^T \, N_{\mathcal{Y}}^T]$ is full rank. By Lemma 3.1, we then have that the feasible region of (3.3) is nonempty.

Since (3.3) is a convex (in β) quadratic program whose feasible region is nonempty, both β and the Lagrange multipliers λ associated with the constraints are unique [5]. Finally, we note that the coefficients α are then also uniquely determined from $M_{\mathcal{Y}}^T \alpha = f - N_{\mathcal{Y}}^T \beta$ since $M_{\mathcal{Y}}^T$ is full rank. □

If $Z^T N_{\mathcal{Y}}^T N_{\mathcal{Y}} Z$ is positive definite, it admits the Cholesky factorization

$$Z^T N_{\mathcal{Y}}^T N_{\mathcal{Y}} Z = L L^T,$$

for a nonsingular lower triangular L. Since Z is orthogonal we have the bound

$$\|\lambda\| = \|Z\omega\| \left\|Z L^{-T} L^{-1} Z^T f\right\| \leq \left\|L^{-1}\right\|^2 \|f\| = \frac{\|f\|}{\sigma_{\min}^2(L)}, \tag{3.7}$$

where $\sigma_{\min}(L)$ is the smallest singular value of L. This relationship will allow us to bound the coefficients $\beta = N_{\mathcal{Y}} \lambda$, and hence bound the Hessians of the model m.

4 The MNH Algorithm

Theorem 3.2 offers a constructive way of obtaining an interpolation set $\mathcal{Y}$ that uniquely defines an underdetermined quadratic model whose Hessian is of minimum norm. We first collect $n+1$ affinely independent points and then add more points while keeping $\sigma_{\min}(L)$ bounded from zero.

We will always keep $y_1 = 0$ in the set $\mathcal{Y}$ to enforce interpolation at the current center. Thus we only need to find n linearly independent points $y_2, \ldots, y_{n+1}$. The resulting points will serve a secondary purpose of providing approximation guarantees for the model. This is formally stated in the following generalization of similar Taylor-like error bounds found in [4].

Theorem 4.1. *Suppose that f and m are continuously differentiable in $\mathcal{B} = \{x : \|x - x_k\| \leq \Delta\}$ and that ∇f and ∇m are Lipschitz continuous in $\mathcal{B}$ with Lipschitz constants γ_f and γ_m, respectively. Further suppose that m satisfies the interpolation conditions in* (2.3) *at a set of points $\mathcal{Y} = \{y_1 = 0, y_2, \ldots, y_{n+1}\} \subseteq \mathcal{B} - x_k$ such that $\left\|[y_2, \cdots, y_{n+1}]^{-1}\right\| \leq \frac{\Lambda_Y}{\Delta}$. Then for any $x \in \mathcal{B}$:*

1. $|m(x) - f(x)| \leq \sqrt{n}\left(\gamma_f + \gamma_m\right)\left(\frac{5}{2}\Lambda_Y + \frac{1}{2}\right)\Delta^2$, *and*
2. $\|\nabla m(x) - \nabla f(x)\| \leq \frac{5}{2}\sqrt{n}\Lambda_Y\left(\gamma_f + \gamma_m\right)\Delta$.

Proved in [14], Theorem 4.1 says that if a model with a Lipschitz continuous gradient interpolates a function on a sufficiently affinely independent set of nearby points, there exist constants $\kappa_f, \kappa_g > 0$ independent of Δ such that conditions (2.4) and (2.5) are satisfied. In our case, the model m will be twice continuously differentiable and hence the following Lemma yields a Lipschitz constant.

Lemma 4.2. *For the model m defined in* (3.1), *$\nabla m(x)$ is $\|\beta\|$-Lipschitz continuous on $\mathbb{R}^n$.*

Proof. Since m is a quadratic, $\nabla m(x) - \nabla m(y) = \nabla^2 m(x)(x-y)$ for all $x, y \in \mathbb{R}^n$. Recalling that $\|\nabla^2 m(x)\|_F = \|\beta\|$ we have

$$\|\nabla m(x) - \nabla m(y)\| \leq \|\nabla^2 m(x)\| \|x-y\| \leq \|\nabla^2 m(x)\|_F \|x-y\| = \|\beta\| \|x-y\|,$$

establishing the result. □

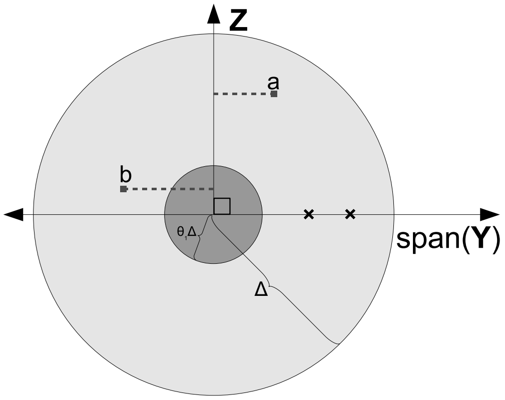

Figure 4.1: Obtaining sufficiently affinely independent points.

4.1 Finding Affinely Independent Points

We now show that we can obtain n points such that $\| [y_2, \cdots, y_{n+1}]^{-1} \|$ is bounded by a quantity of the form $\frac{\Lambda_Y}{\Delta}$ as required in Theorem 4.1. We ensure this by working with a QR factorization of the normalized points $Y = \left[\frac{y_2}{\Delta}, \cdots, \frac{y_{n+1}}{\Delta}\right]$. If we require that these points satisfy $\left\|\frac{y_j}{\Delta}\right\| \leq 1$, and that the resulting pivots satisfy $|R_{j,j}| \geq \theta_1 > 0$, then it is straightforward to show that $\left\|Y^{-1}\right\| \leq \Lambda_Y$ for a constant Λ_Y depending only on n and θ_1 (eg., Lemma 4.2 in [14]).

Figure 4.1 illustrates our procedure graphically. From our bank of points at which the function has been evaluated, we examine all those within Δ of the current center. These points are iteratively added to $\mathcal{Y}$ provided that their projection onto the current null space $Z = \mathcal{N}([y_2, \cdots y_{|\mathcal{Y}|}])$ is at least of magnitude $\theta_1\Delta$. In Figure 4.1 the **x**'s denote the current points, while the projections of two available candidate points, **a** and **b**, show that only **a** would be added to $\mathcal{Y}$.

In practice, we work with an enlarged region with radius $\Delta = \theta_0\Delta_k$ ($\theta_0 \geq 1$), to ensure the availability of some previously evaluated points. Our procedure is detailed formally in Algorithm 4.1.

This procedure also guarantees that such an interpolation set can be constructed for any value of the constant $\theta_1 \leq 1$. In particular, if Z is an orthogonal basis for $\mathcal{N}([y_2, \cdots y_{|\mathcal{Y}|}])$, its columns are directions that result in unit pivots, $|R_{j,j}| = 1$. We call $\pm\Delta z_j$ *model-improving points* because they can be included in $\mathcal{Y}$ to make m fully linear on $\mathcal{B}$.

Upon termination of Algorithm 4.1, the set $\mathcal{Y}$ either contains $n+1$ points (including the initial point 0) which certifies that the model is fully linear on a ball of radius $\theta_0\Delta_k$, or there will be nontrivial model-improving directions in Z which can be evaluated to obtain such a model.

While the trust-region framework in Algorithm 2.1 does not prescribe a fully linear model at each iteration, Theorem 3.2 requires that $\mathcal{Y}$ include $n+1$ affinely independent points. Hence, if

```
0. Input $\mathcal{D} = \{d_1, \dots, d_{|\mathcal{D}|}\} \subset \mathbb{R}^n$, constants $\theta_0 \geq 1$, $\theta_1 \in (0, \theta_0^{-1}]$, $\Delta_k \in (0, \Delta_{\max}]$.
1. Initialize $\mathcal{Y} = \{y_1 = 0\}$, $Z = I_n$.
2. For all $d_j \in \mathcal{D}$ such that $\|d_j\| \leq \theta_0\Delta_k$:
    If $\left|\mathrm{proj}_Z\left(\frac{1}{\theta_0\Delta_k} d_j\right)\right| \geq \theta_1$:
        $\mathcal{Y} \leftarrow \mathcal{Y} \cup \{d_j\}$,
        Update $Z$ to be an orthonormal basis for $\mathcal{N}\left([y_2 \cdots y_{|\mathcal{Y}|}]\right)$.
```

Algorithm 4.1: AffPoints($\mathcal{D}, \theta_0, \theta_1, \Delta_k$) obtains sufficiently affinely independent points.

```
0. Input $\mathcal{Y}$, $\mathcal{D} = \{d_1, \ldots, d_{|\mathcal{D}|}\} \subset \mathbb{R}^n$, constants $\theta_0 \geq 1$, $\theta_2 > 0$, $\Delta_k \in (0, \Delta_{\max}]$.
1. Initialize $QR = M_{\mathcal{Y}}^T$, $Z = \emptyset$.
2. For all $d_j \in \mathcal{D}\backslash\mathcal{Y}$ such that $\|d_j\| \leq \theta_0\Delta_k$:
     Compute $\tilde{N}_{\mathcal{Y}}\tilde{Z}$ as in (4.1).
     If $\sigma_{\min}\left(\tilde{N}_{\mathcal{Y}}\tilde{Z}\right) \geq \theta_2$:
        $\mathcal{Y} \leftarrow \mathcal{Y} \cup \{d_j\}$,
        Update $Z = \tilde{Z}$ and $N_{\mathcal{Y}} = \tilde{N}_{\mathcal{Y}}$.
```

Algorithm 4.2: MorePoints$(\mathcal{D}, \theta_0, \theta_2, \Delta_k)$ adds additional points to $\mathcal{Y}$.

a model is not fully linear, we will rerun Algorithm 4.1 with a larger θ_0. This has the effect of searching for points in the bank within a larger region. If still an insufficient number of points are available, the directions in the resulting Z must be evaluated.

4.2 Adding More Points

After running Algorithm 4.1, and possibly evaluating f at additional points, the interpolation set $\mathcal{Y}$ consists of $n+1$ sufficiently affinely independent points. If no other points are added to $\mathcal{Y}$, we will have $\beta = 0$ and hence m_k would be a linear model. Adding additional points to $\mathcal{Y}$ will not affect the first condition (Y1) of Theorem 3.2, thus our goal is to add more points from the bank to $\mathcal{Y}$ while ensuring the second condition (Y2) is satisfied and (3.6) remains well-conditioned.

We now consider what happens when $d \in \mathbb{R}^n$ is added to the interpolation set $\mathcal{Y}$ and denote the resulting basis matrices by $\tilde{M}_{\mathcal{Y}}$ and $\tilde{N}_{\mathcal{Y}}$:

$$\tilde{M}_{\mathcal{Y}} = \left[\begin{array}{cc} M_{\mathcal{Y}} & \mu(d) \end{array}\right], \qquad \tilde{N}_{\mathcal{Y}} = \left[\begin{array}{cc} N_{\mathcal{Y}} & \nu(d) \end{array}\right].$$

By applying $n+1$ Givens rotations to the full QR factorization of $M_{\mathcal{Y}}^T$, we obtain an orthogonal basis for $\mathcal{N}(M_{\mathcal{Y}})$ of the form:

$$\tilde{Z} = \left[\begin{array}{cc} Z & Q\tilde{g} \\ 0 & \hat{g} \end{array}\right],$$

where Z is any orthogonal basis for $\mathcal{N}(M_{\mathcal{Y}})$. Hence, $\tilde{N}_{\mathcal{Y}}\tilde{Z}$ consists of the previous factors $N_{\mathcal{Y}}Z$ and one additional column:

$$\tilde{N}_{\mathcal{Y}}\tilde{Z} = \left[\begin{array}{cc} N_{\mathcal{Y}}Z & N_{\mathcal{Y}}Q\tilde{g} + \hat{g}\nu(d) \end{array}\right]. \tag{4.1}$$

While beyond the scope of this paper, we note that (4.1) suggests that the resulting Cholesky factorization $\tilde{L}\tilde{L}^T = (\tilde{N}_{\mathcal{Y}}\tilde{Z})^T\tilde{N}_{\mathcal{Y}}\tilde{Z}$ could be updated using the previous factorization. Here we require only a mechanism for bounding $\sigma_{\min}(L)$ for use in the bound (3.7). Since $\sigma_{\min}(N_{\mathcal{Y}}Z) = \sigma_{\min}(L)$, it will suffice to enforce $\sigma_{\min}(N_{\mathcal{Y}}Z) \geq \theta_2$ for a constant $\theta_2 > 0$.

The bound on λ in (3.7) will be used to bound $\|\beta\| = \|N_{\mathcal{Y}}\lambda\|$, which from Lemma 4.2, serves as a Lipschitz constant for m_k, justifying our use of fully linear models. By the discussion in Section 3, the interpolation set must always obey the bound $|\mathcal{Y}| = \frac{(n+1)(n+2)}{2}$ since otherwise $N_{\mathcal{Y}}Z$ would be rank-deficient. Hence in order to bound $\|N_{\mathcal{Y}}\|$, it suffices to keep the points in $\mathcal{Y}$ within a bounded region. We will again assume that this region is contained in a ball of radius $\theta_0\Delta_k$ for some $\theta_0 \geq 1$. Algorithm 4.2 then specifies the resulting subroutine.

By Theorem 3.2, once we have the interpolation set resulting from Algorithms 4.1 and 4.2, we can uniquely obtain a quadratic model whose Hessian is of minimal norm. Furthermore, by construction, we can obtain the model parameters α and β in a computationally stable way by solving the system in (3.5) and (3.6).

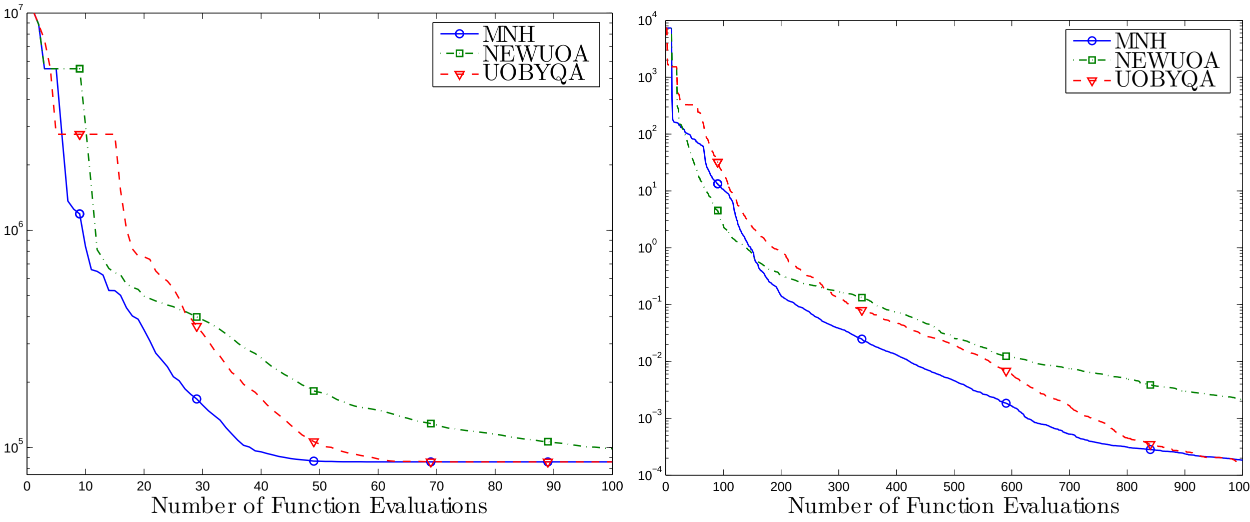

Figure 5.1: Mean of the best function value in 30 Trials ($\log_{10}$-scale, lowest is best): (a) Brown and Dennis function ($n = 4$); (b) Watson function ($n = 9$).

5 Preliminary Numerical Experiments

We have recently completed an initial implementation of the MNH algorithm. In this section we present the results of preliminary numerical tests.

We are particularly interested in how MNH performs compared to the NEWUOA [11] and UOBYQA [9] codes of Powell. NEWUOA was shown to have the best short-term performance on both smooth and mildly noisy functions in a test of three frequently-used derivative-free optimization algorithms [7]. UOBYQA requires more initial function evaluations but forms more accurate models.

Both are trust-region methods that use quadratic interpolation models. NEWUOA works with updates of the Hessian which are of minimal norm and a fixed number of interpolation points $p \in \{n+2, \dots, \frac{(n+1)(n+2)}{2}\}$, the value $p = 2n+1$ being recommended by Powell. Hence each time a newly evaluated point is added to the interpolation set, another point must be removed and will never return to the interpolation set. UOBYQA uses full quadratic models and thus always interpolates at $\frac{(n+1)(n+2)}{2}$ points.

We considered two smooth test functions from the set detailed in [7]. For each, we generated 30 random starting points within the unit hypercube and gave all codes the same starting point and trust-region radius. In Figure 5.1 we show the mean trajectory of the best f value obtained as a function of the number of evaluations of f. The interpretation here is that each solver would output the value shown as its approximate solution given this number of function evaluations.

In Figure 5.1 (a) we show the results for the ($n = 4$)-dimensional Brown and Dennis function. Note that MNH, NEWUOA, and UOBYQA require initializations of $n + 1 = 5$, $2n + 1 = 9$, and $\frac{(n+1)(n+2)}{2} = 15$ function values, respectively. We see that MNH obtains an initial lead because of its shorter initialization and then continues to make marked progress, yielding the best approximate solution for virtually all numbers of evaluations.

In Figure 5.1 (b) we show the results for the ($n = 9$)-dimensional Watson function. We see that MNH again has a slight initial advantage over NEWUOA and UOBYQA because it begins solving trust-region subproblems after $n+1$ evaluations. Further, given between 155 and 1000 evaluations, MNH obtains the best solution on average. For these numbers of function evaluations MNH often has the ability to use a full quadratic number $\frac{(n+1)(n+2)}{2} = 55$ of points from the bank while NEWUOA is

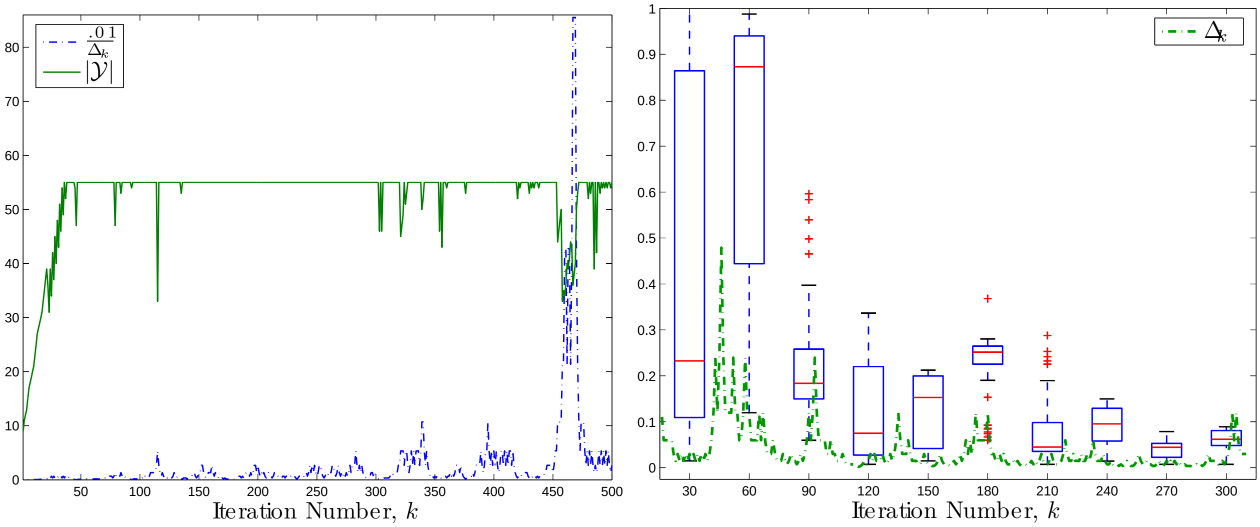

Figure 5.2: One run on the Watson function: (a) Inverse of the trust-region radius and number of interpolation points; (b) Distribution of the distances to the interpolation points and the trust-region radius.

always using only $2n+1 = 19$ points. This allows MNH to form models based on more information. That NEWUOA outperforms MNH between 30 and 155 evaluations is interesting, and we hope that as our implementation matures we may better understand this difference.

For one run on the Watson problem, Figure 5.2 (a) shows the number of points at which the MNH model interpolates the function and the inverse of the trust-region radius Δ_k, scaled for visibility. We note that MNH is able to make efficient use of the bank of points, $|\mathcal{Y}|$ growing from $n+1 = 10$ to the upper bound of 55, using a full quadratic model for the majority of the iterations. The iterations when this upper bound is not achieved usually correspond to those where the trust-region radius Δ_k has experienced considerable decrease.

For the same run, Figure 5.2 (b) shows the distribution of the distances from the interpolation points to the current iterate x_k. Here we see that the interpolation set consists of points which are close to x_k. As expected, the distribution tends toward larger distances after periods of larger trust-regions and the models are constructed in smaller neighborhoods as the algorithm progresses.

6 Conclusions and Future Work

In this paper we have outlined a new algorithm for derivative-free optimization. The quadratic models employed resemble those used by Powell in [10] but our method of constructing the interpolation set allows for a convergence result that is unlikely to be established for NEWUOA. Our method is also able to take advantage of more data in the bank of previously evaluated points, often employing a full quadratic number of them in our tests. Our preliminary results are encouraging and we expect these to improve as our code matures.

The approach outlined can also be extended to other types of interpolation models, from higher order polynomials to different forms of underdetermined quadratics. Regarding the latter we note that it may be advantageous to obtain a better estimate of the gradient than via the system in (3.6). For example, one could obtain the coefficients α using only $n+1$ nearby points and then form the minimal norm Hessian given this fixed α. This is just one of many areas of future work inspired by the approach introduced here.

Acknowledgments

This work was initiated while the author was visiting the Mathematics & Computer Science Division at Argonne National Laboratory. The author is grateful to Jorge Moré for useful conversations throughout the course of this work. The author is also grateful to Christine Shoemaker for providing many of the computationally expensive problems motivating this work.

References

[1] A.R. Conn, N.I.M. Gould, and P.L. Toint, *Trust-region methods*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, PA, USA, 2000.

[2] A.R. Conn, K. Scheinberg, and P.L. Toint, *Recent progress in unconstrained nonlinear optimization without derivatives*, Math. Programming, 79 (1997), pp. 397–414.

[3] A.R. Conn, K. Scheinberg, and L.N. Vicente, *Global convergence of general derivative-free trust-region algorithms to first and second order critical points*, Tech. Report Preprint 06-49, Departamento de Matemática, Universidade de Coimbra, Portugal, 2006.

[4] ——, *Geometry of interpolation sets in derivative free optimization*, Math. Programming, 111 (2008), pp. 141–172.

[5] R. Fletcher, *Practical Methods of Optimization*, J. Wiley & Sons, New York, 2nd ed., 1987.

[6] P.D. Hough, T.G. Kolda, and V.J. Torczon, *Asynchronous parallel pattern search for nonlinear optimization*, SIAM J. on Scientific Computing, 23 (2001), pp. 134–156.

[7] J.J. Moré and S.M. Wild, *Benchmarking derivative-free optimization algorithms*, Tech. Report ANL/MCS-P1471-1207, Argonne National Lab., MCS Division, 2007. Submitted to SIAM Review, January 2008.

[8] R. Oeuvray, *Trust-Region Methods Based on Radial Basis Functions with Application to Biomedical Imaging*, PhD thesis, EPFL, Lausanne, Switzerland, 2005.

[9] M.J.D. Powell, *UOBYQA: unconstrained optimization by quadratic approximation*, Math. Programming, 92 (2002), pp. 555–582.

[10] ——, *Least Frobenius norm updating of quadratic models that satisfy interpolation conditions*, Math. Programming, 100 (2004), pp. 183–215.

[11] ——, *The NEWUOA software for unconstrained optimization without derivatives*, in Large-Scale Nonlinear Optimization, Springer, 2006, pp. 255–297.

[12] H. Wendland, *Scattered Data Approximation*, Cambridge University Press, England, 2005.

[13] S.M. Wild, R.G. Regis, and C.A. Shoemaker, *ORBIT: optimization by radial basis function interpolation in trust-regions*, Tech. Report ORIE-1459, Cornell University, May 2007. Submitted to SIAM J. on Scientific Computing, May 2007.

[14] S.M. Wild and C.A. Shoemaker, *Global convergence of radial basis function trust-region algorithms for computationally expensive derivative-free optimization*, In preparation.