

Model Construction for Convex-Constrained Derivative-Free Optimization

Lindon Roberts*

March 22, 2024

Abstract

We develop a new approximation theory for linear and quadratic interpolation models, suitable for use in convex-constrained derivative-free optimization (DFO). Most existing model-based DFO methods for constrained problems assume the ability to construct sufficiently accurate approximations via interpolation, but the standard notions of accuracy (designed for unconstrained problems) may not be achievable by only sampling feasible points, and so may not give practical algorithms. This work extends the theory of convex-constrained linear interpolation developed in [Hough & Roberts, *SIAM J. Optim.*, 32:4 (2022), pp. 2552–2579] to the case of linear regression models and underdetermined quadratic interpolation models.

Keywords: derivative-free optimization, convex constraints, interpolation

Mathematics Subject Classification: 65K05, 90C30, 90C56

1 Introduction

This work is concerned with model-based methods for derivative-free optimization (DFO). Such methods are widely studied and have proven to be popular in practice [16, 1, 25]. Most commonly, model-based methods construct a local interpolant for the objective from selected evaluations which is then used in place of Taylor series in a standard framework, most commonly trust-region methods [13]. The convergence and worst-case complexity of model-based DFO in the unconstrained setting is well-established. Here, we are interested in nonconvex minimization over a convex set,

$$\min_{\mathbf{x} \in C} f(\mathbf{x}), \tag{1.1}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is smooth (but nonconvex), and $C \subseteq \mathbb{R}^n$ is a closed, convex set with nonempty interior. Such problems appear in the derivative-free setting, for example bound and linear constraints appearing in problems from climate science [31, 30], deep learning [33] and systems design engineering [23]. Although a model-based DFO framework for solving (1.1) was recently developed in [22], it only considered the case of constructing linear interpolants from function evaluations (at feasible points). Outside of structured problems such as nonlinear least-squares [8], quadratic interpolants are the most common models used in model-based DFO [28, 29, 16]. In this work, we develop a comprehensive approximation theory for linear regression and underdetermined quadratic interpolation models based on sampling only from feasible points. This theory fits the requirements of the method from [22], and so allows this framework to be used with more practical model choices.

*School of Mathematics and Statistics, University of Sydney, Camperdown NSW 2006, Australia (lindon.roberts@sydney.edu.au). This work was supported by the Australian Research Council Discovery Early Career Award DE240100006.

1.1 Existing work

Model-based DFO methods for unconstrained problems have well-established convergence [13, 16] and worst-case complexity theory [17, 25]. This theory relies on building (typically linear or quadratic) interpolants for the objective function based on samples of the objective at specific points, which must be “fully linear”, i.e. satisfying specific error bounds. Unlike much approximation theory in numerical analysis (e.g. [32]), it is typically assumed that we have limited ability to select the sample points and instead should use as many pre-existing evaluations as possible (motivated by problems with computationally expensive objectives, an important DFO use case). In this setting, the model accuracy can be bounded by quantities only depending on the choice of sampling points (and information about the smoothness of the objective). Originally in [13], the concept of “ Λ -poised” interpolation sets was introduced as a sufficient condition for constructing fully linear interpolation models, as well as developing procedures for improving the choice of sample points (while retaining as many existing points as possible) to construct a Λ -poised set. This was extended in [14] for polynomial regression and underdetermined polynomial interpolation models. A specific form of underdetermined quadratic interpolation, namely minimum Frobenius interpolation, was introduced and extensively studied in [26, 27] and is the foundation of several state-of-the-art codes [28, 29, 6]. An analysis of minimum Frobenius interpolation in terms of Λ -poisedness/fully linear error bounds was given in [16]. Alternate model constructions have been proposed based on weighted linear regression [5], underdetermined quadratic interpolation based on Sobolev norms [37], radial basis functions (RBF) [35] and combinations of polynomial and RBFs [2], for example.

Many model-based DFO methods have been proposed for constrained optimization; see [25, Section 7] for a thorough survey. We note in particular the following strictly feasible methods: [10] studies convex-constrained problems, [29, 19] and [34, Section 6.3] consider bound constraints, and [20] considers linear inequality constraints. Strictly feasible methods for nonconvex-constrained problems have also been considered in [11, 9, 36]. Of these, where convergence was established it was based on the assumption that fully linear models could be constructed using the same Λ -poised interpolation sets as in the unconstrained case. However, this is potentially impractical, as in some settings “*it may be impossible to obtain a fully linear model using only feasible points*” [25, p. 362].

More recently, [22] introduces a model-based DFO method for solving (1.1) by adapting the unconstrained approach from [15] based on gradient-based trust-region methods for convex-constrained optimization [12, Chapter 12]. The method has convergence and worst-case complexity results which align with unconstrained model-based DFO [17] and convex-constrained gradient-based trust-region methods [7]. To make this approach practical, it introduced a weaker notion of fully linear models which are sufficient for algorithm convergence, but which can be attained using linear interpolation over a Λ -poised set of only feasible points. It also demonstrated how to modify an interpolation set to make it Λ -poised (without changing too many points). However, linear models are not typically used in practice, except for special cases (such as nonlinear least-squares minimization [8]), with quadratic models being preferred [28, 29, 6].

We additionally note that the BOBYQA code [29] for bound-constrained optimization uses maximization of Lagrange polynomials inside the feasible region—similar to how we will define Λ -poisedness in this setting—but this was a heuristic with no theoretical analysis.¹

1.2 Contributions

The contributions of this work are:

- A generalization of the linear interpolation theory (based on sampling feasible points) from [22] to the case of linear regression models. This includes fully linear error bounds and procedures to generate Λ -poised interpolation sets (which are simpler than those developed in [22]).

¹In fact, our results here give some theoretical justification to this approach.

- A full approximation theory for underdetermined quadratic interpolation (in the style of [26, 27]) based on sampling only feasible points. This includes derivation of fully linear error bounds, defining the relevant notion of Λ -poisedness in this setting, and procedures for efficiently generating Λ -poised sets. By construction, in the case of a full interpolation set of $(n+1)(n+2)/2$ points for quadratic interpolation, this approach produces the same result as standard quadratic interpolation, and so our results also apply to this setting.

While the linear regression theory is of independent interest for DFO applied to noisy objectives [24], it is primarily included as a prerequisite for deriving the quadratic results. The results proven in both cases are sufficient to enable these model constructions to be used in the algorithmic framework from [22] (with the associated convergence and worst-case complexity guarantees).

While the formulations and results are very similar to those for the unconstrained case from [14], the constrained setting necessitates new approaches for proving the results (which when applied to the $C = \mathbb{R}^n$ case provide simpler proofs of existing results).

Organization A summary of the modified fully linear model accuracy requirement and associated algorithm from [22] is given in Section 2. That Λ -poisedness (only over the feasible region) guarantees fully linear interpolation models is first shown for linear regression models in Section 3 and then for (underdetermined) quadratic interpolation models in Section 4. Lastly, Section 5 provides concrete procedures for identifying and constructing Λ -poised quadratic interpolation models using only feasible points.

Notation We let $\|\cdot\|$ be the Euclidean norm of vectors and the operator 2-norm of matrices, and use $B(\mathbf{x}, \Delta)$ for $\mathbf{x} \in \mathbb{R}^n$ and $\Delta > 0$ for the closed ball $\{\mathbf{y} \in \mathbb{R}^n : \|\mathbf{y} - \mathbf{x}\| \leq \Delta\}$.

2 Background

This section summarizes the key background and results from [22], which provide a model-based DFO algorithm for solving (1.1).

We now outline how problem (1.1) can be solved using model-based DFO methods. We begin with outlining our problem assumptions.

Assumption 2.1. The objective function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is bounded below by f_{low} and continuously differentiable, and ∇f is $L_{\nabla f}$ -Lipschitz continuous.

Although ∇f exists, we assume that it is not available to the algorithm as an oracle.

Assumption 2.2. The feasible set $C \subseteq \mathbb{R}^n$ is closed and convex with nonempty interior.

Assumption 2.2 implies that the Euclidean projection operator for C ,

$$\text{proj}_C(\mathbf{y}) := \arg \min_{\mathbf{x} \in C} \|\mathbf{y} - \mathbf{x}\|^2, \quad (2.1)$$

is a well-defined function $\text{proj}_C : \mathbb{R}^n \rightarrow C$ [3, Theorem 6.25]. Our framework is based on a setting in which proj_C is inexpensive to evaluate (at least compared to evaluations of the objective f). Several examples of feasible sets C which have simple analytic expressions for proj_C are given in [3, Table 6.1].

In this work, as in [22], we aim to find a first-order optimal solution to (1.1). To measure this, we use the following first-order criticality measure from [12, Section 12.1.4], defined for all $\mathbf{x} \in C$:

$$\pi^f(\mathbf{x}) := \left| \min_{\substack{\mathbf{x} + \mathbf{d} \in C \\ \|\mathbf{d}\| \leq 1}} \nabla f(\mathbf{x})^T \mathbf{d} \right|. \quad (2.2)$$

From [12, Theorem 12.1.6], we have that π^f is a non-negative, continuous function of \mathbf{x} and $\pi^f(\mathbf{x}^*) = 0$ if and only if \mathbf{x}^* is a first-order critical point for (1.1). In the unconstrained case $C = \mathbb{R}^n$, then (2.2) simplifies to $\pi^f(\mathbf{x}) = \|\nabla f(\mathbf{x})\|$. For notational convenience, we define $\pi_k^f := \pi^f(\mathbf{x}_k)$, where $\mathbf{x}_k \in \mathbb{R}^n$ is the k -th iterate of our algorithm.

2.1 Model Construction & Accuracy

We will consider a model-based DFO approach for solving (1.1). Specifically, at each iteration k we construct a local quadratic approximation to f which we hope to be accurate near the current iterate $\mathbf{x}_k \in \mathbb{R}^n$:

$$f(\mathbf{y}) \approx m_k(\mathbf{y}) := c_k + \mathbf{g}_k^T(\mathbf{y} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{y} - \mathbf{x}_k)^T H_k(\mathbf{y} - \mathbf{x}_k), \quad (2.3)$$

for some $c_k \in \mathbb{R}$, $\mathbf{g}_k \in \mathbb{R}^n$ and $H_k \in \mathbb{R}^{n \times n}$ with $H_k = H_k^T$.

Assumption 2.3. There exists $\kappa_H \geq 1$ such that $\|H_k\| \leq \kappa_H - 1$ for all k .

Convergence of our algorithm will require that the local models m_k (2.3) are sufficiently accurate approximations of the objective f (at least on some iterations). The required notion of ‘sufficiently accurate’ is the following:

Definition 2.4. The model m_k (2.3) is C -fully linear in $B(\mathbf{x}_k, \Delta_k)$ if there exist $\kappa_{\text{ef}}, \kappa_{\text{eg}} > 0$, independent of k , such that

$$\max_{\substack{\mathbf{x}_k + \mathbf{d} \in C \\ \|\mathbf{d}\| \leq \Delta_k}} |f(\mathbf{x}_k + \mathbf{d}) - m_k(\mathbf{x}_k + \mathbf{d})| \leq \kappa_{\text{ef}} \Delta_k^2, \quad (2.4a)$$

$$\max_{\substack{\mathbf{x}_k + \mathbf{d} \in C \\ \|\mathbf{d}\| \leq 1}} |(\nabla f(\mathbf{x}_k) - \mathbf{g}_k)^T \mathbf{d}| \leq \kappa_{\text{eg}} \Delta_k. \quad (2.4b)$$

When we wish to refer to this notion in the abstract sense, we use the term “ C -full linearity”.

We note in particular that (2.4a) uses the constraint $\|\mathbf{d}\| \leq \Delta_k$ and (2.4b) uses $\|\mathbf{d}\| \leq 1$. This essentially corresponds to the standard definition of fully linear models [15, Definition 3.1] in the unconstrained case $C = \mathbb{R}^n$.

Our algorithm will require the existence of two procedures related to C -full linearity: given a model m_k , iterate \mathbf{x}_k and trust-region radius Δ_k , we assume that we can

- Verify whether or not m_k is C -fully linear in $B(\mathbf{x}_k, \Delta_k)$; and
- If m_k is not C -fully linear, create a new model (typically a small modification of m_k) which is C -fully linear in $B(\mathbf{x}_k, \Delta_k)$.

Lastly, our model m_k (2.3) induces an approximate first-order criticality measure, namely

$$\pi^m(\mathbf{x}) := \left| \min_{\substack{\mathbf{x} + \mathbf{d} \in C \\ \|\mathbf{d}\| \leq 1}} \mathbf{g}_k^T \mathbf{d} \right|. \quad (2.5)$$

As above, for convenience we will define $\pi_k^m := \pi^m(\mathbf{x}_k)$.

2.2 Algorithm Specification

Our DFO method is based on a trust-region framework. That is, given the local model m_k (2.3), we calculate a potential new iterate as $\mathbf{x}_k + \mathbf{s}_k$, where \mathbf{s}_k is an approximate minimizer of the trust-region subproblem

$$\min_{\substack{\mathbf{x} + \mathbf{s} \in C \\ \|\mathbf{s}\| \leq \Delta_k}} m_k(\mathbf{x}_k + \mathbf{s}), \quad (2.6)$$

where $\Delta_k > 0$ is a parameter updated dynamically inside the algorithm. Formally, we require the following, which can be achieved using a Goldstein-type linesearch method [12, Algorithm 12.2.2 & Theorem 12.2.2].

Algorithm 1 CDFO-TR: model-based DFO method for (1.1), from [22].

Input: Starting point $\mathbf{x}_0 \in \mathbb{R}^n$ and trust-region radius $\Delta_0 > 0$.

Parameters: maximum trust-region radius $\Delta_{\max} \geq \Delta_0$, scaling factors $0 < \gamma_{\text{dec}} < 1 < \gamma_{\text{inc}}$, criticality constants $\epsilon_C, \mu > 0$, and acceptance threshold $\eta \in (0, 1)$.

1: Build an initial model m_0 (2.3).

2: **for** $k = 0, 1, 2, \dots$ **do**

3: **if** $\pi_k^m < \epsilon_C$ **and** ($\pi_k^m < \mu^{-1}\Delta_k$ **or** m_k is not C -fully linear in $B(\mathbf{x}_k, \Delta_k)$) **then**

4: Criticality step: Set $\mathbf{x}_{k+1} = \mathbf{x}_k$. If m_k is C -fully linear in $B(\mathbf{x}_k, \Delta_k)$, set $\Delta_{k+1} = \gamma_{\text{dec}}\Delta_k$, otherwise set $\Delta_{k+1} = \Delta_k$. Construct m_{k+1} to be C -fully linear in $B(\mathbf{x}_{k+1}, \Delta_{k+1})$.

5: **else** $\leftarrow \pi_k^m \geq \epsilon_C$ **or** ($\pi_k^m \geq \mu^{-1}\Delta_k$ **and** m_k is C -fully linear in $B(\mathbf{x}_k, \Delta_k)$)

6: Approximately solve (2.6) to get a step \mathbf{s}_k .

7: Evaluate $f(\mathbf{x}_k + \mathbf{s}_k)$ and calculate ratio

$$\rho_k := \frac{f(\mathbf{x}_k) - f(\mathbf{x}_k + \mathbf{s}_k)}{m_k(\mathbf{x}_k) - m_k(\mathbf{x}_k + \mathbf{s}_k)}. \quad (2.8)$$

8: **if** $\rho_k \geq \eta$ **then**

9: Successful step: Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$ and $\Delta_{k+1} = \min(\gamma_{\text{inc}}\Delta_k, \Delta_{\max})$. Form m_{k+1} in any manner.

10: **else if** m_k is not C -fully linear in $B(\mathbf{x}_k, \Delta_k)$ **then**

11: Model-improving step: Set $\mathbf{x}_{k+1} = \mathbf{x}_k$ and $\Delta_{k+1} = \Delta_k$, and construct m_{k+1} to be C -fully linear in $B(\mathbf{x}_{k+1}, \Delta_{k+1})$.

12: **else** $\leftarrow \rho_k < \eta$ **and** m_k is C -fully linear in $B(\mathbf{x}_k, \Delta_k)$

13: Unsuccessful step: Set $\mathbf{x}_{k+1} = \mathbf{x}_k$ and $\Delta_{k+1} = \gamma_{\text{dec}}\Delta_k$. Form m_{k+1} in any manner.

14: **end if**

15: **end if**

16: **end for**

Assumption 2.5. There exists a constant $c_1 \in (0, 1)$ such that the computed step \mathbf{s}_k satisfies $\mathbf{x}_k + \mathbf{s}_k \in C$, $\|\mathbf{s}_k\| \leq \Delta_k$ and the generalized Cauchy decrease condition:

$$m_k(\mathbf{x}_k) - m_k(\mathbf{x}_k + \mathbf{s}_k) \geq c_1 \pi_k^m \min\left(\frac{\pi_k^m}{1 + \|H_k\|}, \Delta_k, 1\right). \quad (2.7)$$

The full derivative-free algorithm for solving (1.1) from [22], called CDFO-TR, is given in Algorithm 1. The overall structure is similar to other model-based DFO methods, such as [15, Algorithm 4.1] for unconstrained minimization.

We have the following global convergence and worst-case complexity results for Algorithm 1 from [22].

Theorem 2.6 (Theorem 3.10 & Corollary 3.15, [22]). *If Assumptions 2.1, 2.3 and 2.5 hold, then $\lim_{k \rightarrow \infty} \pi_k^f = 0$. Moreover, if $\epsilon \in (0, 1]$ and $\epsilon_C \geq c_2 \epsilon$ for some constant $c_2 > 0$, then the number of iterations k before Algorithm 1 produces an iterate with $\pi_k^f < \epsilon$ is at most $\mathcal{O}(\kappa_H \kappa_d^2 \epsilon^{-2})$, where $\kappa_d := \max(\kappa_{\text{ef}}, \kappa_{\text{eg}})$.*

In [22], details are given on how to construct fully linear models based on linear interpolation to feasible points. Although linear models can be practical for some structured problems, such as nonlinear least-squares objectives (e.g. [8]), quadratic models are generally preferred. The remainder of this paper is devoted to the construction of fully linear quadratic models by only sampling the objective at feasible points.

3 Linear Regression Models

We first consider how to extend the linear interpolation approximation theory from [22] to the case of linear regression models (with the slight extra generalization that the base point \mathbf{x} does

not need to be an interpolation point). The purpose of this is twofold: regression models can be useful for modelling noisy objectives (e.g. [5]), and this theory will be necessary to develop the corresponding quadratic interpolation theory in Section 4. In the unconstrained case $C = \mathbb{R}^n$, these results were originally proven in [14].

Suppose we have sampled the function f at p points $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset \mathbb{R}^n$ (where $p \geq n + 1$), and given this information we wish to find a linear model

$$f(\mathbf{y}) \approx m(\mathbf{y}) := c + \mathbf{g}^T(\mathbf{y} - \mathbf{x}), \quad (3.1)$$

by solving

$$\min_{c, \mathbf{g} \in \mathbb{R} \times \mathbb{R}^n} \sum_{t=1}^p (f(\mathbf{y}_t) - m(\mathbf{y}_t))^2. \quad (3.2)$$

Equivalently, we can find the c and \mathbf{g} for our model by finding the least-squares solution to the $p \times (n + 1)$ system

$$M \begin{bmatrix} c \\ \mathbf{g} \end{bmatrix} := \begin{bmatrix} 1 & (\mathbf{y}_1 - \mathbf{x})^T \\ \vdots & \vdots \\ 1 & (\mathbf{y}_p - \mathbf{x})^T \end{bmatrix} \begin{bmatrix} c \\ \mathbf{g} \end{bmatrix} = \begin{bmatrix} f(\mathbf{y}_1) \\ \vdots \\ f(\mathbf{y}_p) \end{bmatrix}, \quad (3.3)$$

which may be written as

$$\begin{bmatrix} c \\ \mathbf{g} \end{bmatrix} = M^\dagger \begin{bmatrix} f(\mathbf{y}_1) \\ \vdots \\ f(\mathbf{y}_p) \end{bmatrix}, \quad (3.4)$$

where $M^\dagger \in \mathbb{R}^{(n+1) \times p}$ is the Moore-Penrose pseudoinverse of M .

Later we will require the following standard properties of M^\dagger (see, e.g. [18, Section 5.5.4]).

Lemma 3.1. *For $p \geq n + 1$, the Moore-Penrose pseudoinverse M^\dagger of M (3.3) satisfies $(M^T)^\dagger = (M^\dagger)^T$ and $M^\dagger M M^\dagger = M^\dagger$. The minimal-norm solution to the underdetermined system $M^T \mathbf{u} = \mathbf{v}$ is $\mathbf{u} = (M^T)^\dagger \mathbf{v}$.*

The quality of the choice of interpolation points will be assessed by considering the associated set of Lagrange polynomials. In this case, the Lagrange polynomials associated with our interpolation set are the linear functions

$$\ell_t(\mathbf{y}) := c_t + \mathbf{g}_t^T(\mathbf{y} - \mathbf{x}), \quad \forall t = 1, \dots, p, \quad (3.5)$$

each defined by the least-squares regression problem

$$\min_{c_t, \mathbf{g}_t \in \mathbb{R} \times \mathbb{R}^n} \sum_{s=1}^{p+1} (\delta_{s,t} - \ell_t(\mathbf{y}_s))^2, \quad \forall t = 1, \dots, p, \quad (3.6)$$

or equivalently

$$\begin{bmatrix} c_t \\ \mathbf{g}_t \end{bmatrix} = M^\dagger \mathbf{e}_t, \quad \forall t = 1, \dots, p. \quad (3.7)$$

Our notion of the sampled points $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ being a ‘good’ choice is given by the Lagrange polynomials having small magnitude in the region of interest. This is formalized in the following notion, which generalizes [14, Definition 2.7] to the convex feasible region C .

Definition 3.2. Given $\Lambda \geq 1$, the set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset C$ is Λ -poised for linear regression in $B(\mathbf{x}, \Delta) \cap C$ if $\{\mathbf{y}_2 - \mathbf{y}_1, \dots, \mathbf{y}_p - \mathbf{y}_1\}$ spans \mathbb{R}^n and

$$\max_{t=1, \dots, p} |\ell_t(\mathbf{y})| \leq \Lambda, \quad \forall \mathbf{y} \in C \cap B(\mathbf{x}, \min(\Delta, 1)). \quad (3.8)$$

We note that if $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ is Λ -poised for linear regression, then M has full column rank (since $\{\mathbf{y}_2 - \mathbf{y}_1, \dots, \mathbf{y}_p - \mathbf{y}_1\}$ is assumed to span \mathbb{R}^n). This requirement also necessitates that $p \geq n + 1$.

Assuming the Λ -poisedness of $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ will be sufficient for the associated linear regression model to be C -fully linear. The proof of this follows a similar structure to [22, Lemma 4.3 & Theorem 4.4], but with an increased complexity to the proofs coming from using regression instead of interpolation.

Lemma 3.3. *Suppose f satisfies Assumption 2.1 and C satisfies Assumption 2.2. Then if $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ is Λ -poised for linear regression in $B(\mathbf{x}, \Delta) \cap C$ and $\|\mathbf{y}_t - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ for all $t = 1, \dots, p$ and some $\beta > 0$, we have*

$$|m(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| \leq \frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2, \quad (3.9)$$

for all $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$. If we also have $\mathbf{x} \in C$, this in turn implies error bounds on c and \mathbf{g} individually, namely

$$|c - f(\mathbf{x})| \leq \frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2, \quad (3.10)$$

and

$$|(\mathbf{y} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| \leq p\Lambda L_{\nabla f} \beta^2 \min(\Delta, 1)^2, \quad (3.11)$$

for all $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$.

Proof. We begin by defining the residual of the least-squares problem (3.3) as

$$\mathbf{r} := M \begin{bmatrix} c \\ \mathbf{g} \end{bmatrix} - \begin{bmatrix} f(\mathbf{y}_1) \\ \vdots \\ f(\mathbf{y}_p) \end{bmatrix} = (MM^\dagger - I) \begin{bmatrix} f(\mathbf{y}_1) \\ \vdots \\ f(\mathbf{y}_p) \end{bmatrix} \in \mathbb{R}^p. \quad (3.12)$$

Then for all $t = 1, \dots, p$ have

$$m(\mathbf{y}_t) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y}_t - \mathbf{x}) = \begin{bmatrix} c \\ \mathbf{g} \end{bmatrix}^T \underbrace{\begin{bmatrix} 1 \\ \mathbf{y}_t - \mathbf{x} \end{bmatrix}}_{=M^T \mathbf{e}_t} - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y}_t - \mathbf{x}), \quad (3.13)$$

$$= \mathbf{r}^T \mathbf{e}_t + f(\mathbf{y}_t) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y}_t - \mathbf{x}). \quad (3.14)$$

Now, fix $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$. Since M has full column rank, its rows span \mathbb{R}^{n+1} and so there exist constants $\alpha_t(\mathbf{y})$ such that

$$\begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix} = \sum_{t=1}^p \alpha_t(\mathbf{y}) \begin{bmatrix} 1 \\ \mathbf{y}_t - \mathbf{x} \end{bmatrix} = M^T \boldsymbol{\alpha}(\mathbf{y}). \quad (3.15)$$

Of these, we take the $\boldsymbol{\alpha}(\mathbf{y})$ with minimal norm (c.f. Lemma 3.1),

$$\boldsymbol{\alpha}(\mathbf{y}) = (M^T)^\dagger \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix}. \quad (3.16)$$

There are two key properties of $\boldsymbol{\alpha}(\mathbf{y})$: firstly,

$$\ell_t(\mathbf{y}) = \begin{bmatrix} c_t \\ \mathbf{g}_t \end{bmatrix}^T \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix} = \mathbf{e}_t^T (M^\dagger)^T \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix} = \alpha_t(\mathbf{y}), \quad (3.17)$$

where the last equality follows $(M^T)^\dagger = (M^\dagger)^T$ (Lemma 3.1). Hence $|\alpha_t(\mathbf{y})| = |\ell_t(\mathbf{y})| \leq \Lambda$ from the Λ -poisedness condition. Secondly, we have

$$\boldsymbol{\alpha}(\mathbf{y})^T \mathbf{r} = \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix}^T M^\dagger (MM^\dagger - I) \begin{bmatrix} f(\mathbf{y}_1) \\ \vdots \\ f(\mathbf{y}_p) \end{bmatrix} = 0, \quad (3.18)$$

from $M^\dagger MM^\dagger = M^\dagger$ (Lemma 3.1). All together, we have

$$|m(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| = \left| \begin{bmatrix} c \\ \mathbf{g} \end{bmatrix}^T \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix} - \begin{bmatrix} f(\mathbf{x}) \\ \nabla f(\mathbf{x}) \end{bmatrix}^T \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix} \right|, \quad (3.19)$$

$$= \left| \sum_{t=1}^p \alpha_t(\mathbf{y}) \left(\begin{bmatrix} c \\ \mathbf{g} \end{bmatrix}^T \begin{bmatrix} 1 \\ \mathbf{y}_t - \mathbf{x} \end{bmatrix} - \begin{bmatrix} f(\mathbf{x}) \\ \nabla f(\mathbf{x}) \end{bmatrix}^T \begin{bmatrix} 1 \\ \mathbf{y}_t - \mathbf{x} \end{bmatrix} \right) \right|, \quad (3.20)$$

$$= \left| \sum_{t=1}^p \alpha_t(\mathbf{y}) \{ \mathbf{r}^T \mathbf{e}_t + f(\mathbf{y}_t) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y}_t - \mathbf{x}) \} \right|, \quad (3.21)$$

$$\leq |\boldsymbol{\alpha}(\mathbf{y})^T \mathbf{r}| + \frac{L_{\nabla f}}{2} \sum_{t=1}^p |\alpha_t(\mathbf{y})| \cdot \|\mathbf{y}_t - \mathbf{x}\|^2, \quad (3.22)$$

$$\leq \frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2, \quad (3.23)$$

and we recover (3.9), where we used (3.18), $|\alpha_t(\mathbf{y})| \leq \Lambda$ and $\|\mathbf{y}_t - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ to get the last inequality.

If $\mathbf{x} \in C$, we can take $\mathbf{y} = \mathbf{x}$ in (3.9) to get (3.10). Combining (3.10) with (3.9) we get

$$|(\mathbf{y} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| \leq |c + \mathbf{g}^T(\mathbf{y} - \mathbf{x}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| + |c - f(\mathbf{x})|, \quad (3.24)$$

$$\leq p\Lambda L_{\nabla f} \beta^2 \min(\Delta, 1)^2, \quad (3.25)$$

and we get (3.11). \square

Theorem 3.4. *Suppose the assumptions of Lemma 3.3 hold and $\mathbf{x} \in C$. Then the regression model (3.1) is C -fully linear in $B(\mathbf{x}, \Delta)$ with constants*

$$\kappa_{\text{ef}} = p\Lambda L_{\nabla f} \beta^2 + \frac{L_{\nabla f}}{2}, \quad \text{and} \quad \kappa_{\text{eg}} = p\Lambda L_{\nabla f} \beta^2, \quad (3.26)$$

in (2.4).

Proof. Fix $\mathbf{y} \in B(\mathbf{x}, \Delta) \cap C$. We first derive κ_{ef} by considering the cases $\Delta > 1$ and $\Delta \leq 1$ separately.

If $\Delta > 1$, then $\hat{\mathbf{y}} = \mathbf{x} + \frac{1}{\Delta}(\mathbf{y} - \mathbf{x}) \in B(\mathbf{x}, 1) = B(\mathbf{x}, \min(\Delta, 1))$. Since C is convex and $\mathbf{x}, \mathbf{y} \in C$ we also have $\hat{\mathbf{y}} \in C$. Hence (3.9) gives

$$|c + \mathbf{g}^T(\hat{\mathbf{y}} - \mathbf{x}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\hat{\mathbf{y}} - \mathbf{x})| \leq \frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2, \quad (3.27)$$

and so

$$\left| c + \frac{1}{\Delta} \mathbf{g}^T(\mathbf{y} - \mathbf{x}) - f(\mathbf{x}) - \frac{1}{\Delta} \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) \right| \leq \frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2. \quad (3.28)$$

This gives us

$$|f(\mathbf{y}) - m(\mathbf{y})| \leq |f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| + |m(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})|, \quad (3.29)$$

$$\leq \frac{L_{\nabla f}}{2} \|\mathbf{y} - \mathbf{x}\|^2 + \Delta \left| \frac{1}{\Delta} c + \frac{1}{\Delta} \mathbf{g}^T(\mathbf{y} - \mathbf{x}) - \frac{1}{\Delta} f(\mathbf{x}) - \frac{1}{\Delta} \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) \right|, \quad (3.30)$$

$$\leq \frac{L_{\nabla f}}{2} \|\mathbf{y} - \mathbf{x}\|^2 + \Delta \left[\left| c + \frac{1}{\Delta} \mathbf{g}^T(\mathbf{y} - \mathbf{x}) - f(\mathbf{x}) - \frac{1}{\Delta} \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) \right| + \left| \left(\frac{1}{\Delta} - 1 \right) (c - f(\mathbf{x})) \right| \right], \quad (3.31)$$

$$\leq \frac{L_{\nabla f}}{2} \Delta^2 + \Delta \left[\frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2 + \left(1 - \frac{1}{\Delta} \right) \frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2 \right], \quad (3.32)$$

$$\leq \frac{L_{\nabla f}}{2} \Delta^2 + p\Lambda L_{\nabla f} \beta^2 \Delta \min(\Delta, 1)^2, \quad (3.33)$$

$$= \frac{L_{\nabla f}}{2} \Delta^2 + p\Lambda L_{\nabla f} \beta^2 \Delta^2, \quad (3.34)$$

where we use (3.10) to get the fourth inequality, and the last line follows from $\Delta > 1$. Instead if $\Delta \leq 1$, then $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$ already, and so (3.9) immediately gives

$$|f(\mathbf{y}) - m(\mathbf{y})| = |f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| + |m(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})|, \quad (3.35)$$

$$\leq \frac{L_{\nabla f}}{2} \Delta^2 + \frac{p\Lambda L_{\nabla f} \beta^2}{2} \min(\Delta, 1)^2, \quad (3.36)$$

$$\leq \frac{L_{\nabla f}}{2} \Delta^2 + \frac{p\Lambda L_{\nabla f} \beta^2}{2} \Delta^2. \quad (3.37)$$

Either way, we get the desired value of κ_{ef} .

To get κ_{eg} we now fix an arbitrary $\tilde{\mathbf{y}} \in B(\mathbf{x}, 1) \cap C$ and again consider the cases $\Delta \geq 1$ and $\Delta < 1$ separately. First, if $\Delta \geq 1$, then $\tilde{\mathbf{y}} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$. From (3.11) we get

$$|(\tilde{\mathbf{y}} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| \leq p\Lambda L_{\nabla f} \beta^2 \min(\Delta, 1)^2 \leq p\Lambda L_{\nabla f} \beta^2 \Delta, \quad (3.38)$$

where the second inequality follows from $\Delta \geq 1$. Alternatively, if $\Delta < 1$ then the convexity of C implies that $\hat{\mathbf{y}} := \mathbf{x} + \Delta(\tilde{\mathbf{y}} - \mathbf{x}) \in B(\mathbf{x}, \Delta) \cap C = B(\mathbf{x}, \min(\Delta, 1)) \cap C$. Again from (3.11) we get

$$|(\tilde{\mathbf{y}} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| = \frac{1}{\Delta} |(\hat{\mathbf{y}} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))|, \quad (3.39)$$

$$\leq p\Lambda L_{\nabla f} \beta^2 \Delta^{-1} \min(\Delta, 1)^2, \quad (3.40)$$

$$= p\Lambda L_{\nabla f} \beta^2 \Delta, \quad (3.41)$$

where the last line follows from $\Delta < 1$. Again, either way we get the desired value of κ_{eg} . \square

4 Underdetermined Quadratic Interpolation Models

We now consider the case of forming C -fully linear quadratic interpolation models. Our approach follows that of [26] for the unconstrained case, although the fully linear error bounds for this approach were shown later in [16]. We note that this approach is different to the underdetermined quadratic interpolation used in [14].

Here, we aim to construct a quadratic model

$$f(\mathbf{y}) \approx m(\mathbf{y}) := c + \mathbf{g}^T(\mathbf{y} - \mathbf{x}) + \frac{1}{2}(\mathbf{y} - \mathbf{x})^T H(\mathbf{y} - \mathbf{x}), \quad (4.1)$$

where $c \in \mathbb{R}$, $\mathbf{g} \in \mathbb{R}^n$ and $H \in \mathbb{R}^{n \times n}$ with $H = H^T$. We assume that our interpolation set is $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ with $p \in \{n+2, \dots, (n+1)(n+2)/2\}$. The case $p = (n+1)(n+2)/2$ corresponds to full quadratic interpolation [13, Section 4.2]. We exclude the case $p = n+1$ which, from (4.2) below, corresponds to linear interpolation and was analyzed (in the convex-constrained case) in [22, Section 4].

If $n+2 \leq p < (n+1)(n+2)/2$ then there are infinitely many models satisfying the interpolation conditions $f(\mathbf{y}_t) = m(\mathbf{y}_t)$ for all $t = 1, \dots, p$. So, following [26] we choose the model with minimum Frobenius norm Hessian by solving

$$\min_{c, \mathbf{g}, H \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{n \times n}} \frac{1}{4} \|H\|_F^2, \quad (4.2a)$$

$$\text{s.t. } f(\mathbf{y}_t) = m(\mathbf{y}_t), \quad \forall t = 1, \dots, p. \quad (4.2b)$$

This is a convex quadratic program, and (as shown in [26]), reduces to solving the $(p+n+1) \times (p+n+1)$ linear system

$$F \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_p \\ c \\ \mathbf{g} \end{bmatrix} := \left[\begin{array}{c|c} Q & M \\ \hline M^T & 0 \end{array} \right] \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_p \\ c \\ \mathbf{g} \end{bmatrix} = \begin{bmatrix} f(\mathbf{y}_1) \\ \vdots \\ f(\mathbf{y}_p) \\ 0 \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^{p+n+1}, \quad (4.3)$$

where $M \in \mathbb{R}^{p \times (n+1)}$ is from the linear regression problem (3.3) and $Q \in \mathbb{R}^{p \times p}$ has entries $Q_{i,j} = \frac{1}{2} [(\mathbf{y}_i - \mathbf{x})^T (\mathbf{y}_j - \mathbf{x})]^2$ for $i, j = 1, \dots, p$. The solution to (4.3) immediately gives us c and \mathbf{g} ; the (symmetric) model Hessian is given by $H = \sum_{t=1}^p \lambda_t (\mathbf{y}_t - \mathbf{x})(\mathbf{y}_t - \mathbf{x})^T$. We note that Q is symmetric positive semidefinite [26, eq. 2.10], and $\lambda_1, \dots, \lambda_p$ are the Lagrange multipliers associated with constraints (4.2b).

The Lagrange polynomials associated with our interpolation set are

$$\ell_t(\mathbf{y}) := c_t + \mathbf{g}_t^T (\mathbf{y} - \mathbf{x}) + \frac{1}{2} (\mathbf{y} - \mathbf{x})^T H_t (\mathbf{y} - \mathbf{x}), \quad \forall t = 1, \dots, p, \quad (4.4)$$

where c_t , \mathbf{g}_t and H_t come from solving (4.2) with (4.2b) replaced by $m(\mathbf{y}_s) = \delta_{s,t}$ for $s = 1, \dots, p$. Equivalently, we have

$$F \begin{bmatrix} \lambda_t \\ c_t \\ \mathbf{g}_t \end{bmatrix} = \mathbf{e}_t, \quad \forall t = 1, \dots, p, \quad (4.5)$$

and $H_t = \sum_{s=1}^p [\lambda_t]_s (\mathbf{y}_s - \mathbf{x})(\mathbf{y}_s - \mathbf{x})^T$. This gives

$$\ell_t(\mathbf{y}) = c_t + \mathbf{g}_t^T (\mathbf{y} - \mathbf{x}) + \frac{1}{2} \sum_{s=1}^p [\lambda_t]_s [(\mathbf{y} - \mathbf{x})^T (\mathbf{y}_s - \mathbf{x})]^2, \quad (4.6)$$

$$= \begin{bmatrix} \lambda_t \\ c_t \\ \mathbf{g}_t \end{bmatrix}^T \underbrace{\begin{bmatrix} \left\{ \frac{1}{2} [(\mathbf{y} - \mathbf{x})^T (\mathbf{y}_s - \mathbf{x})]^2 \right\}_{s=1, \dots, p} \\ 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix}}_{=: \phi(\mathbf{y})}, \quad (4.7)$$

$$= \mathbf{e}_t^T F^{-1} \phi(\mathbf{y}). \quad (4.8)$$

Given (4.3) and (4.5), we conclude that

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_p \end{bmatrix} = \sum_{t=1}^p f(\mathbf{y}_t) \lambda_t, \quad c = \sum_{t=1}^p f(\mathbf{y}_t) c_t, \quad \text{and} \quad \mathbf{g} = \sum_{t=1}^p f(\mathbf{y}_t) \mathbf{g}_t, \quad (4.9)$$

which gives

$$m(\mathbf{y}) = \sum_{t=1}^p f(\mathbf{y}_t) \ell_t(\mathbf{y}). \quad (4.10)$$

We now have our new definition of Λ -poisedness:

Definition 4.1. Given $\Lambda \geq 1$, the interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset C$ is Λ -poised for minimum Frobenius norm interpolation in $B(\mathbf{x}, \Delta) \cap C$ if F is invertible, and

$$\max_{t=1, \dots, p} |\ell_t(\mathbf{y})| \leq \Lambda, \quad \forall \mathbf{y} \in C \cap B(\mathbf{x}, \min(\Delta, 1)). \quad (4.11)$$

Lemma 4.2. *If the set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset C$ is Λ -poised for minimum Frobenius norm interpolation, then it is $\sqrt{p}\Lambda$ -poised for linear regression.*

Proof. Since F is invertible, the sub-matrix M has full column rank by [4, Theorem 3.3], and so $\{\mathbf{y}_2 - \mathbf{y}_1, \dots, \mathbf{y}_p - \mathbf{y}_1\}$ spans \mathbb{R}^n . The remainder of this proof is based on the argument in [16, p. 83]. We note that since $H = 0$ is a global minimizer of the objective function (4.2), that if f is linear then we have exact interpolation, $m = f$. Applying this to the functions $f(\mathbf{y}) = 1$ and $f(\mathbf{y}) = (\mathbf{y} - \mathbf{x})^T \mathbf{e}_i$ for $i = 1, \dots, n$, we get from (4.10) that

$$1 = \sum_{t=1}^p \ell_t(\mathbf{y}), \quad (\mathbf{y} - \mathbf{x})^T \mathbf{e}_i = \sum_{t=1}^p (\mathbf{y}_t - \mathbf{x})^T \mathbf{e}_i \cdot \ell_t(\mathbf{y}), \quad \forall i = 1, \dots, n. \quad (4.12)$$

Denoting $\boldsymbol{\ell}(\mathbf{y}) \in \mathbb{R}^p$ as the vector of all $\ell_1(\mathbf{y}), \dots, \ell_p(\mathbf{y})$, these are equivalent to

$$M^T \boldsymbol{\ell}(\mathbf{y}) = \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix}. \quad (4.13)$$

Hence $\boldsymbol{\ell}(\mathbf{y})$ is another solution to the (underdetermined) system (3.15). Since the minimal norm solution to (3.15) was $\boldsymbol{\alpha}(\mathbf{y})$, we must have $\|\boldsymbol{\alpha}(\mathbf{y})\| \leq \|\boldsymbol{\ell}(\mathbf{y})\|$. However from (3.17) we know $\boldsymbol{\alpha}(\mathbf{y}) = \boldsymbol{\ell}^{\text{reg}}(\mathbf{y})$, where $\boldsymbol{\ell}^{\text{reg}}(\mathbf{y})$ are the Lagrange polynomials associated with linear regression for $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$. Thus $\|\boldsymbol{\ell}^{\text{reg}}(\mathbf{y})\| \leq \|\boldsymbol{\ell}(\mathbf{y})\|$. Now, fixing any $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$ we get

$$\|\boldsymbol{\ell}^{\text{reg}}(\mathbf{y})\|_\infty \leq \|\boldsymbol{\ell}^{\text{reg}}(\mathbf{y})\| \leq \|\boldsymbol{\ell}(\mathbf{y})\| \leq \sqrt{p} \|\boldsymbol{\ell}(\mathbf{y})\|_\infty \leq \sqrt{p} \Lambda, \quad (4.14)$$

and we are done. \square

We are now in a position to construct our fully linear error bounds. We will begin by proving a bound on the size of the model Hessian, which requires the following technical result.

Lemma 4.3. *Fix $t > 1$ and $c_1, c_2 \geq 0$. If $a, b \in \mathbb{R}$ satisfy $|a + b| \leq c_1$ and $|ta + b| \leq c_2$ then $|a| \leq (c_1 + c_2)/(t - 1)$ and $|b| \leq (tc_1 + c_2)/(t - 1)$.*

Proof. We first prove the bound on $|a|$. To find a contradiction, first suppose that $a > (c_1 + c_2)/(t - 1)$ and so $a > 0$. Since $a + b \geq -c_1$ we have $b \geq -c_1 - a$, which means $ta + b \geq (t - 1)a - c_1 > c_2$, contradicting $|ta + b| \leq c_2$. Instead, suppose that $a < -(c_1 + c_2)/(t - 1)$ and so $a < 0$. Then $a + b \leq c_1$ means $b \leq c_1 - a$ and so $ta + b \leq (t - 1)a + c_1 < -c_2$, again contradicting $|ta + b| \leq c_2$.

The bound on $|b|$ follows from similar reasoning. First suppose that $b > (tc_1 + c_2)/(t - 1)$ and so $b > 0$. Since $a + b \leq c_1$ we have $a \leq c_1 - b$ and so $ta + b \leq tc_1 - (t - 1)b < -c_2$, contradicting $|ta + b| \leq c_2$. Instead suppose that $b < -(tc_1 + c_2)/(t - 1)$ and so $b < 0$. Then since $a + b \geq -c_1$ we have $a \geq -c_1 - b$ and so $ta + b \geq -tc_1 - (t - 1)b > c_2$, again contradicting $|ta + b| \leq c_2$. \square

We can now give our bound on the model Hessian. In the existing (unconstrained) theory, this is presented as a bound on $\|H\|$ [16, Theorem 5.7], but here we only need to consider specific Rayleigh-type quotients.

Lemma 4.4. *Suppose f and C satisfy Assumptions 2.1 and 2.2 respectively. Then if $\mathbf{x} \in C$, $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ is Λ -poised for underdetermined quadratic interpolation in $B(\mathbf{x}, \Delta) \cap C$ and $\|\mathbf{y}_t - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ for all $t = 1, \dots, p$ and some $\beta > 0$, then the model m generated by (4.2) has Hessian H satisfying*

$$\max_{s,t=1,\dots,p} \frac{|(\mathbf{y}_s - \mathbf{x})^T H(\mathbf{y}_t - \mathbf{x})|}{\beta^2 \min(\Delta, 1)^2} \leq \kappa_H := L_{\nabla f p} [8\Lambda\beta^2 + 36\Lambda\beta + 58\Lambda + 6]. \quad (4.15)$$

Proof. First, fix $u \in \{1, \dots, p\}$ and consider the associated Lagrange polynomial

$$\ell_u(\mathbf{y}) = c_u + \mathbf{g}_u^T(\mathbf{y} - \mathbf{x}) + \frac{1}{2}(\mathbf{y} - \mathbf{x})^T H_u(\mathbf{y} - \mathbf{x}). \quad (4.16)$$

Additionally, fix $s, t \in \{1, \dots, p\}$, and we will first provide a bound on

$$|(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_t - \mathbf{x})|. \quad (4.17)$$

To do this, we consider the value of ℓ_u at five different points: \mathbf{x} , \mathbf{y}_s , \mathbf{y}_t , plus

$$\hat{\mathbf{y}}_s := \mathbf{x} + \frac{1}{\hat{\beta}}(\mathbf{y}_s - \mathbf{x}), \quad \text{and} \quad \hat{\mathbf{y}}_t := \mathbf{x} + \frac{1}{\hat{\beta}}(\mathbf{y}_t - \mathbf{x}), \quad (4.18)$$

where $\hat{\beta} := \max(\beta, 2)$. Since $\|\mathbf{y}_s - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ we have $\|\hat{\mathbf{y}}_s - \mathbf{x}\| \leq \frac{\beta}{\hat{\beta}} \min(\Delta, 1) \leq \min(\Delta, 1)$, and $\hat{\mathbf{y}}_s \in C$ by convexity, since $\mathbf{x}, \mathbf{y}_s \in C$. Similarly, we also have $\hat{\mathbf{y}}_t \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$. For these points, from the definition of Λ -poisedness we know $|\ell_u(\mathbf{x})|, |\ell_u(\hat{\mathbf{y}}_s)|, |\ell_u(\hat{\mathbf{y}}_t)| \leq \Lambda$ and by definition of Lagrange polynomials we have $\ell_u(\mathbf{y}_s), \ell_u(\mathbf{y}_t) \in \{0, 1\}$ and so $|\ell_u(\mathbf{y}_s)|, |\ell_u(\mathbf{y}_t)| \leq 1$.

From $|\ell_u(\mathbf{x})| \leq \Lambda$ we have $|c_u| \leq \Lambda$, and so $|\ell_u(\hat{\mathbf{y}}_s)| \leq \Lambda$ implies

$$\Lambda \geq \left| c_u + \mathbf{g}_u^T(\hat{\mathbf{y}}_s - \mathbf{x}) + \frac{1}{2}(\hat{\mathbf{y}}_s - \mathbf{x})^T H_u(\hat{\mathbf{y}}_s - \mathbf{x}) \right|, \quad (4.19)$$

$$= \left| c_u + \frac{1}{\hat{\beta}} \mathbf{g}_u^T(\mathbf{y}_s - \mathbf{x}) + \frac{1}{2\hat{\beta}^2}(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_s - \mathbf{x}) \right|, \quad (4.20)$$

$$\geq \frac{1}{\hat{\beta}^2} \left| \hat{\beta} \mathbf{g}_u^T(\mathbf{y}_s - \mathbf{x}) + \frac{1}{2}(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_s - \mathbf{x}) \right| - |c_u|, \quad (4.21)$$

where the last line follows from the reverse triangle inequality. Together with $|\ell_u(\mathbf{y}_s)| \leq 1$, we get

$$\left| \mathbf{g}_u^T(\mathbf{y}_s - \mathbf{x}) + \frac{1}{2}(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_s - \mathbf{x}) \right| \leq 1 + |c_u| \leq \Lambda + 1, \quad (4.22)$$

$$\left| \hat{\beta} \mathbf{g}_u^T(\mathbf{y}_s - \mathbf{x}) + \frac{1}{2}(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_s - \mathbf{x}) \right| \leq 2\Lambda\hat{\beta}^2. \quad (4.23)$$

Since $\hat{\beta} > 1$ by definition, applying Lemma 4.3 we conclude

$$|\mathbf{g}_u^T(\mathbf{y}_s - \mathbf{x})| \leq \frac{\Lambda + 1 + 2\Lambda\hat{\beta}^2}{\hat{\beta} - 1}, \quad (4.24)$$

$$\frac{1}{2} |(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_s - \mathbf{x})| \leq \frac{\hat{\beta}(\Lambda + 1) + 2\Lambda\hat{\beta}}{\hat{\beta} - 1}. \quad (4.25)$$

Since s was arbitrary, the same inequalities hold with \mathbf{y}_s replaced by \mathbf{y}_t .

Now, consider the point

$$\hat{\mathbf{y}}_{s,t} := \mathbf{x} + \frac{1}{2}(\hat{\mathbf{y}}_s - \mathbf{x}) + \frac{1}{2}(\hat{\mathbf{y}}_t - \mathbf{x}). \quad (4.26)$$

Since $\mathbf{x}, \hat{\mathbf{y}}_s, \hat{\mathbf{y}}_t \in C$, we have $\hat{\mathbf{y}}_{s,t} \in C$, and also

$$\|\hat{\mathbf{y}}_{s,t} - \mathbf{x}\| \leq \frac{1}{2}\|\hat{\mathbf{y}}_s - \mathbf{x}\| + \frac{1}{2}\|\hat{\mathbf{y}}_t - \mathbf{x}\| \leq \min(\Delta, 1), \quad (4.27)$$

and so $|\ell_u(\hat{\mathbf{y}}_{s,t})| \leq \Lambda$. Written in full, this is

$$\left| c_u + \frac{1}{2\hat{\beta}} \mathbf{g}_u^T(\mathbf{y}_s - \mathbf{x}) + \frac{1}{2\hat{\beta}} \mathbf{g}_u^T(\mathbf{y}_t - \mathbf{x}) + \frac{1}{2} \left(\frac{1}{2\hat{\beta}}(\mathbf{y}_s - \mathbf{x}) + \frac{1}{2\hat{\beta}}(\mathbf{y}_t - \mathbf{x}) \right)^T H_u \left(\frac{1}{2\hat{\beta}}(\mathbf{y}_s - \mathbf{x}) + \frac{1}{2\hat{\beta}}(\mathbf{y}_t - \mathbf{x}) \right) \right| \leq \Lambda. \quad (4.28)$$

That is,

$$\begin{aligned} \frac{1}{4\hat{\beta}^2} |(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_t - \mathbf{x})| &\leq \Lambda + |c_u| + \frac{1}{2\hat{\beta}} (|\mathbf{g}_u^T(\mathbf{y}_s - \mathbf{x})| + |\mathbf{g}_u^T(\mathbf{y}_t - \mathbf{x})|) \\ &\quad + \frac{1}{8\hat{\beta}^2} (|(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_s - \mathbf{x})| + |(\mathbf{y}_t - \mathbf{x})^T H_u(\mathbf{y}_t - \mathbf{x})|). \end{aligned} \quad (4.29)$$

Applying $|c_u| \leq \Lambda$, (4.24) and (4.25), we conclude

$$\frac{1}{4\hat{\beta}^2} |(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_t - \mathbf{x})| \leq 2\Lambda + \frac{\Lambda + 1 + 2\Lambda\hat{\beta}^2}{\hat{\beta}(\hat{\beta} - 1)} + \frac{\hat{\beta}(\Lambda + 1) + 2\Lambda\hat{\beta}}{2\hat{\beta}^2(\hat{\beta} - 1)}, \quad (4.30)$$

or

$$|(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_t - \mathbf{x})| \leq \tilde{\kappa}, \quad (4.31)$$

for all $s, t, u = 1, \dots, p$, where

$$\tilde{\kappa} := 8\Lambda\hat{\beta}^2 + \frac{4\Lambda\hat{\beta} + 4\hat{\beta} + 8\Lambda\hat{\beta}^3}{\hat{\beta} - 1} + \frac{2\hat{\beta}(\Lambda + 1) + 4\Lambda\hat{\beta}}{\hat{\beta} - 1} = 8\Lambda\hat{\beta}^2 + \frac{8\Lambda\hat{\beta}^3 + 10\Lambda\hat{\beta} + 6\hat{\beta}}{\hat{\beta} - 1}. \quad (4.32)$$

More simply, we have

$$\tilde{\kappa} = 8\Lambda\hat{\beta}^2 + \frac{8\Lambda(\hat{\beta} - 1)^3 + 24\Lambda(\hat{\beta} - 1)^2 + (34\Lambda + 6)(\hat{\beta} - 1) + 18\Lambda + 6}{\hat{\beta} - 1}, \quad (4.33)$$

$$\leq 8\Lambda(\beta + 2)^2 + 8\Lambda(\beta + 1)^2 + 24\Lambda(\beta + 1) + 34\Lambda + 6 + 18\Lambda + 6, \quad (4.34)$$

$$= 16\Lambda\beta^2 + 72\Lambda\beta + 116\Lambda + 12, \quad (4.35)$$

using $2 \leq \hat{\beta} = \max(\beta, 2) \leq \beta + 2$ to get the inequality. Now let us turn our attention to the model m (4.2). Note that we may add/subtract a linear function to $f(\mathbf{y})$ without changing H (since this just changes c and \mathbf{g} in (4.2)). So, we consider the model \tilde{m} generated by interpolation to $\tilde{f}(\mathbf{y}) := f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})$. From (4.10) we may write

$$H = \tilde{H} = \sum_{u=1}^p \tilde{f}(\mathbf{y}_u) H_u, \quad (4.36)$$

where H_u are the Hessians of the Lagrange polynomials. From Assumption 2.1, we also have $|\tilde{f}(\mathbf{y})| \leq \frac{L_{\nabla f}}{2} \|\mathbf{y} - \mathbf{x}\|^2$. Then for any $s, t = 1, \dots, p$ we conclude

$$|(\mathbf{y}_s - \mathbf{x})^T H(\mathbf{y}_t - \mathbf{x})| \leq \sum_{u=1}^p |\tilde{f}(\mathbf{y}_u)| \cdot |(\mathbf{y}_s - \mathbf{x})^T H_u(\mathbf{y}_t - \mathbf{x})|, \quad (4.37)$$

$$\leq \sum_{u=1}^p \frac{L_{\nabla f}}{2} \tilde{\kappa} \|\mathbf{y}_u - \mathbf{x}\|^2, \quad (4.38)$$

$$\leq \frac{L_{\nabla f}}{2} \tilde{\kappa} p \beta^2 \min(\Delta, 1)^2. \quad (4.39)$$

and we are done after applying (4.35). \square

Remark 4.5. In the unconstrained case, [16, Theorem 5.7] gives the bound

$$\|H\| \leq 4(p+1)\sqrt{\frac{(n+1)(n+2)}{2}} L_{\nabla f} \Lambda \max(1, \Delta_{\max}^2), \quad (4.40)$$

where Δ_{\max} is an upper bound for Δ . This result assumes $\|\mathbf{y}_t - \mathbf{x}\| \leq \Delta$, and so to match our assumptions it requires $\|\mathbf{y}_t - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ where $\beta = \Delta_{\max}$. Thus we see that Lemma 4.4 improves on [16, Theorem 5.7] by a factor of $\mathcal{O}(n)$.

Next, we can prove an analogous result to Lemma 3.3.

Lemma 4.6. *Suppose the assumptions of Lemma 4.4 are satisfied. Then the model m generated by (4.2) satisfies*

$$|c + \mathbf{g}^T(\mathbf{y} - \mathbf{x}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| \leq \frac{1}{2} p^{3/2} \Lambda (L_{\nabla f} + \kappa_H) \beta^2 \min(\Delta, 1)^2, \quad (4.41)$$

for all $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$. Furthermore if $\mathbf{x} \in C$, we have

$$|c - f(\mathbf{x})| \leq \frac{1}{2} p^{3/2} \Lambda (L_{\nabla f} + \kappa_H) \beta^2 \min(\Delta, 1)^2, \quad (4.42)$$

and

$$|(\mathbf{g} - \nabla f(\mathbf{x}))^T(\mathbf{y} - \mathbf{x})| \leq p^{3/2} \Lambda (L_{\nabla f} + \kappa_H) \beta^2 \min(\Delta, 1)^2, \quad (4.43)$$

for all $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$.

Proof. Fix $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$. From Lemma 4.2, our interpolation set is $\sqrt{p} \Lambda$ -poised for linear regression. Therefore there exist constants $\{\alpha_t(\mathbf{y})\}_{t=1}^p$ such that

$$\begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix} = \sum_{t=1}^p \alpha_t(\mathbf{y}) \begin{bmatrix} 1 \\ \mathbf{y}_t - \mathbf{x} \end{bmatrix}, \quad (4.44)$$

where $|\alpha_t(\mathbf{y})| = |\ell_t^{\text{reg}}(\mathbf{y})| \leq \sqrt{p} \Lambda$ (provided the minimal norm solution is taken as in (3.15)). Thus we have

$$\begin{aligned} & |c + \mathbf{g}^T(\mathbf{y} - \mathbf{x}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| \\ &= \left| \begin{bmatrix} c - f(\mathbf{x}) \\ \mathbf{g} - \nabla f(\mathbf{x}) \end{bmatrix}^T \begin{bmatrix} 1 \\ \mathbf{y} - \mathbf{x} \end{bmatrix} \right|, \end{aligned} \quad (4.45)$$

$$\leq \sum_{t=1}^p \sqrt{p} \Lambda \left| \begin{bmatrix} c - f(\mathbf{x}) \\ \mathbf{g} - \nabla f(\mathbf{x}) \end{bmatrix}^T \begin{bmatrix} 1 \\ \mathbf{y}_t - \mathbf{x} \end{bmatrix} \right|, \quad (4.46)$$

$$= \sqrt{p} \Lambda \sum_{t=1}^p \left| m(\mathbf{y}_t) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y}_t - \mathbf{x}) - \frac{1}{2}(\mathbf{y}_t - \mathbf{x})^T H(\mathbf{y}_t - \mathbf{x}) \right|, \quad (4.47)$$

$$\leq \sqrt{p} \Lambda \sum_{t=1}^p \left[|f(\mathbf{y}_t) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y}_t - \mathbf{x})| + \frac{1}{2} |(\mathbf{y}_t - \mathbf{x})^T H(\mathbf{y}_t - \mathbf{x})| \right], \quad (4.48)$$

$$\leq p^{3/2} \Lambda \left(\frac{L_{\nabla f}}{2} \beta^2 \min(\Delta, 1)^2 + \frac{\kappa_H}{2} \beta^2 \min(\Delta, 1)^2 \right), \quad (4.49)$$

using Assumption 2.1 and Lemma 4.4 in the last line, and we have (4.41). To get (4.42), we just take $\mathbf{y} = \mathbf{x}$ in (4.41). Finally, to get (4.43), we use

$$|(\mathbf{g} - \nabla f(\mathbf{x}))^T(\mathbf{y} - \mathbf{x})| \leq |c + \mathbf{g}^T(\mathbf{y} - \mathbf{x}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| + |c - f(\mathbf{x})|, \quad (4.50)$$

together with (4.41) and (4.42). \square

Finally, we can give our fully linear error bounds for underdetermined quadratic interpolation models.

Theorem 4.7. *Suppose f and C satisfy Assumptions 2.1 and 2.2 respectively. Then if $\mathbf{x} \in C$, $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ is Λ -poised for underdetermined quadratic interpolation in $B(\mathbf{x}, \Delta) \cap C$ and $\|\mathbf{y}_t - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ for all $t = 1, \dots, p$ and some $\beta > 0$, then the model m generated by (4.2) is fully linear in $B(\mathbf{x}, \Delta)$ (in the sense of Definition 2.4) with constants*

$$\kappa_{\text{ef}} = \frac{1}{2}L_{\nabla f} + \frac{3}{2}p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2 + \frac{1}{2}p\Lambda^2\kappa_H\beta^2, \quad \text{and} \quad \kappa_{\text{eg}} = p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2, \quad (4.51)$$

in (2.4), where κ_H is defined in Lemma 4.4.

Proof. First fix $\mathbf{y} \in B(\mathbf{x}, \Delta) \cap C$ and we will derive κ_{ef} by considering the cases $\Delta \geq 1$ and $\Delta < 1$ separately. If $\Delta \geq 1$ then $\hat{\mathbf{y}} := \mathbf{x} + \Delta^{-1}(\mathbf{y} - \mathbf{x})$ is in $B(\mathbf{x}, 1) = B(\mathbf{x}, \min(\Delta, 1))$ and $\hat{\mathbf{y}} \in C$ since C is convex. We then apply Lemma 4.6 to get

$$|(\mathbf{y} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| = \Delta|(\hat{\mathbf{y}} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))|, \quad (4.52)$$

$$\leq p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2\Delta \min(\Delta, 1)^2, \quad (4.53)$$

$$\leq p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2\Delta^2, \quad (4.54)$$

where the last inequality follows from $\min(\Delta, 1)^2 = 1 \leq \Delta$. Also, from (4.44) we have $\hat{\mathbf{y}} - \mathbf{x} = \sum_{t=1}^p \alpha_t(\hat{\mathbf{y}})(\mathbf{y}_t - \mathbf{x})$ with $|\alpha_t(\hat{\mathbf{y}})| \leq \sqrt{p}\Lambda$, and so from Lemma 4.4,

$$|(\mathbf{y} - \mathbf{x})^T H(\mathbf{y} - \mathbf{x})| = \Delta^2|(\hat{\mathbf{y}} - \mathbf{x})^T H(\hat{\mathbf{y}} - \mathbf{x})|, \quad (4.55)$$

$$\leq \Delta^2 \sum_{s,t=1}^p |\alpha_s(\hat{\mathbf{y}})\alpha_t(\hat{\mathbf{y}})(\mathbf{y}_s - \mathbf{x})^T H(\mathbf{y}_t - \mathbf{x})|, \quad (4.56)$$

$$\leq p\Lambda^2\kappa_H\beta^2\Delta^2 \min(\Delta, 1)^2, \quad (4.57)$$

$$= p\Lambda^2\kappa_H\beta^2\Delta^2, \quad (4.58)$$

again using $\Delta \geq 1$ in the last line.

Instead, if $\Delta < 1$ then $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$ and we apply Lemma 4.6 directly to get

$$|(\mathbf{y} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| \leq \frac{1}{2}p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2 \min(\Delta, 1)^2, \quad (4.59)$$

$$= p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2\Delta^2. \quad (4.60)$$

Also, using Lemma 4.4 we have

$$|(\mathbf{y} - \mathbf{x})^T H(\mathbf{y} - \mathbf{x})| \leq \sum_{t=1}^p |\alpha_s(\mathbf{y})\alpha_t(\mathbf{y})(\mathbf{y}_s - \mathbf{x})^T H(\mathbf{y}_t - \mathbf{x})|, \quad (4.61)$$

$$\leq p\Lambda^2\kappa_H\beta^2 \min(\Delta, 1)^2, \quad (4.62)$$

$$= p\Lambda^2\kappa_H\beta^2\Delta^2. \quad (4.63)$$

In either case, we use (4.54) and (4.58), or (4.60) and (4.63), with Assumption 2.1 and Lemma 4.6 to get

$$|f(\mathbf{y}) - m(\mathbf{y})| \leq |f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})| + |c + \mathbf{g}^T(\mathbf{y} - \mathbf{x}) + \frac{1}{2}(\mathbf{y} - \mathbf{x})^T H(\mathbf{y} - \mathbf{x}) - f(\mathbf{x}) - \nabla f(\mathbf{x})^T(\mathbf{y} - \mathbf{x})|, \quad (4.64)$$

$$\leq \frac{1}{2}L_{\nabla f}\Delta^2 + |c - f(\mathbf{x})| + |(\mathbf{y} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| + \frac{1}{2}|(\mathbf{y} - \mathbf{x})^T H(\mathbf{y} - \mathbf{x})|, \quad (4.65)$$

$$\leq \frac{1}{2}L_{\nabla f}\Delta^2 + \frac{1}{2}p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2 \min(\Delta, 1)^2 + p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2\Delta^2 + \frac{1}{2}p\Lambda^2\kappa_H\beta^2\Delta^2, \quad (4.66)$$

and we get the value of κ_{ef} after $\min(\Delta, 1) \leq \Delta$.

To get κ_{eg} we now fix an arbitrary $\tilde{\mathbf{y}} \in B(\mathbf{x}, 1) \cap C$ and again consider the cases $\Delta \geq 1$ and $\Delta < 1$ separately. First, if $\Delta \geq 1$, then $\tilde{\mathbf{y}} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$, and applying Lemma 4.6 we get

$$|(\tilde{\mathbf{y}} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| \leq p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2 \min(\Delta, 1)^2, \quad (4.67)$$

$$\leq p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2\Delta, \quad (4.68)$$

since $\min(\Delta, 1)^2 = 1 \leq \Delta$. Alternatively, if $\Delta < 1$ then the convexity of C implies that $\hat{\mathbf{y}} := \mathbf{x} + \Delta(\tilde{\mathbf{y}} - \mathbf{x}) \in B(\mathbf{x}, \Delta) \cap C = B(\mathbf{x}, \min(\Delta, 1)) \cap C$. Again we apply Lemma 4.6 and get

$$|(\tilde{\mathbf{y}} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))| = \Delta^{-1}|(\hat{\mathbf{y}} - \mathbf{x})^T(\mathbf{g} - \nabla f(\mathbf{x}))|, \quad (4.69)$$

$$\leq p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2\Delta^{-1} \min(\Delta, 1)^2, \quad (4.70)$$

$$= p^{3/2}\Lambda(L_{\nabla f} + \kappa_H)\beta^2\Delta. \quad (4.71)$$

The value for κ_{eg} then follows from (4.68) and (4.71). \square

Remark 4.8. The fully linear error bounds for the unconstrained case [14, 16] give $\kappa_{\text{ef}}, \kappa_{\text{eg}} = \mathcal{O}(p^{3/2}\Lambda(L_{\nabla f} + \kappa_H))$, so Theorem 4.7 matches the bound for κ_{eg} up to a factor of β^2 (but with a value of κ_H which is $\mathcal{O}(n)$ smaller than the unconstrained case; see Remark 4.5). Our value of κ_{ef} has an extra term of size $\mathcal{O}(p\Lambda^2\kappa_H\beta^2)$ and so may be larger depending on the relative sizes of p and Λ .

5 Constructing Λ -Poised Quadratic Models

To meet all the requirements of Algorithm 1, we require procedures which can verify whether or not a given model is C -fully linear, and if not, modify it to be C -fully linear. From Theorem 4.7 it is clear that it suffices to verify/ensure that $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ is Λ -poised for underdetermined quadratic interpolation in $B(\mathbf{x}, \Delta) \cap C$ and $\|\mathbf{y}_t - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ for all $t = 0, \dots, p$.

In the first case—verifying whether not the interpolation set is Λ -poised—we may follow the approach in [22, Section 4.3.1] and simply maximize/minimize each ℓ_t in $B(\mathbf{x}, \min(\Delta, 1)) \cap C$. Checking $\|\mathbf{y}_t - \mathbf{x}\| \leq \beta \min(\Delta, 1)$ is straightforward. Although maximizing/minimizing ℓ_t , a possibly nonconvex quadratic function, in $B(\mathbf{x}, \min(\Delta, 1)) \cap C$ may not be straightforward depending on the structure of C , we only need to know if the solution is above/below Λ and not the exact solution. Existing solvers, including potentially global optimization solvers, are sufficient for this.

We now consider the situation where $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ is not Λ -poised, and we wish to construct a new interpolation set which is Λ -poised. There are two possible scenarios:

- The associated matrix F in (4.3) is not invertible; or
- The matrix F is invertible but $|\ell_t(\mathbf{y})| > \Lambda$ for some $t = 1, \dots, p$ and $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$.

We will describe how to handle both of these situations in Section 5.2, but we will first need a technical result about how the matrix F in (4.3) changes as interpolation points are changed.

5.1 Updating the matrix F

In this section we analyze changes in the determinant of the matrix F as interpolation points are updated. These results are based on ideas from [26, Section 4], which considers changes in F^{-1} from updating interpolation points, rather than $\det(F)$.

Lemma 5.1. *If A is a symmetric, invertible matrix and we form \tilde{A} by changing the t -th row and column of A to $\tilde{\mathbf{v}}$, then*

$$\det(\tilde{A}) = [(\mathbf{e}_t^T A^{-1} \tilde{\mathbf{v}})^2 + (\mathbf{e}_t^T A^{-1} \mathbf{e}_t) \tilde{\mathbf{v}}^T (\mathbf{e}_t - A^{-1} \tilde{\mathbf{v}})] \det(A). \quad (5.1)$$

Proof. Denote the t -th row and column of A by $\mathbf{v} := A\mathbf{e}_t$. We first note that \tilde{A} can be written as a symmetric rank-3 update of A :

$$\tilde{A} = A + (\tilde{\mathbf{v}} - \mathbf{v})\mathbf{e}_t^T + \mathbf{e}_t(\tilde{\mathbf{v}} - \mathbf{v})^T - (\tilde{v}_t - v_t)\mathbf{e}_t\mathbf{e}_t^T. \quad (5.2)$$

The first two update terms add $\tilde{\mathbf{v}} - \mathbf{v}$ to the t -th column and row respectively, and the last term corrects for the double update of $A_{t,t}$ from the first two updates. We can then apply the matrix determinant lemma² to get

$$\frac{\det(\tilde{A})}{\det(A)} = \det \left(\begin{bmatrix} 1 + \mathbf{e}_t^T A^{-1}(\tilde{\mathbf{v}} - \mathbf{v}) & \mathbf{e}_t^T A^{-1}\mathbf{e}_t & \mathbf{e}_t^T A^{-1}\mathbf{e}_t \\ (\tilde{\mathbf{v}} - \mathbf{v})^T A^{-1}(\tilde{\mathbf{v}} - \mathbf{v}) & 1 + (\tilde{\mathbf{v}} - \mathbf{v})^T A^{-1}\mathbf{e}_t & (\tilde{\mathbf{v}} - \mathbf{v})^T A^{-1}\mathbf{e}_t \\ -(\tilde{v}_t - v_t)\mathbf{e}_t^T A^{-1}(\tilde{\mathbf{v}} - \mathbf{v}) & -(\tilde{v}_t - v_t)\mathbf{e}_t^T A^{-1}\mathbf{e}_t & 1 - (\tilde{v}_t - v_t)\mathbf{e}_t^T A^{-1}\mathbf{e}_t \end{bmatrix} \right), \quad (5.3)$$

$$= \det \left(\begin{bmatrix} 1 + \tau & \alpha & \alpha \\ \beta & 1 + \tau & \tau \\ -\gamma\tau & -\gamma\alpha & 1 - \gamma\alpha \end{bmatrix} \right), \quad (5.4)$$

where $\alpha := \mathbf{e}_t^T A^{-1}\mathbf{e}_t$, $\beta := (\tilde{\mathbf{v}} - \mathbf{v})^T A^{-1}(\tilde{\mathbf{v}} - \mathbf{v})$, $\gamma := \tilde{v}_t - v_t$ and $\tau := \mathbf{e}_t^T A^{-1}(\tilde{\mathbf{v}} - \mathbf{v}) = (\tilde{\mathbf{v}} - \mathbf{v})^T A^{-1}\mathbf{e}_t$ since A^{-1} is symmetric. Directly computing, we get

$$\frac{\det(\tilde{A})}{\det(A)} = (1 + \tau) [(1 + \tau)(1 - \gamma\alpha) + \tau\gamma\alpha] - \beta [\alpha(1 - \gamma\alpha) + \gamma\alpha^2] - \gamma\tau [\alpha\tau - \alpha(1 + \tau)], \quad (5.5)$$

$$= (1 + \tau) [1 + \tau - \gamma\alpha] - \beta\alpha + \gamma\tau\alpha, \quad (5.6)$$

$$= (1 + \tau)^2 - \alpha(\gamma + \beta). \quad (5.7)$$

Finally, since $\mathbf{v} = A\mathbf{e}_t$, we get

$$\tau = \mathbf{e}_t^T A^{-1}(\tilde{\mathbf{v}} - \mathbf{v}) = \mathbf{e}_t^T A^{-1}\tilde{\mathbf{v}} - \underbrace{\mathbf{e}_t^T A^{-1}\mathbf{v}}_{=\mathbf{e}_t} = \mathbf{e}_t^T A^{-1}\tilde{\mathbf{v}} - 1, \quad (5.8)$$

and

$$\gamma + \beta = (\tilde{\mathbf{v}} - \mathbf{v})^T \mathbf{e}_t + (\tilde{\mathbf{v}} - \mathbf{v})^T A^{-1}(\tilde{\mathbf{v}} - \mathbf{v}), \quad (5.9)$$

$$= \tilde{\mathbf{v}}^T \mathbf{e}_t - \mathbf{v}^T \mathbf{e}_t + \tilde{\mathbf{v}}^T A^{-1}\tilde{\mathbf{v}} - 2\tilde{\mathbf{v}}^T A^{-1}\mathbf{v} + \mathbf{v}^T A^{-1}\mathbf{v}, \quad (5.10)$$

$$= \tilde{\mathbf{v}}^T \mathbf{e}_t - \mathbf{v}^T \mathbf{e}_t + \tilde{\mathbf{v}}^T A^{-1}\tilde{\mathbf{v}} - 2\tilde{\mathbf{v}}^T \mathbf{e}_t + \mathbf{v}^T \mathbf{e}_t, \quad (5.11)$$

$$= \tilde{\mathbf{v}}^T (A^{-1}\tilde{\mathbf{v}} - \mathbf{e}_t). \quad (5.12)$$

Our result then follows from (5.7) combined with the definition of α , (5.8) and (5.12). \square

Theorem 5.2. *Suppose the interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \in \mathbb{R}^n$ is such that the matrix F in (4.3) is invertible. If the point \mathbf{y}_t for some $t \in \{1, \dots, p\}$ is changed to a new point $\mathbf{y} \in \mathbb{R}^n$, then the new interpolation set yields a new matrix \tilde{F} in (4.3) which satisfies*

$$|\det(\tilde{F})| \geq \ell_t(\mathbf{y})^2 |\det(F)|. \quad (5.13)$$

Proof. If $\ell_t(\mathbf{y}) = 0$, then the result holds trivially, so we assume without loss of generality that $\ell_t(\mathbf{y}) \neq 0$.

Changing the interpolation point \mathbf{y}_t to \mathbf{y} requires replacing the t -th row and column of F by $\tilde{\mathbf{v}}$, where

$$\tilde{v}_s = \frac{1}{2} [(\mathbf{y} - \mathbf{x})^T (\mathbf{y}_s - \mathbf{x})]^2, \quad \text{if } s \in \{1, \dots, p\} \text{ except for } s = t, \quad (5.14a)$$

$$\tilde{v}_t = \frac{1}{2} [(\mathbf{y} - \mathbf{x})^T (\mathbf{y} - \mathbf{x})]^2, \quad (5.14b)$$

$$\tilde{v}_{p+2} = 1, \quad (5.14c)$$

$$\tilde{v}_{p+2+i} = [\mathbf{y} - \mathbf{x}]_i, \quad \forall i = 1, \dots, n. \quad (5.14d)$$

²That is, $\det(A + UV^T) = \det(I + V^T A^{-1}U) \det(A)$ if A is invertible, e.g. [21, Corollary 18.1.4].

Algorithm 2 Construct an underdetermined quadratic interpolation set with invertible F (4.3).

Input: Trust region center $\mathbf{x} \in C$ and radius $\Delta > 0$, number of interpolation points $p \in \{n+2, \dots, (n+1)(n+2)/2\}$.

- 1: Generate an initial collection of p points $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset B(\mathbf{x}, \min(\Delta, 1))$ using the method from [29, Section 2].
 - 2: **while** there exists $t \in \{1, \dots, p\}$ with $\mathbf{y}_t \notin C$ **do**
 - 3: Calculate Lagrange polynomial ℓ_t and find $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$ with $\ell_t(\mathbf{y}) \neq 0$.
 - 4: Replace the interpolation point \mathbf{y}_t by \mathbf{y} .
 - 5: **end while**
 - 6: **return** interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$.
-

That is, $\tilde{\mathbf{v}}$ is the same as $\phi(\mathbf{y})$ from (4.8) except for the t -th entry. Specifically, we have

$$\tilde{\mathbf{v}} = \phi(\mathbf{y}) + \eta_t \mathbf{e}_t, \quad (5.15)$$

where

$$\eta_t = \frac{1}{2} [(\mathbf{y} - \mathbf{x})^T (\mathbf{y} - \mathbf{x})]^2 - \frac{1}{2} [(\mathbf{y} - \mathbf{x})^T (\mathbf{y}_t - \mathbf{x})]^2 = \frac{1}{2} \|\mathbf{y} - \mathbf{x}\|^4 - \phi(\mathbf{y})^T \mathbf{e}_t. \quad (5.16)$$

Given this, Lemma 5.1 gives us (denoting $\alpha_t := \mathbf{e}_t^T F^{-1} \mathbf{e}_t$ for notational convenience)

$$\frac{\det(\tilde{F})}{\det(F)} = (\mathbf{e}_t^T F^{-1} \phi(\mathbf{y}) + \eta_t \mathbf{e}_t^T F^{-1} \mathbf{e}_t)^2 + (\mathbf{e}_t^T F^{-1} \mathbf{e}_t) (\phi(\mathbf{y}) + \eta_t \mathbf{e}_t)^T (\mathbf{e}_t - F^{-1} (\phi(\mathbf{y}) + \eta_t \mathbf{e}_t)), \quad (5.17)$$

$$= (\ell_t(\mathbf{y}) + \eta_t \alpha_t)^2 + \alpha_t (\phi(\mathbf{y})^T \mathbf{e}_t - \phi(\mathbf{y})^T F^{-1} \phi(\mathbf{y}) - \eta_t \ell_t(\mathbf{y}) + \eta_t - \eta_t \ell_t(\mathbf{y}) - \eta_t^2 \alpha_t), \quad (5.18)$$

$$= \ell_t(\mathbf{y})^2 + \alpha_t (\phi(\mathbf{y})^T \mathbf{e}_t - \phi(\mathbf{y})^T F^{-1} \phi(\mathbf{y}) + \eta_t). \quad (5.19)$$

So from the definition of η_t , we get

$$\det(\tilde{F}) = (\ell_t(\mathbf{y})^2 + \alpha_t \beta_t) \det(F), \quad (5.20)$$

where $\beta_t := \frac{1}{2} \|\mathbf{y} - \mathbf{x}\|^4 - \phi(\mathbf{y})^T F^{-1} \phi(\mathbf{y})$. Finally, [26, Lemma 2] gives us $\alpha_t, \beta_t \geq 0$ provided $\ell_t(\mathbf{y}) \neq 0$, which gives us the result.³ \square

Remark 5.3. Theorem 5.2 does not require any of the points $\mathbf{y}_1, \dots, \mathbf{y}_p$ or \mathbf{y} to be in C . We will use this when constructing interpolation sets for which F is invertible (Algorithm 2).

Remark 5.4. A similar analysis to the above holds for linear interpolation ($p = n$), where Theorem 5.2 simplifies to $|\det(\tilde{F})| = |\ell_t(\mathbf{y})| \cdot |\det(F)|$ as shown in [8, Section 4.2]. This allows us to use an approach like Algorithm 3 below to construct Λ -poised sets for linear interpolation, a simpler method than the one outlined in [22, Section 4.3].

5.2 Constructing Λ -Poised Sets

We begin by describing how to construct an initial interpolation set, contained in $B(\mathbf{x}, \min(\Delta, 1)) \cap C$ for which F is invertible. The full process for this is given in Algorithm 2. We first generate an initial set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ in $B(\mathbf{x}, \min(\Delta, 1))$, but not necessarily in C , for which F is invertible. We can do this using the approach outlined in [29, Section 2], which chooses points of the form $\mathbf{x} \pm \min(\Delta, 1) \mathbf{e}_t$ or $\mathbf{x} \pm \min(\Delta, 1) \mathbf{e}_s \pm \min(\Delta, 1) \mathbf{e}_t$ for some $s, t \in \{1, \dots, n\}$ in a specific way. We then replace any of these points which are not in C by new points in $B(\mathbf{x}, \min(\Delta, 1)) \cap C$ while maintaining an invertible F .

³That $\ell_t(\mathbf{y}) \neq 0$ is not explicitly assumed in the statement of [26, Lemma 2] but is required as the proof relies on [26, Theorem, p. 196].

Algorithm 3 Construct a Λ -poised underdetermined quadratic interpolation set.

Input: Trust region center $\mathbf{x} \in C$ and radius $\Delta > 0$, number of interpolation points $p \in \{n+2, \dots, (n+1)(n+2)/2\}$, optional initial interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset C$, poisedness constant $\Lambda > 1$.

- 1: **if** initial interpolation set not provided or yields singular F (4.3) or $\max_{t=1, \dots, p} \|\mathbf{y}_t - \mathbf{x}\| > \min(\Delta, 1)$ **then**
 - 2: Set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ to be the output of Algorithm 2.
 - 3: **end if**
 - 4: **while** $\max_{\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C} \max_{t=1, \dots, p} |\ell_t(\mathbf{y})| > \Lambda$ **do**
 - 5: Find $t \in \{1, \dots, p\}$ and $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$ with $|\ell_t(\mathbf{y})| > \Lambda$.
 - 6: Replace the interpolation point \mathbf{y}_t by \mathbf{y} and recalculate Lagrange polynomials.
 - 7: **end while**
 - 8: **return** interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$.
-

Theorem 5.5. For any $\mathbf{x} \in C$, $\Delta > 0$ and $p \in \{n+2, \dots, (n+1)(n+2)/2\}$, the output of Algorithm 2 is an interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset B(\mathbf{x}, \min(\Delta, 1)) \cap C$ for which F (4.3) is invertible.

Proof. Firstly, Algorithm 2 terminates after at most p iterations, since an index t can only ever be selected once (after which the new $\mathbf{y}_t \in C$).

Next, all points in the output set were either originally generated in line 1, in which case they must have originally been in both $B(\mathbf{x}, \min(\Delta, 1))$ (from the construction in line 1) and C (in order to never be replaced in the main loop), or were a $\mathbf{y} \in B(\mathbf{x}, \min(\Delta, 1)) \cap C$ chosen in line 3. Hence $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset B(\mathbf{x}, \min(\Delta, 1)) \cap C$.

Finally, the initial collection $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ from [29, Section 2] generates an invertible F , with an explicit formula for F^{-1} given in [29, eqns. 2.11–2.15]. Hence the initial F satisfies $|\det(F)| > 0$. Then at each iteration replace \mathbf{y}_t with a point \mathbf{y} for which $\ell_t(\mathbf{y}) \neq 0$, and so the new F also satisfies $|\det(F)| > 0$ by Theorem 5.2. Thus the output interpolation set has an invertible F . \square

We can now present our algorithm to generate a Λ -poised set for minimum Frobenius norm interpolation, given in Algorithm 3. This algorithm is very similar to existing methods for unconstrained linear/quadratic (e.g. [16, Algorithm 6.3]) or constrained linear poisedness ([22, Algorithm 4.2]). Essentially we first ensure we have an interpolation set with invertible F (so we can construct the associated Lagrange polynomials), and we then iteratively find indices t and points \mathbf{y} with $|\ell_t(\mathbf{y})| > \Lambda$, replacing \mathbf{y}_t with \mathbf{y} .

Theorem 5.6. For any $\mathbf{x} \in C$, $\Delta > 0$, $p \in \{n+2, \dots, (n+1)(n+2)/2\}$ and $\Lambda > 1$, Algorithm 3 produces an interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ which is Λ -poised for minimum Frobenius norm interpolation in $B(\mathbf{x}, \Delta) \cap C$ and for which $\|\mathbf{y}_t - \mathbf{x}\| \leq \min(\Delta, 1)$ for all $t = 1, \dots, p$.

Proof. If Algorithm 3 terminates, then by the definition of the loop check it must satisfy the requirements of the theorem, so it suffices to show that the ‘while’ loop terminates.

At the start of the ‘while’ loop in line 4, we have that $\{\mathbf{y}_1, \dots, \mathbf{y}_p\} \subset B(\mathbf{x}, \min(\Delta, 1)) \cap C$ and F is invertible (either because the ‘if’ statement in line 1 evaluates to false or by Theorem 5.5). Hence at the start of the ‘while’ loop we have $|\det(F)| > 0$. Let us call this value $d_0 > 0$.

In each iteration of the ‘while’ loop, we increase $|\det(F)|$ by a factor of at least Λ^2 by Theorem 5.2. Hence after k iterations of the loop, we have $|\det(F)| \geq \Lambda^{2k} d_0$. However, since by construction our interpolation set is contained in the compact set $B(\mathbf{x}, \min(\Delta, 1)) \cap C$, there is a global upper bound d_{\max} on $|\det(F)|$ (which is a continuous function of the interpolation points). Hence the loop must terminate after at most $\log(d_{\max}/d_0)/(2 \log \Lambda)$ iterations. \square

Remark 5.7. In the unconstrained case $C = \mathbb{R}^n$, Theorem 5.6 and in particular its reliance on Theorem 5.2 clarifies the argument behind [16, Theorem 6.6].

Theorems 4.7 (with $\beta = 1$) and 5.6 then give us the following.

Corollary 5.8. *Suppose f and C satisfy Assumptions 2.1 and 2.2 respectively. Then for any $\mathbf{x} \in C$, $\Delta > 0$, $p \in \{n+2, \dots, (n+1)(n+2)/2\}$ and $\Lambda > 1$, Algorithm 3 produces an interpolation set $\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$ for which the model m generated by (4.2) is fully linear in $B(\mathbf{x}, \Delta)$.*

6 Conclusions and Future Work

In [22], a weaker notion of fully linear models was used to construct convergent algorithms for (1.1) (namely Algorithm 1), and it was shown that linear interpolation models constructed only with feasible points can yield such models. Here, we show that Algorithm 1 can also be implemented using underdetermined quadratic interpolation models in the minimum Frobenius sense of [26], which is a much more useful construction in practice. As an extra benefit, it also establishes the same results for linear regression models, as a strict generalization of the analysis in [22] that is independently useful in the case where only noisy objective evaluations are available.

The natural extension of this work is to the construction and analysis of second-order accurate models (“fully quadratic” in the language of [16]) built only using feasible points. A concrete implementation of Algorithm 1 with quadratic interpolation models would also be a useful way to extend the implementation from [22] (which only considers nonlinear least-squares objectives, for which linear interpolation models can be practically useful).

References

- [1] C. AUDET AND W. HARE, *Derivative-Free and Blackbox Optimization*, Springer Series in Operations Research and Financial Engineering, Springer, Cham, Switzerland, 2017.
- [2] F. AUGUSTIN AND Y. M. MARZOUK, *A trust-region method for derivative-free nonlinear constrained stochastic optimization*, arXiv preprint arXiv:1703.04156, (2017).
- [3] A. BECK, *First-Order Methods in Optimization*, MOS-SIAM Series on Optimization, MOS/SIAM, Philadelphia, 2017.
- [4] M. BENZI, G. H. GOLUB, AND J. LIESEN, *Numerical solution of saddle point problems*, Acta Numerica, 14 (2005), pp. 1–137.
- [5] S. C. BILLUPS, J. LARSON, AND P. GRAF, *Derivative-free optimization of expensive functions with computational error using weighted regression*, SIAM Journal on Optimization, 23 (2013), pp. 27–53.
- [6] C. CARTIS, J. FIALA, B. MARTEAU, AND L. ROBERTS, *Improving the flexibility and robustness of model-based derivative-free optimization solvers*, ACM Transactions on Mathematical Software, 45 (2019), pp. 32:1–41.
- [7] C. CARTIS, N. I. M. GOULD, AND P. L. TOINT, *An adaptive cubic regularization algorithm for nonconvex optimization with convex constraints and its function-evaluation complexity*, IMA Journal of Numerical Analysis, 32 (2012), pp. 1662–1695.
- [8] C. CARTIS AND L. ROBERTS, *A derivative-free Gauss-Newton method*, Mathematical Programming Computation, 11 (2019), pp. 631–674.
- [9] P. CONEJO, E. KARAS, AND L. PEDROSO, *A trust-region derivative-free algorithm for constrained optimization*, Optimization Methods and Software, 30 (2015), pp. 1126–1145.
- [10] P. CONEJO, E. KARAS, L. PEDROSO, A. RIBEIRO, AND M. SACHINE, *Global convergence of trust-region algorithms for convex constrained minimization without derivatives*, Applied Mathematics and Computation, 220 (2013), pp. 324–330.

- [11] A. CONN, K. SCHEINBERG, AND PH. TOINT, *A derivative free optimization algorithm in practice*, in 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO, USA, 1998, American Institute of Aeronautics and Astronautics.
- [12] A. R. CONN, N. I. M. GOULD, AND P. L. TOINT, *Trust-Region Methods*, vol. 1 of MPS-SIAM Series on Optimization, MPS/SIAM, Philadelphia, 2000.
- [13] A. R. CONN, K. SCHEINBERG, AND L. N. VICENTE, *Geometry of interpolation sets in derivative free optimization*, Mathematical Programming, 111 (2007), pp. 141–172.
- [14] ———, *Geometry of sample sets in derivative-free optimization: Polynomial regression and underdetermined interpolation*, IMA Journal of Numerical Analysis, 28 (2008), pp. 721–748.
- [15] ———, *Global convergence of general derivative-free trust-region algorithms to first- and second-order critical points*, SIAM Journal on Optimization, 20 (2009), pp. 387–415.
- [16] ———, *Introduction to Derivative-Free Optimization*, vol. 8 of MPS-SIAM Series on Optimization, MPS/SIAM, Philadelphia, 2009.
- [17] R. GARMANJANI, D. JÚDICE, AND L. N. VICENTE, *Trust-region methods without using derivatives: Worst case complexity and the nonsmooth case*, SIAM Journal on Optimization, 26 (2016), pp. 1987–2011.
- [18] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, 3rd ed., 1996.
- [19] S. GRATTON, P. L. TOINT, AND A. TRÖLTZSCH, *An active-set trust-region method for derivative-free nonlinear bound-constrained optimization*, Optimization Methods and Software, 26 (2011), pp. 873–894.
- [20] E. A. E. GUMMA, M. H. A. HASHIM, AND M. M. ALI, *A derivative-free algorithm for linearly constrained optimization problems*, Computational Optimization and Applications, 57 (2014), pp. 599–621.
- [21] D. A. HARVILLE, *Matrix Algebra From a Statistician’s Perspective*, Springer, New York, 2008.
- [22] M. HOUGH AND L. ROBERTS, *Model-based derivative-free methods for convex-constrained optimization*, SIAM Journal on Optimization, 32 (2022), pp. 2552–2579.
- [23] K. KULSHRESHTHA, B. JURGELUCKS, F. BAUSE, J. RAUTENBERG, AND C. UNVERZAGT, *Increasing the sensitivity of electrical impedance to piezoelectric material parameters with non-uniform electrical excitation*, Journal of Sensors and Sensor Systems, 4 (2015), pp. 217–227.
- [24] J. LARSON AND S. C. BILLUPS, *Stochastic derivative-free optimization using a trust region framework*, Computational Optimization and Applications, 64 (2016), pp. 619–645.
- [25] J. LARSON, M. MENICKELLY, AND S. M. WILD, *Derivative-free optimization methods*, Acta Numerica, 28 (2019), pp. 287–404.
- [26] M. J. D. POWELL, *Least Frobenius norm updating of quadratic models that satisfy interpolation conditions*, Mathematical Programming, 100 (2004), pp. 183–215.
- [27] ———, *On the use of quadratic models in unconstrained minimization without derivatives*, Optimization Methods and Software, 19 (2004), pp. 399–411.
- [28] ———, *The NEWUOA software for unconstrained optimization without derivatives*, in Large-Scale Nonlinear Optimization, P. Pardalos, G. Di Pillo, and M. Roma, eds., vol. 83, Springer US, Boston, MA, 2006, pp. 255–297.

- [29] ———, *The BOBYQA algorithm for bound constrained optimization without derivatives*, Tech. Rep. DAMTP 2009/NA06, University of Cambridge, 2009.
- [30] M. SCHEUERER AND T. M. HAMILL, *Statistical postprocessing of ensemble precipitation forecasts by fitting censored, shifted gamma distributions*, *Monthly Weather Review*, 143 (2015), pp. 4578 – 4596.
- [31] S. F. B. TETT, J. M. GREGORY, N. FREYCHET, C. CARTIS, M. J. MINETER, AND L. ROBERTS, *Does model calibration reduce uncertainty in climate projections?*, *Journal of Climate*, 35 (2022), pp. 2585–2602.
- [32] L. N. TREFETHEN, *Approximation Theory and Approximation Practice*, SIAM, Philadelphia, 2020.
- [33] G. UGHI, V. ABROL, AND J. TANNER, *An empirical study of derivative-free-optimization algorithms for targeted black-box attacks in deep neural networks*, *Optimization and Engineering*, 23 (2022), pp. 1319–1346.
- [34] S. M. WILD, *Derivative-Free Optimization Algorithms for Computationally Expensive Functions*, PhD thesis, Cornell University, 2009.
- [35] S. M. WILD, R. G. REGIS, AND C. A. SHOEMAKER, *ORBIT: Optimization by radial basis function interpolation in trust-regions*, *SIAM Journal on Scientific Computing*, 30 (2008), pp. 3197–3219.
- [36] M. Q. XUAN AND J. NOCEDAL, *A feasible method for constrained derivative-free optimization*, arXiv preprint arXiv:2402.11920, (2024).
- [37] Z. ZHANG, *Sobolev seminorm of quadratic functions with applications to derivative-free optimization*, *Mathematical Programming*, 146 (2014), pp. 77–96.