ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Lemont, IL 60439

Stochastic Derivative-Free Optimization Using a Trust Region Framework[1]

Jeffrey Larson and Stephen C. Billups

Mathematics and Computer Science Division

Preprint ANL/MCS-P5268-0115

January 2015 (*Revised January 2016*)

[1]This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research under contract number DE-AC02-06CH11357.

Stochastic Derivative-Free Optimization Using a Trust Region Framework

Jeffrey Larson Stephen C. Billups

March 7, 2016

Abstract

This paper presents a trust region algorithm to minimize a function f when one has access only to noise-corrupted function values $\bar{f}$. The model-based algorithm dynamically adjusts its step length, taking larger steps when the model and function agree and smaller steps when the model is less accurate. The method does not require the user to specify a fixed pattern of points used to build local models and does not repeatedly sample points. If f is sufficiently smooth and the noise is independent and identically distributed with mean zero and finite variance, we prove that our algorithm produces iterates such that the corresponding function gradients converge in probability to zero. We present a prototype of our algorithm that, while simplistic in its management of previously evaluated points, solves benchmark problems in fewer function evaluations than do existing stochastic approximation methods.

1 Introduction

In this article, we propose and analyze an algorithm that minimizes an unconstrained function $f : \mathbb{R}^n \to \mathbb{R}$ when no reliable information about ∇f is available and noise is present in the function evaluations. That is, the value of f at a given point x cannot be observed directly; rather, the optimization routine can access only a noise-corrupted function value $\bar{f}(x)$. Such noise may be deterministic (for example, as a result of using iterative methods) or stochastic (for example, from Monte Carlo simulations). We analyze the convergence of the algorithm in the case of stochastic noise. In particular, for our analysis we assume that the computed function values have the form

$$\bar{f}(x) = f(x) + \epsilon,$$

where the noise ϵ is drawn independently from a distribution with mean 0 and variance $\sigma^2 < \infty$. Since the mean of the noise is zero, this is equivalent to solving

$$\underset{x}{\text{minimize}}\ \mathbb{E}\left[\bar{f}(x)\right]. \tag{1}$$

Various methods for solving (1) have been presented in the literature. Stochastic approximation techniques—for example, the classical Kiefer-Wolfowitz algorithm [14] or the more recent simultaneous perturbation stochastic approximation algorithm [24]—generate iterates of the form

$$x^{k+1} = x^k + a_k G(x^k),$$

where $G(x^k)$ is a finite-difference estimate for $\nabla f(x^k)$ and a_k is the step size. These methods have been shown to converge to a stationary point under suitable conditions. This convergence is slow, however, because the step sizes are chosen a priori. An alternative to stochastic approximation methods has been developed in the experimental design community, for example in the response surface methodology literature (first presented by [5]). Response surface methods often build models by using a fixed pattern of points, for example cubic, spherical, or orthogonal designs [19]. However, determining a design that constructs response surfaces adequately approximating the function without requiring excessive function evaluations

can be difficult for problems where the user has no prior expertise. Moreover, convergence analysis of these methods is lacking.

Another possibility for solving (1) is to repeatedly sample $\bar{f}$ in order to more accurately approximate f at points of interest. This approach has been used in conjunction with stochastic approximation methods [12] and response surface methodologies [6], as well as with algorithms originally designed for deterministic derivative-free optimization. For example, Deng and Ferris [9] modify Powell's UOBYQA algorithm [22], Tomick et al. [26] modify the Nelder-Mead algorithm [21], and Deng and Ferris [10] modify the DIRECT algorithm [13] all by repeatedly sampling $\bar{f}$ to deduce information about f. None of these papers provide convergence proofs.

We find such repeated sampling to be undesirable for two reasons. First, repeated function evaluations at a point x provide information only about the noise ϵ and no information about the behavior of the function f near x. Because many points of interest will be far from the optimum of f, refining the estimate of $f(x)$ at such points wastes computational resources. Second, if the noise in $\bar{f}$ is deterministic rather than stochastic, repeated evaluations of $\bar{f}$ provide no additional information, because the same value will be returned every time. In such a case, evaluating f at nearby points is the only recourse. (Deterministic noise is less understood than its stochastic counterpart, but it has been addressed in [17, 18, 20].)

In this paper, we propose an algorithm for solving (1) that dynamically adjusts its step sizes, does not use a fixed pattern of points to build models, and does not repeatedly sample $\bar{f}$ to gain information about f. We base our algorithm on model-based trust region methods, one of the most successful classes of methods for solving problems without derivative information [23]. We believe such methods will be similarly useful when the function evaluations are corrupted by noise. Information about f is gathered by building models using function values from points in the region of interest. This approach allows the algorithm to take larger steps when the models are accurate and more conservative steps when the models are less accurate in predicting the behavior of f.

We prove that under reasonable smoothness and noise assumptions, our algorithm produces iterates $\{x^k\}$ such that $\nabla f(x^k)$ converges in probability to zero. We present numerical comparisons of three stochastic approximation methods and a prototype of the proposed algorithm on a benchmark set of optimization problems containing significant stochastic noise. The prototype of our trust region method outperforms these methods, even though it does not retain previously evaluated points from any past iteration.

Derivative-free trust region methods construct a model function at each iteration that approximates the objective function in a trust region $B(x^k; \Delta_k)$ around the current iterate x^k (where Δ_k denotes the trust region radius). This model function is minimized over the trust region to determine the next iterate. Each model function can be constructed by interpolation or regression to fit previously evaluated function values on a set of nearby sample points. In the presence of noise, these models are *random* functions. In [2], Bandeira et al. consider the convergence of trust region methods based on probabilistic models. However, their analysis relies on the assumption that exact function values are evaluated (the randomness in their models comes from choosing sample points randomly, rather than from noise in the function evaluations). In contrast, our algorithm must overcome the difficulty that exact function evaluations are not available.

The method proposed in this paper differs markedly from previous attempts in the literature to implement trust region methods in order to optimize functions with inexact evaluations. The recent algorithm QNSTOP [1] is a quasi-Newton trust region method that builds regression models of evaluated points. The algorithm requires the user to declare a sequence of decaying trust region radii to ensure convergence. (In [1], the authors suggest $\Delta_k = 2\Delta_0^{7/16}(k+1)^{-7/16}$, where Δ_0 is the initial trust region radius.) Bastin et al. [3] propose a trust region method to compute parameters for mixed logit models, which are evaluated by using a Monte Carlo approximation. Their method achieves convergence by using increasingly larger sample sizes in the Monte Carlo simulations to obtain increasingly greater accuracy in the function evaluations. The STRONG method of Chang et al. [6] combines traditionally heuristic response surface methodologies with a trust region method. The method computes estimates of the function (and possibly gradient) values by averaging multiple replications of a simulation. Convergence is ensured by an inner loop that increases the number of replicates to achieve greater accuracy. Both Bastin et al.'s method and STRONG rely on repeated sampling, which is explicitly avoided in our algorithm. We also note similarities between our approach and

the recently proposed method of Chen et al. [7], which uses a similar trust region framework; however, their assumptions and analysis differ from our approach.

The structure of the paper is as follows. In Section 2, we provide some preliminary results and definitions. We then outline the algorithm in Section 3. In Section 4, we prove convergence of our algorithm provided the function, noise, and algorithmic constants satisfy certain assumptions. In Section 5, we discuss numerical experiments. A summary and appendix of results conclude the paper.

We use the following notation in this paper. Let $\mathbb{R}$ denote the set of all reals, and let $\mathcal{P}_n^d$ be the set of polynomials in $\mathbb{R}^n$ with degree at most d. The ball of radius Δ centered at $x \in \mathbb{R}^n$ is denoted $B(x;\Delta)$. Let $\|\cdot\|$ denote the Euclidean norm. C^k denotes the set of functions on $\mathbb{R}^n$ with k continuous derivatives, LC^k denotes the set of functions in C^k such that the kth derivative is Lipschitz continuous, and e_j denotes the jth column of the identity matrix. For $F : \mathbb{R}^n \to \mathbb{R}^m$, $m > 1$, $\nabla F(x)$ denotes the $m \times n$ Jacobian matrix; however, for $f : \mathbb{R}^n \to \mathbb{R}$, the gradient $\nabla f(x)$ is interpreted as a column vector.

2 Background

Traditional trust region methods use derivative information to construct a local approximation m_k of the objective function at every iterate x^k. When derivatives are unavailable, the model function can be constructed by interpolation or regression using a set of points $Y = \{y^0, \ldots, y^p\}$ where the function has previously been evaluated. The model functions have the form $m(x) = \sum_{i=0}^{q} \mu_i \phi_i(x)$, where $\phi = \{\phi_0, \ldots, \phi_q\}$ is a basis for the space of possible model functions. To construct the model function, we define the *design matrix* by

$$M = M(\phi, Y) = \begin{bmatrix} \phi_0(y^0) & \phi_1(y^0) & \cdots & \phi_q(y^0) \\ \phi_0(y^1) & \phi_1(y^1) & \cdots & \phi_q(y^1) \\ \vdots & & \ddots & \vdots \\ \phi_0(y^p) & \phi_1(y^p) & \cdots & \phi_q(y^p) \end{bmatrix}. \tag{2}$$

If M has full column rank, the sample set is said to be *poised*, in which case the coefficients $\mu = [\mu_0, \ldots, \mu_q]^T$ of the model function can be determined by solving the system

$$M(\phi, Y)\mu = \bar{f}, \tag{3}$$

where $\bar{f} = \left[\bar{f}(y^0), \ldots, \bar{f}(y^p)\right]^T$ denotes the computed function values at the sample points. For interpolation models, the number of sample points $p+1 = |Y|$ is equal to the dimension $q+1$ of the model function space. In this case, M is a square matrix; and if Y is poised, $\mu = M^{-1}\bar{f}$. For regression models, $p+1 > q+1$, (3) is solved in the least-squares sense, with solution $\mu = M^+\bar{f}$, where $M^+ = (M^T M)^{-1} M^T$ is the pseudoinverse of M.

2.1 Regression Lagrange polynomials

An important tool for analyzing of regression models is *regression Lagrange polynomials* [8].

Definition 1. *Given a sample set $Y = \{y^0, \ldots, y^p\}$ with $p > q$, a set of polynomials $\ell_j(x), j = 0, \ldots, p$ in $\mathcal{P}_n^d$ is called a set of* regression Lagrange polynomials *with respect to the sample set Y if for each j, ℓ_j is a least-squares solution to the equations*

$$\ell_j(y^i) = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}, \quad i = 0, \ldots, p.$$

The following properties of regression Lagrange polynomials are developed in [8].

Lemma 1. *[8, Lemma 4.5 & page 62] If Y is poised, then the set of regression Lagrange polynomials exists and is uniquely defined. In fact,*

$$\ell(x) = (M^T)^+\phi(x),$$

where $\ell(x) = (\ell_0(x), \ldots, \ell_p(x))^T$ and $(M^T)^+ = M(M^TM)^{-1}$ is the pseudoinverse of M^T.

Lemma 2. *[8, Lemma 4.6] For any function $f : \mathbb{R}^n \to \mathbb{R}$, if Y is poised for regression, the unique polynomial $m(x)$ that approximates $f(x)$ by least-squares regression at the points in Y can be expressed as*

$$m(x) = \sum_{i=0}^{p} f(y^i)\ell_i(x).$$

This lemma clearly shows that the geometry of the sample set plays a crucial role in the quality of the model function produced by least-squares regression. Specifically, with respect to sensitivity to errors, one should choose the sample points in such a way that the Lagrange polynomials have small absolute values. This leads to the following definition from [8]:

Definition 2. *Let $\Lambda > 0$ and let a set $B \subset \mathbb{R}^n$ be given. Let $\phi = \{\phi_0, \ldots, \phi_q\}$ be a basis for $\mathcal{P}_n^d$. A poised set $Y = \{y^0, \ldots, y^p\}$ with $p+1 > q+1$ points is said to be* strongly Λ-poised *(in the regression sense) on B if and only if*

$$\frac{q+1}{\sqrt{p+1}}\Lambda \geq \max_{x\in B} \|\ell(x)\|,$$

where $\ell(x) = [\ell_0(x), \ldots, \ell_p(x)]^T$.

We focus on regression models because stochastic noise is present in our function evaluations. For simplicity, only first-order polynomials ($q = n$) will be considered in this paper. We recall the following definition from [8] that characterizes model functions that behave similarly to Taylor approximations of f in a given neighborhood.

Definition 3. *Let $f \in LC^1$ be given. Let $\kappa = (\kappa_{\text{ef}}, \kappa_{\text{eg}}, \nu_1^{\text{m}})$ be a given vector of constants, and let $\Delta > 0$ be given. A function $m \in LC^1$ with Lipschitz constant ν_1^{m} is a κ-*fully linear *model of f on $B(x;\Delta)$ if, for all $y \in B(x;\Delta)$,*

$$\begin{aligned} \|\nabla f(y) - \nabla m(y)\| &\leq \kappa_{\text{eg}}\Delta, \text{ and} \\ |f(y) - m(y)| &\leq \kappa_{\text{ef}}\Delta^2. \end{aligned}$$

2.2 Random models

When noise is present, the model functions $\{m_k\}$ constructed from interpolation or regression are necessarily *random* models. Consequently, the iterates $\{x^k\}$, the trust region radii $\{\Delta_k\}$, and the solutions $\{s^k\}$ to the trust region are also random quantities. For convenience, we do not distinguish notationally between the random quantities and their realizations; however, this distinction should be clear from context.

The concept of κ-fully linear models was generalized to random models in [15] and [2]; the following definition is adapted from [2]. Let $\mathcal{F}_k$ denote the realizations of all the random events for the first k iterations of the algorithm.

Definition 4. *Let $\kappa = (\kappa_{\text{ef}}, \kappa_{\text{eg}}, \nu_1^{\text{m}})$ be a given vector of constants, and let $\alpha \in (0,1)$. Let $\Delta > 0$ be given. A random model m_k generated at the kth iteration of an algorithm is α-probabilistically κ-fully linear on $B(x;\Delta)$ if*

$$P\left(m_k \text{ is a } \kappa\text{-fully linear model of } f \text{ on } B(x;\Delta) \middle| \mathcal{F}_{k-1}\right) \geq \alpha.$$

Many types of general models can be used; the models do not have to be linear. All that is required is that the models approximate the function in a manner similar to Taylor models with a certain probability and that they generate trust region subproblems that can be solved accurately. Proposition 3 in the appendix shows that linear models regressing enough points that are strongly Λ-poised are α-probabilistically κ-fully linear.

3 Algorithm

The algorithm presented here requires several elements. First, at each iteration, a model function m_k is constructed that approximates f on the trust region $B(x^k;\Delta_k)$. To prove convergence, we require that for some $\kappa = (\kappa_{\rm ef}, \kappa_{\rm eg}, \nu_1^{\rm m})$ and for all k sufficiently large, $\{m_k\}$ is α-probabilistically κ-fully linear on the corresponding sequence $\{B(x^k;\Delta_k)\}$ of trust regions where α is a certain threshold satisfying

$$\alpha \geq \max\left\{\frac{1}{2}, 1 - \frac{\frac{\gamma_{\rm inc}-1}{\gamma_{\rm inc}}}{4\left[\frac{\gamma_{\rm inc}-1}{2\gamma_{\rm inc}} + \frac{1-\gamma_{\rm dec}}{\gamma_{\rm dec}}\right]}, 1 - \frac{1-\gamma_{\rm dec}}{2(\gamma_{\rm inc}^2 - \gamma_{\rm dec})}\right\}, \tag{4}$$

and where $\gamma_{\rm inc}$ and $\gamma_{\rm dec}$ are parameters in the algorithm. As a reference, the choices $\gamma_{\rm inc} = 2$ and $\gamma_{\rm dec} = 0.5$ imply that $\alpha \geq 0.9$, whereas $\gamma_{\rm inc} = 2$ and $\gamma_{\rm dec} = 0.9$ imply that $\alpha \geq 0.845$.

In our implementation of our algorithm presented in Section 5, m_k is a linear function determined by least-squares regression on a strongly Λ-poised sample set $Y_k \subset B(x^k;\Delta_k)$ consisting of at least c_k/Δ_k^4 points, where $\{c_k\}$ is a positive sequence diverging to $+\infty$. Proposition 3 in the appendix demonstrates that this method of constructing m_k satisfies the above requirement.

After the model function has been constructed, a trial step s^k is determined by solving the trust region subproblem

$$\min_{\|s\|\leq\Delta_k} m_k(x^k + s).$$

To determine whether to accept the trial step, we require estimates F_k^0 and F_k^s of the function values $f(x^k)$ and $f(x^k + s^k)$, respectively. To prove convergence, we assume that the estimates satisfy the following conditions, where β and η are parameters in Algorithm 1.

Condition 1: There exists $\bar{k}$ such that for all $k > \bar{k}$,

$$\mathbb{P}\left[\left|F_k^0 - f(x^k) + f(x^k + s^k) - F_k^s\right| > \beta\eta \min\left\{\Delta_k, \Delta_k^2\right\} \middle| \mathcal{F}_{k-1}\right] \leq 1 - \alpha. \tag{5}$$

Condition 2: There are constants $\theta > 0$ and $\hat{k}$ such that for all $k > \hat{k}$ and $\xi > 0$,

$$\mathbb{P}\left[F_k^0 - f(x^k) + f(x^k + s^k) - F_k^s > (\beta\eta + \xi) \min\left\{\Delta_k, \Delta_k^2\right\} \middle| \mathcal{F}_{k-1}\right] \leq \frac{\theta}{\xi}. \tag{6}$$

In our implementation, we compute F_k^0 and F_k^s by constructing linear models by least-squares regression as follows. Let $\{c_k\}$ and $\{a_k\}$ be positive sequences such that $c_k \to +\infty$ and $a_k \downarrow 0$, and define $\delta_k = \min\{\Delta_k, 1\}$. Choose strongly Λ-poised sample sets $Y_k^0 \subset B(x^k; a_k\delta_k)$ and $Y_k^s \subset B(x^k + s^k; a_k\delta_k)$ with at least $c_k/(a_k^5\delta_k^4)$ points in each set. Let m_k^0 and m_k^s be the linear polynomials determined by least-squares regression on the sample sets Y_k^0 and Y_k^s, respectively. Define $F_k^0 = m_k^0(x^k)$ and $F_k^s = m_k^s(x^k + s^k)$.

Proposition 2 in the appendix demonstrates that Conditions 1 and 2 are satisfied by this method of constructing the function estimates F_k^0 and F_k^s.

The algorithm measures the accuracy of the model m_k by comparing the ratio of decrease $F_k^0 - F_k^s$ and the model's predicted decrease $m_k(x^k) - m_k(x^k + s^k)$. When this ratio is sufficiently positive, m_k is considered to sufficiently predict the behavior of f near x^k, and the trust region radius for the next iteration is increased. On the other hand, if the ratio is not sufficiently positive, the next iteration will use a smaller trust region radius.

We define an iteration where $x^{k+1} = x^k + s^k$ to be *successful* and any iteration where $x^{k+1} = x^k$ to be *unsuccessful*. For ease of analysis, we assume that our models are α-probabilistically κ-fully linear every iteration, although a practical algorithm would likely force the models to be good only when the algorithm stops progressing.

Algorithm 1: Trust region algorithm for minimizing a stochastic function.

Pick $0 < \gamma_{\text{dec}} < 1 < \gamma_{\text{inc}}$, $0 < \eta,\ \beta < 1$, $0 < \Delta_0$, and $\alpha \in (0,1)$, satisfying (4). Set $k = 0$;
Start
Build an α_k-probabilistically κ-fully linear model m_k on $B(x^k; \Delta_k)$ for some $\alpha_k \in (0,1)$ such that $\alpha_k \geq \alpha$ for sufficiently large k;
Compute $s^k = \arg\min_{s:\|s\| \leq \Delta_k} m_k(x^k + s)$;
if $m_k(x^k) - m_k(x^k + s^k) \geq \beta \min\left\{\Delta_k, \Delta_k^2\right\}$ **then**
  Calculate $\rho_k = \dfrac{F_k^0 - F_k^s}{m_k(x^k) - m_k(x^k + s^k)}$;
  if $\rho_k \geq \eta$ **then**
    $x^{k+1} = x^k + s^k$; $\Delta_{k+1} = \gamma_{\text{inc}} \Delta_k$;
  else
    $x^{k+1} = x^k$; $\Delta_{k+1} = \gamma_{\text{dec}} \Delta_k$;
  end
else
  $x^{k+1} = x^k$; $\Delta_{k+1} = \gamma_{\text{dec}} \Delta_k$;
end
$k = k + 1$ and go to **Start**;

4 Convergence Analysis

We first establish a lower bound on the decrease of the model function at each iteration.

Lemma 3. *If a model m_k has a Lipschitz continuous gradient with Lipschitz constant ν_1^{m}, then*

$$m_k(x^k) - m_k(x^k + s^k) \geq \frac{\left\|\nabla m_k(x^k)\right\|}{2} \min\left\{\frac{\left\|\nabla m_k(x^k)\right\|}{2\nu_1^{\text{m}}}, \Delta_k\right\}. \tag{7}$$

Proof. Define $g^k = \nabla m_k(x^k)$ and $\theta(t) = m_k(x^k - tg^k)$. By the mean value theorem,

$$m_k(x^k) - \theta(t) = t\nabla m_k(\xi(t))^T g^k \geq t\left\|g^k\right\|^2 - t\left\|g^k - \nabla m_k(\xi(t))\right\|\left\|g^k\right\|,$$

where $\xi(t)$ is a point on the line segment connecting x^k to $x^k - tg^k$.

By assumption, $\left\|g^k - \nabla m_k(\xi(t))\right\| \leq \nu_1^{\text{m}} t \left\|g^k\right\|$. Thus,

$$m_k(x^k) - \theta(t) \geq t\left\|g^k\right\|^2 - t^2 \nu_1^{\text{m}} \left\|g^k\right\|^2.$$

Note that the right-hand side is a concave quadratic function of t, which is maximized by $t^* = \dfrac{1}{2\nu_1^{\text{m}}}$, yielding a value of $\dfrac{\left\|g^k\right\|^2}{4\nu_1^{\text{m}}}$.

If $\left\|g^k\right\| \leq 2\nu_1^{\text{m}} \Delta_k$, then $t^* \left\|g^k\right\| \leq \Delta_k$. Recalling that $s^k = \arg\min_{s:\|s\| \leq \Delta_k} m_k(x^k + s)$ yields the inequality

$$m_k(x^k) - m_k(x^k + s^k) \geq m_k(x^k) - \theta(t^*) \geq \frac{\left\|g^k\right\|^2}{4\nu_1^{\text{m}}}.$$

If $\left\|g^k\right\| > 2\nu_1^{\text{m}} \Delta_k$, then

$$m_k(x^k) - m_k(x^k + s^k) \geq m_k(x^k) - \theta\left(\frac{\Delta_k}{\|g^k\|}\right) \geq \Delta_k \left\|g^k\right\| - \nu_1^{\text{m}} \Delta_k^2 \geq \Delta_k \frac{\left\|g^k\right\|}{2}.$$

In either case, (7) is satisfied. □

For the remainder of this section, we make the following assumptions.

Assumption 1. *The additive noise ϵ observed when computing $\bar{f}$ is drawn from a distribution with mean zero and finite variance.*

Assumption 2. *f has bounded level sets and ∇f is Lipschitz continuous with constant L_g.*

We now prove that if Assumptions 1–2 are satisfied, then the algorithm generates iterates $\{x^k\}$ such that $\nabla f(x^k)$ converges in probability to zero. For the sake of clarity, we first outline the logic used to prove convergence.

- Lemma 4 shows that the sequence of trust region radii $\{\Delta_k\}$ from Algorithm 1 goes to zero almost surely.
- Lemma 5 shows that for k sufficiently large, if Δ_k falls below some constant multiple of the model gradient, then the step will be successful with high probability.
- Theorem 1 proves that the sequence of gradients $\left\{\nabla f(x^k)\right\}$ converges in probability to zero.

Lemma 4. *If f has bounded level sets and the estimates F_k^0 and F_k^s are constructed to satisfy Conditions 1 and 2 for α satisfying* (4)*, then*

$$\sum_{k=0}^{\infty} \Delta_k^2 < \infty$$

almost surely (and therefore $\Delta_k \to 0$ almost surely as well).

Proof. Let θ and $\hat{k}$ be as defined in Condition 2. Let $\nu = \dfrac{1-\gamma_{\text{dec}}}{4\theta + 2(1-\gamma_{\text{dec}})}$, and define $\varphi(x,\Delta) = \nu f(x) + (1-\nu)\Delta \min\{1,\Delta\}$. For $k \in \mathbb{N}$, define $\delta_k = \min\{1,\Delta_k\}$ and $\xi_k = \dfrac{f(x^k+s^k) - f(x^k)}{\Delta_k \delta_k}$.

For a successful step, $x^{k+1} = x^k + s^k$ and $\Delta_{k+1} = \gamma_{\text{inc}}\Delta_k$. Thus,

$$\begin{aligned}
\varphi(x^{k+1},\Delta_{k+1}) - \varphi(x^k,\Delta_k) &= \nu\left(f(x^k+s^k) - f(x^k)\right) + (1-\nu)\left(\Delta_{k+1}\delta_{k+1} - \Delta_k\delta_k\right) \\
&\le \nu\xi_k\Delta_k\delta_k + (1-\nu)\left(\gamma_{\text{inc}}\Delta_k \min\{1,\gamma_{\text{inc}}\Delta_k\} - \Delta_k\delta_k\right) \\
&\le \nu\xi_k\Delta_k\delta_k + (1-\nu)(\gamma_{\text{inc}}^2 - 1)\Delta_k\delta_k \\
&= \left(\nu\xi_k + (1-\nu)(\gamma_{\text{inc}}^2-1)\right)\Delta_k\delta_k. \qquad (8)
\end{aligned}$$

For an unsuccessful step, $x^{k+1} = x^k$ and $\Delta_{k+1} = \gamma_{\text{dec}}\Delta_k$. Thus,

$$\begin{aligned}
\varphi(x^{k+1},\Delta_{k+1}) - \varphi(x^k,\Delta_k) &= (1-\nu)\left(\gamma_{\text{dec}}\Delta_k \min\{1,\gamma_{\text{dec}}\Delta_k\} - \Delta_k\min\{1,\Delta_k\}\right) \\
&\le (1-\nu)(\gamma_{\text{dec}}-1)\Delta_k\delta_k. \qquad (9)
\end{aligned}$$

Let π_k denote the probability that the kth step is successful given $\mathcal{F}_{k-1}$. Then,

$$\begin{aligned}
\pi_k &= \mathbb{P}\left[m_k(x^k) - m_k(x^k+s^k) \ge \beta\Delta_k\delta_k \middle| \mathcal{F}_{k-1}\right] \mathbb{P}\left[\rho_k \ge \eta \middle| \mathcal{F}_{k-1}, m_k(x^k) - m_k(x^k+s^k) \ge \beta\Delta_k\delta_k\right] \\
&\le \mathbb{P}\left[\rho_k \ge \eta \middle| \mathcal{F}_{k-1}, m_k(x^k) - m_k(x^k+s^k) \ge \beta\Delta_k\delta_k\right] \\
&= \mathbb{P}\left[F_k^0 - F_k^s \ge \eta\left(m_k(x^k) - m_k(x^k+s^k)\right) \middle| \mathcal{F}_{k-1}, m_k(x^k) - m_k(x^k+s^k) \ge \beta\Delta_k\delta_k\right] \\
&\le \mathbb{P}\left[F_k^0 - F_k^s \ge \eta\beta\Delta_k\delta_k \middle| \mathcal{F}_{k-1}\right] \\
&= \mathbb{P}\left[F_k^0 - f(x^k) + f(x^k) - f(x^k+s^k) + f(x^k+s^k) - F_k^s \ge \eta\beta\Delta_k\delta_k \middle| \mathcal{F}_{k-1}\right] \\
&= \mathbb{P}\left[F_k^0 - f(x^k) + f(x^k+s^k) - F_k^s \ge (\eta\beta + \xi_k)\Delta_k\delta_k \middle| \mathcal{F}_{k-1}\right]. \qquad (10)
\end{aligned}$$

Suppose $\xi_k > 0$. By (5) with α satisfying (4), there is a constant $\bar{k}$ such that for all $k > \bar{k}$,

$$\mathbb{P}\left[\left|F_k^0 - f(x^k) + f(x^k + s^k) - F_k^s\right| > \beta\eta \min\left\{\Delta_k, \Delta_k^2\right\} \middle| \mathcal{F}_{k-1}\right] \leq 1 - \alpha \leq \frac{1 - \gamma_{\text{dec}}}{2(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}})}.$$

Combining this with (6) and (10), we conclude that for all $k > \max\{\bar{k}, \hat{k}\}$,

$$\pi_k \leq \min\left\{\frac{\theta}{\xi_k}, \frac{1 - \gamma_{\text{dec}}}{2(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}})}\right\}. \tag{11}$$

Combining (8), (9), and (11) yields the following inequality for $k > \max\{\bar{k}, \hat{k}\}$:

$$\begin{aligned}
\frac{E\left(\varphi(x^{k+1}, \Delta_{k+1}) - \varphi(x^k, \Delta_k)\middle|\mathcal{F}_{k-1}\right)}{\Delta_k \delta_k} &\leq \pi_k\left(\nu\xi_k + (1-\nu)\left(\gamma_{\text{inc}}^2 - 1\right)\right) + (1-\pi_k)(1-\nu)\left(\gamma_{\text{dec}} - 1\right) \\
&= \pi_k \nu \xi_k + (1-\nu)\left(\gamma_{\text{dec}} - 1\right) + \pi_k(1-\nu)\left(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}}\right) \\
&\leq \nu\theta + (1-\nu)\left(\gamma_{\text{dec}} - 1\right) + \frac{1 - \gamma_{\text{dec}}}{2\left(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}}\right)}(1-\nu)\left(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}}\right) \\
&= \nu\theta + (1-\nu)\frac{(\gamma_{\text{dec}} - 1)}{2} \\
&= \frac{\gamma_{\text{dec}} - 1}{2} + \nu\left(\theta + \frac{1 - \gamma_{\text{dec}}}{2}\right) \\
&= \frac{\gamma_{\text{dec}} - 1}{2} + \left(\frac{1 - \gamma_{\text{dec}}}{4\theta + 2(1 - \gamma_{\text{dec}})}\right)\left(\frac{2\theta + (1 - \gamma_{\text{dec}})}{2}\right) \\
&= \frac{\gamma_{\text{dec}} - 1}{4}.
\end{aligned}$$

In the case when $\xi_k \leq 0$, combining (8) and (9) and noting that $\nu < 1/2$, we have the inequality

$$\begin{aligned}
\frac{E\left(\varphi(x^{k+1}, \Delta_{k+1}) - \varphi(x^k, \Delta_k)\middle|\mathcal{F}_{k-1}\right)}{\Delta_k \delta_k} &\leq \pi_k\left(\nu\xi_k + (1-\nu)\left(\gamma_{\text{inc}}^2 - 1\right)\right) + (1-\pi_k)(1-\nu)\left(\gamma_{\text{dec}} - 1\right) \\
&\leq (1-\nu)\left(\gamma_{\text{dec}} - 1\right) + \pi_k(1-\nu)\left(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}}\right) \\
&\leq (1-\nu)\left(\gamma_{\text{dec}} - 1\right) + \frac{1 - \gamma_{\text{dec}}}{2\left(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}}\right)}(1-\nu)\left(\gamma_{\text{inc}}^2 - \gamma_{\text{dec}}\right) \\
&= (1-\nu)\frac{(\gamma_{\text{dec}} - 1)}{2} \\
&< \frac{\gamma_{\text{dec}} - 1}{4}.
\end{aligned}$$

Combining the two cases above, for $k > \max\{\bar{k}, \hat{k}\}$ we have

$$E\left(\varphi(x^{k+1}, \Delta_{k+1}) - \varphi(x^k, \Delta_k)\middle|\mathcal{F}_{k-1}\right) \leq \frac{\gamma_{\text{dec}} - 1}{4}\Delta_k \delta_k.$$

Since f is bounded below, it follows that $\sum_{k=0}^{\infty} \Delta_k \delta_k < \infty$ almost surely. This implies that $\Delta_k < 1$ for all but finitely many k, so $\sum_{k=0}^{\infty} \Delta_k^2 < \infty$ almost surely. □

Lemma 5. *Let $\kappa = (\kappa_{\text{ef}}, \kappa_{\text{eg}}, \nu_1^{\text{m}})$, and let $\alpha \in (0, 1)$ be given. Let η and β be the constants specified in the algorithm. Then for k sufficiently large, if*

$$\Delta_k \leq \min\left\{\frac{\left\|\nabla f(x^k)\right\|(1-\eta)}{(1-\eta)\kappa_{\text{eg}} + 2(2\kappa_{\text{ef}} + \beta\eta)}, \frac{\left\|\nabla f(x^k)\right\|}{2\nu_1^{\text{m}} + \kappa_{\text{eg}}}\right\}, \tag{12}$$

then $\rho_k \geq \eta$ with probability at least $2\alpha - 1$.

Proof. For any k, since the model is α-probabilistically κ-fully linear, then with probability at least α, both the following inequalities hold for all $x \in B(x^k; \Delta_k)$.

$$\|\nabla f(x)\| - \kappa_{\text{eg}}\Delta_k \leq \|\nabla m_k(x)\| \tag{13}$$

$$|f(x) - m_k(x)| \leq \kappa_{\text{ef}}\Delta_k^2. \tag{14}$$

In this case

$$\Delta_k \leq \frac{\left(\left\|\nabla f(x^k)\right\| - \kappa_{\text{eg}}\Delta_k\right)(1-\eta)}{2(2\kappa_{\text{ef}} + \beta\eta)} \leq \frac{\left\|\nabla m_k(x^k)\right\|(1-\eta)}{2(2\kappa_{\text{ef}} + \beta\eta)}, \tag{15}$$

where the first inequality is implied by (12) and the second is a direct application of (13). Similarly, from (12) and (13),

$$\Delta_k \leq \frac{\left\|\nabla m_k(x^k)\right\|}{2\nu_1^{\text{m}}}.$$

Thus, by Lemma 3,

$$m_k(x^k) - m_k(x^k + s^k) \geq \frac{\left\|\nabla m_k(x^k)\right\|}{2}\Delta_k. \tag{16}$$

By the definition of F_k^0 and F_k^s, there exists $\bar{k}$ such that for all $k > \bar{k}$

$$\left|F_k^0 - f(x^k) + f(x^k + s^k) - F_k^s\right| \leq \beta\eta\Delta_k^2, \tag{17}$$

holds with probability at least α. Therefore, the set of outcomes satisfying (13), (14), (16), and (17) occurs with probability at least $2\alpha - 1$. In this case

$$\begin{aligned}
\left(m_k(x^k) - m_k(x^k + s^k)\right)\rho_k &= F_k^0 - F_k^s \\
&= \left(F_k^0 - f(x^k)\right) + \left(f(x^k) - m_k(x^k)\right) + \left(m_k(x^k) - m_k(x^k + s^k)\right) \\
&\quad + \left(m_k(x^k + s^k) - f(x^k + s^k)\right) + \left(f(x^k + s^k) - F_k^s\right) \\
&\geq m_k(x^k) - m_k(x^k + s^k) - 2\kappa_{\text{ef}}\Delta_k^2 - \beta\eta\Delta_k^2.
\end{aligned}$$

Thus, for $k > \bar{k}$,

$$\rho_k \geq 1 - \frac{(2\kappa_{\text{ef}} + \beta\eta)\Delta_k^2}{m_k(x^k) - m_k(x^k + s^k)} \geq 1 - \frac{2(2\kappa_{\text{ef}} + \beta\eta)\Delta_k^2}{\|\nabla m(x^k)\|\,\Delta_k} \geq \eta$$

with probability at least $2\alpha - 1$, where the last inequality comes from (15). □

Theorem 1. *If Assumptions 1–2 are satisfied and α is chosen to satisfy* (4)*, then $\left\{\left\|\nabla f(x^k)\right\|\right\}$ converges in probability to zero. That is, for all $\epsilon > 0$,*

$$\lim_{k\to\infty} \mathbb{P}\left[\left\|\nabla f(x^k)\right\| > \epsilon\right] = 0$$

Proof. Define the following values:

$$\psi_k = \frac{\left\|\nabla f(x^k)\right\|}{\Delta_k},$$

$$\mathfrak{L}_1 = \max\left\{2\nu_1^{\text{m}} + \kappa_{\text{eg}}, \frac{(1-\eta)\kappa_{\text{eg}} + 2(2\kappa_{\text{ef}} + \beta\eta)}{(1-\eta)}, \frac{2L_g}{\gamma_{inc} - 1}, \frac{L_g}{\gamma_{\text{inc}}}\left(\frac{1-\gamma_{\text{dec}}}{\gamma_{\text{dec}}} - \frac{1-\gamma_{\text{inc}}}{\gamma_{\text{inc}}}\right)^{-1}\right\},$$

$$\mathfrak{L}_2 = \max\left\{\frac{\mathfrak{L}_1}{\gamma_{\text{dec}}}, L_g + \mathfrak{L}_1\right\}.$$

Before proceeding, we note that, by (4),

$$2(1-\alpha)\left(\frac{1-\gamma_{\text{dec}}}{\gamma_{\text{dec}}}\right)+(2\alpha-1)\left(\frac{1-\gamma_{\text{inc}}}{2\gamma_{\text{inc}}}\right)\leq 0. \tag{18}$$

Consider the case where $\psi_k \geq \mathfrak{L}_1$. Then,

$$\Delta_k=\frac{\|\nabla f(x^k)\|}{\psi_k}\leq\frac{\|\nabla f(x^k)\|}{\mathfrak{L}_1}\leq\min\left\{\frac{\|\nabla f(x^k)\|}{2\nu_1^{\text{m}}+\kappa_{\text{eg}}},\frac{\|\nabla f(x^k)\|(1-\eta)}{(1-\eta)\kappa_{\text{eg}}+2(2\kappa_{\text{ef}}+\beta\eta)}\right\}.$$

Therefore by Lemma 5, for k sufficiently large, we have successful iterates with probability at least $2\alpha-1$. On those iterates

$$\begin{aligned}\psi_{k+1}-\psi_k&=\frac{\|\nabla f(x^{k+1})\|}{\gamma_{\text{inc}}\Delta_k}-\psi_k\\&\leq\frac{\|\nabla f(x^k)\|+L_g\Delta_k}{\gamma_{\text{inc}}\Delta_k}-\frac{\|\nabla f(x^k)\|}{\Delta_k}\\&=\frac{\|\nabla f(x^k)\|}{\Delta_k}\frac{1-\gamma_{\text{inc}}}{\gamma_{\text{inc}}}+\frac{L_g}{\gamma_{\text{inc}}}\\&=\psi_k\frac{1-\gamma_{\text{inc}}}{\gamma_{\text{inc}}}+\frac{L_g}{\gamma_{\text{inc}}}\leq 0,\end{aligned}$$

where the last inequality is implied by $\psi_k \geq \mathfrak{L}_1 \geq 2L_g/(\gamma_{\text{inc}}-1)$.

On unsuccessful iterates, which occur with probability at most $2(1-\alpha)$, $\psi_{k+1}=\dfrac{\psi_k}{\gamma_{\text{dec}}}$ or $\psi_{k+1}-\psi_k=\left(\dfrac{1}{\gamma_{\text{dec}}}-1\right)\psi_k$. Note that since

$$\psi_k\geq\mathfrak{L}_1\geq\frac{L_g}{\gamma_{\text{inc}}}\left(\frac{1-\gamma_{\text{dec}}}{\gamma_{\text{dec}}}-\frac{1-\gamma_{\text{inc}}}{\gamma_{\text{inc}}}\right)^{-1},$$

this implies

$$\psi_k\frac{1-\gamma_{\text{inc}}}{\gamma_{\text{inc}}}+\frac{L_g}{\gamma_{\text{inc}}}\leq\left(\frac{1}{\gamma_{\text{dec}}}-1\right)\psi_k.$$

Considering both successful iterates and unsuccessful steps together, we have

$$\begin{aligned}\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_k\right]-\psi_k&\leq(2\alpha-1)\left[\psi_k\frac{1-\gamma_{\text{inc}}}{\gamma_{\text{inc}}}+\frac{L_g}{\gamma_{\text{inc}}}\right]+2(1-\alpha)\left(\frac{1}{\gamma_{\text{dec}}}-1\right)\psi_k\\&=(2\alpha-1)\left[\psi_k\frac{1-\gamma_{\text{inc}}}{2\gamma_{\text{inc}}}+\frac{L_g}{\gamma_{\text{inc}}}\right]+\psi_k\left[2(1-\alpha)\left(\frac{1}{\gamma_{\text{dec}}}-1\right)+(2\alpha-1)\frac{1-\gamma_{\text{inc}}}{2\gamma_{\text{inc}}}\right]\\&\leq(2\alpha-1)\left[\psi_k\frac{1-\gamma_{\text{inc}}}{2\gamma_{\text{inc}}}+\frac{L_g}{\gamma_{\text{inc}}}\right],\end{aligned} \tag{19}$$

where the last inequality holds by (18).

Define $a=\dfrac{(\gamma_{\text{inc}}-1)(2\alpha-1)}{2\gamma_{\text{inc}}}$ and $b=\max\left\{L_2,\dfrac{(2\alpha-1)L_g}{\gamma_{\text{inc}}}\right\}$ and note that $0<a<1$. If $\psi_k\geq\mathfrak{L}_1$, rearranging (19) yields

$$\begin{aligned}\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_k\right]&\leq\left[\frac{(2\alpha-1)(1-\gamma_{\text{inc}})}{2\gamma_{\text{inc}}}+1\right]\psi_{k+1}+\frac{(2\alpha-1)L_g}{\gamma_{\text{inc}}}\\&\leq(1-a)\psi_k+b.\end{aligned}$$

When $\psi_k \leq \mathfrak{L}_1$, on unsuccessful iterates

$$\psi_{k+1} = \frac{\psi_k}{\gamma_{\text{dec}}} \leq \frac{\mathfrak{L}_1}{\gamma_{\text{dec}}} \leq \mathfrak{L}_2,$$

and on successful iterates

$$\psi_{k+1} = \frac{\left\|\nabla f(x^{k+1})\right\|}{\gamma_{\text{inc}}\Delta_k} \leq \frac{L_g\Delta_k + \left\|\nabla f(x^k)\right\|}{\gamma_{\text{inc}}\Delta_k} \leq \frac{L_g\Delta_k + \left\|\nabla f(x^k)\right\|}{\Delta_k} = L_g + \psi_k \leq L_g + \mathfrak{L}_1 \leq \mathfrak{L}_2.$$

If $\psi_k \leq \mathfrak{L}_1$,

$$\psi_{k+1} \leq \mathfrak{L}_2 \leq b \leq (1-a)\psi_k + b.$$

Therefore, independent of the value of ψ_k, $\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_k\right] \leq (1-a)\psi_k + b$, and

$$\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_{k-1}\right] = \mathbb{E}\left[\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_k\right]|\mathcal{F}_{k-1}\right] \leq (1-a)\,\mathbb{E}\left[\psi_k|\mathcal{F}_{k-1}\right] + b. \tag{20}$$

Continuing in this fashion,

$$\begin{aligned}
\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_{k-2}\right] &= \mathbb{E}\left[\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_{k-1}\right]|\mathcal{F}_{k-2}\right] \\
&\leq (1-a)\,\mathbb{E}\left[\mathbb{E}\left[\psi_k|\mathcal{F}_{k-1}\right]|\mathcal{F}_{k-2}\right] + b \qquad \text{by (20)} \\
&\leq (1-a)\left[(1-a)\,\mathbb{E}\left[\psi_{k-1}|\mathcal{F}_{k-2}\right] + b\right] + b \\
&= (1-a)^2\,\mathbb{E}\left[\psi_{k-1}|\mathcal{F}_{k-2}\right] + \left[(1-a)+1\right]b.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\mathbb{E}\left[\psi_{k+1}|\mathcal{F}_0\right] &\leq (1-a)^k\psi_0 + b\sum_{j=0}^{k-1}(1-a)^j \\
&\leq (1-a)^k\psi_0 + \frac{b}{a} \\
&\leq \psi_0 + \frac{b}{a}.
\end{aligned}$$

Defining $c = \psi_0 + \dfrac{b}{a}$, we have shown that $\mathbb{E}\left[\psi_k|\mathcal{F}_0\right] \leq c$ for all k. For any $\epsilon > 0$,

$$\begin{aligned}
\mathbb{P}\left[\left\|\nabla f(x^k)\right\| > \epsilon|\mathcal{F}_0\right] &= \mathbb{P}\left[\psi_k > \frac{\epsilon}{\Delta_k}|\mathcal{F}_0\right] \\
&= \mathbb{P}\left[\psi_k > \frac{\epsilon}{\Delta_k c}c|\mathcal{F}_0\right] \\
&\leq \mathbb{P}\left[\psi_k > \frac{\epsilon}{\Delta_k c}\,\mathbb{E}\left[\psi_k|\mathcal{F}_0\right]|\mathcal{F}_0\right] \\
&\leq \frac{c\Delta}{\epsilon}.
\end{aligned}$$

Since $\Delta_k \to 0$ almost surely, $\mathbb{P}\left[\left\|\nabla f(x^k)\right\| > \epsilon|\mathcal{F}_0\right] \to 0$. □

5 Numerical Results

We now compare four algorithms on a broad suite of stochastic optimization problems. Specifically, we compare three existing stochastic approximation methods—simultaneous perturbation stochastic approximation (SPSA), Kiefer-Wolfowitz (KW), and the stochastic version of QNSTOP [1]—and a prototype of Algorithm 1,

Table 1: Three relevant values of $\rho_{\hat{s}}$ and their interpretations.

Value	Interpretation
$\rho_{\hat{s}}(1)$	Fraction of problems method $\hat{s}$ solves first.
$\rho_{\hat{s}}(\hat{\alpha})$	Fraction of problems method $\hat{s}$ solves in fewer than $\hat{\alpha}$ times the number of function evaluations required by the best method.
$\lim_{\hat{\alpha}\to\infty} \rho_{\hat{s}}(\hat{\alpha})$	Fraction of problems method $\hat{s}$ eventually solves.

which is described in Section 5.2. The MATLAB implementations for a prototype of Algorithm 1, KW, and SPSA are available online at

`mcs.anl.gov/~jlarson/Stochastic`.

For SPSA and KW, we obtained MATLAB implementations from the website for Spall's textbook [25]. Specifically, we extracted MATLAB code for each algorithm from the file `SP_vs_FD.txt` to create separate MATLAB functions for each algorithm. We obtained the Fortran code for QNSTOP from Layne Watson.

5.1 Test set

The benchmark suite used to compare the different algorithms consists of the 53 stochastic problems from [16], where the computed function value has the form

$$\bar{f}(x) = \sum_{i=1}^{m} [F_i(x)]^2 + \epsilon \left(\sum_{i=1}^{m} \left[F_i(x^0)\right]^2 - F_i(\hat{x}^*) \right), \tag{21}$$

where $r \sim U[-0.1, 0.1]$, x^0 is a predefined starting point, and $\hat{x}^*$ is an estimate of the global minimum of the noiseless problem. The dimension of the 53 problems ranges from 2 to 12, and each method is given at most 5,000 function evaluations to solve each problem.

Although all the methods can access only the noise-corrupted values $\bar{f}$ when deciding which points to evaluate, the noiseless function value f (i.e., (21) with $r = 0$) is used in all benchmarking results. Because the methods are solving problems that are stochastic, each method was given 10 attempts to solve each of the 53 problems.

The relative performances of the three algorithms were compared by using performance profiles [11, 16]. If S is the set of optimization algorithms to be compared on a suite of problems P, let $t_{\hat{p},\hat{s}}$ be the number of function evaluations required for method $\hat{s} \in S$ on a problem $\hat{p} \in P$ to find a function value satisfying

$$f(x) \leq f_L + \tau \left(f(x^0) - f_L \right),$$

where f_L is the best function value achieved by any $\hat{s} \in S$. That is, we consider problem $\hat{p}$ to be "solved" by method $\hat{s}$ in $t_{\hat{p},\hat{s}}$ function evaluations when the method finds a function value less than τ times the reduction found by the best method. The performance profile of a solver $\hat{s} \in S$ is the following fraction:

$$\rho_{\hat{s}}(\hat{\alpha}) = \frac{1}{|P|} \left| \left\{ \hat{p} \in P : \frac{t_{\hat{p},\hat{s}}}{\min\{t_{\hat{p},\hat{s}} : \hat{s} \in S\}} \leq \hat{\alpha} \right\} \right|.$$

The performance measure $\rho_{\hat{s}}$ attempts to capture the relative performance of $\hat{s}$ versus that of the other methods in S on the set of problems P. Some relevant values of $\rho_{\hat{s}}$ are given in Table 1.

5.2 Description of prototype

To specify a particular implementation of Algorithm 1, we need to define how the model functions m_k and approximations F_k^0 and F_k^s are constructed and how the trust region subproblem is solved to compute s^k. For our prototype algorithm, we used the following simple strategies.

1. At the kth iteration, the model function m_k is constructed by using least-squares regression on a sample set of $(n+1)\zeta_k$ sample points, where ζ_k is defined by

$$\zeta_k = \left\lceil \frac{k}{10^8 \min\{1, \Delta_k^4\}} \right\rceil.$$

The sample set consists of ζ_k randomly rotated copies of the set $\{x^k, x^k + \Delta_k e_1, \ldots, x^k + \Delta_k e_n\}$, resulting in a strongly 1-poised set of points.

2. Since the model functions are linear, the trust region subproblem has the solution

$$s^k = -\Delta_k \frac{\nabla m_k(x^k)}{\|\nabla m_k(x^k)\|}.$$

3. The function approximations F_k^0 and F_k^s are computed by building linear model functions m_k^0 and m_k^s and evaluating $F_k^0 = m_k^0(x^k)$ and $F_k^s = m_k^s(x^k + s^k)$. The model functions are constructed by using least-squares regression on sample sets consisting of

$$\zeta_k = \left\lceil \frac{k}{10^8 a_k^4 \min\{1, \Delta_k^4\}} \right\rceil$$

randomly rotated copies of the sets $\{x^k, x^k + a_k\Delta_k e_1, \ldots, x^k + a_k\Delta_k e_n\}$ and $\{x^k + s^k, x^k + s^k + a_k\Delta_k e_1, \ldots, x^k + s^k + a_k\Delta_k e_n\}$, respectively, where a_k is the fraction of the trust region radius Δ_k that will be used to build the models F_k^0 and F_k^s.

4. The following parameter values are used: $\gamma_{\text{dec}} = 0.5$, $\gamma_{\text{inc}} = 2$, $a_k = 0.99^k$, $\eta = 10^{-6}$, $\beta = 0.5$, $\Delta_0 = 1$, and $k = 0$.

We make the following notes about the prototype outlined above.

1. The formula for ζ_k ensures that for k sufficiently large, F_k^0 and F_k^s satisfy (5) and (6), as shown by Proposition 2. The 10^8 term ensures that the number of copies is relatively small until Δ_k is fairly small. Nevertheless, for k sufficiently large, the model functions m_k still are α-probabilistically κ-fully linear on $B(x^k; \Delta_k)$, for some κ, where α is defined by (4). Therefore, we need not explicitly define the parameters $\{\alpha_k\}$.

2. The strategy above is inefficient in how it manages previous function evaluations, building every model with an entirely new set of points. We chose this approach because the proof of Proposition 3 requires that the function evaluations are independent processes; this cannot be directly assumed when using previously evaluated points. A more practical implementation would reuse old function values in constructing the model functions. Nevertheless, even though we throw out all previous function values, this prototype still outperforms SPSA and KW on a benchmark set of problems.

For SPSA and KW, the constants used to determine the sequence of step sizes follow the recommendations in Sections 6.6 and 7.5.2 of [25]. Specifically, for both algorithms we set the sequence of step sizes to $a_k = \dfrac{1}{(k+1+A)^{0.602}}$ and the sequence of finite-difference parameters to $c_k = \dfrac{1}{(k+1)^{0.101}}$, respectively, where A is one-tenth of the total budget of function evaluations per Spall's recommendations.

Because QNSTOP requires bound constraints, we set the domain for each problem to be

$$\left\{x : \left\|x - x^0\right\|_\infty \leq 10 \left\|x - \hat{x}^*\right\|_2\right\},$$

where $\hat{x}$ is an estimate of the problem's global minimum and x^0 is the starting point given to all algorithms.

Each algorithm was run on the 53 problems 10 times to create a set of 530 test problems. Figure 1 shows performance profiles for the four algorithms on this set of problems for $\tau = 0.1$ and $\tau = 0.01$. These

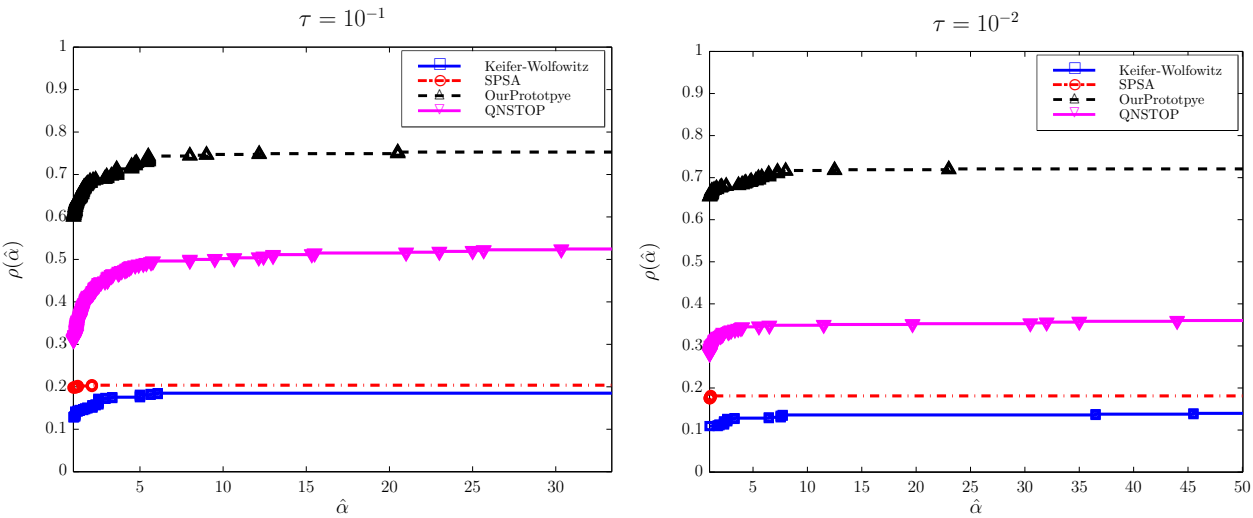

Figure 1: Performance profiles for four methods on a benchmark suite of stochastic problems.

tolerances imply that we consider the method to have located a sufficiently accurate solution for a problem if it finds a point where the true function value is within one-tenth (resp. one-hundredth) of the reduction found by the best solver on that problem. Since the absolute noise for the problems in the benchmark suite is one-tenth of the possible decrease in f, we consider $\tau = 0.1$ to be especially relevant. We can see that the prototype significantly outperforms QNSTOP, SPSA, and KW on the benchmark set of problems, solving 60% of the problems first and 75% of the problems within the alloted 5,000 function evaluations. All experimental results and performance profiles can be recreated by using the MATLAB scripts available at the aforementioned website.

6 Conclusion

In this paper, we present a trust region algorithm for minimizing a function when only noise-corrupted function values are available. Contrary to most other algorithms that solve this problem, our algorithm does not require a fixed pattern of points to build the local models, does not require repeated sampling, and adjusts its step sizes depending on how well the models approximate the behavior of the function.

The algorithm is designed to handle both stochastic and deterministic noise. However, our analysis considers only the case of stochastic noise. In this case, our main convergence result is that, under appropriate assumptions, the function gradients converge in probability to zero. While this is a relatively weak form of convergence, it is still useful in practice because it ensures that if the algorithm is terminated after a large number of iterations, then the function gradient at the last iterate will be small with high probability.

The strategy of avoiding repeated sampling is essential in the case of deterministic noise because repeated function evaluations at the same point provide no additional information. In the case of stochastic noise, the advantage of this strategy is less clear. For example, our implementation avoids repeated sampling at the cost of evaluating a large number of function values at nearby points. Conceptually, this strategy offers a potential advantage because sampling at distinct points provides information about the behavior of the function, whereas repeated sampling does not. However, translating this into a computational advantage remains a topic for further research.

We note that the ultimate goal is to modify this algorithm so it converges while still using every point that has been evaluated. Using every point greatly complicates convergence analysis because negative bias in the noise can cause convergence to points that are not stationary points of f. This algorithm does not

fully achieve this goal because one may still need to remove old points depending on how the local models are constructed. Nevertheless, a prototype of our algorithm that uses no previously evaluated point to construct any local model is seen to outperform existing stochastic approximation methods.

Acknowledgements

We thank Alexandre Proutiere for providing critical insights that allowed this work to be completed. We also thank Katya Scheinberg and an anonymous referee for alerting us to errors in earlier drafts of our analysis. We thank Layne Watson for sending us the Fortran code for QNSTOP. This material is based upon work supported by the U.S. Department of Energy, Office of Science, under Contract DE-AC02-06CH11357. We thank Gail Pieper for her useful language editing.

Appendix

We now prove a sequence of results culminating in a theorem showing that a model regressing a sufficiently large, strongly Λ-poised set of points will be α-probabilistically κ-fully linear.

Lemma 6. *Let f be continuously differentiable on Ω, and let $m(x)$ denote the linear model regressing a set of strongly Λ-poised points $Y = \{y^0, \ldots, y^p\}$. Then, for any $x \in \Omega$ the following identities hold:*

1. $m(x) - f(x) = \sum_{i=0}^{p} G_i(x)^T (y^i - x)\ell_i(x) + \sum_{i=0}^{p} \epsilon_i \ell_i(x),$
2. $\nabla m(x) - \nabla f(x) = \sum_{i=0}^{p} G_i(x)^T (y^i - x)\nabla\ell_i(x) + \sum_{i=0}^{p} \epsilon_i \nabla\ell_i(x),$

where ϵ_i denotes the noise at y^i, ℓ_i is the ith Lagrange polynomial, and $G_i(x) = \nabla f(v_i(x)) - \nabla f(x)$ for some point $v_i(x) = \theta x + (1-\theta)y^i, 0 \le \theta \le 1$ on the line segment connecting x to y^i.

Proof. The proof is nearly identical to that of [4, Lemma 4.5]. □

The following lemma is similar to [4, Lemma 4.4].

Lemma 7. *Given a sample set $Y = \{y^0, \ldots, y^p\} \subset \mathbb{R}^n$, let $\hat{Y} = \{\hat{y}^0, \ldots, \hat{y}^p\}$ be the shifted and scaled sample set defined by $\hat{y}^i = (y^i - \bar{y})/R$ for some $R > 0$ and $\bar{y} \in \mathbb{R}^n$. Let $\phi = \{\phi_0(x), \ldots, \phi_q(x)\}$ be a basis for $\mathcal{P}_n^d$, and define the basis $\hat{\phi} = \left\{\hat{\phi}_0(x), \ldots, \hat{\phi}_q(x)\right\}$ by $\hat{\phi}_i(x) = \phi_i(Rx + \bar{y})$, $i = 0, \ldots, n$. Let $\{\ell_0(x), \ldots, \ell_p(x)\}$ be regression Lagrange polynomials for Y, and let $\left\{\hat{\ell}_0(x), \ldots, \hat{\ell}_p(x)\right\}$ be regression Lagrange polynomials for $\hat{Y}$. Then $M(\phi, Y) = M(\hat{\phi}, \hat{Y})$. Moreover, if Y is poised, then*

$$\ell_j(x) = \hat{\ell}_j\left(\frac{x - \bar{y}}{R}\right), \quad \text{for } j = 0, \ldots, p.$$

Proof. The proof is identical to that of [4, Lemma 4.4]. (Note that the wording there gives specific choices for $R, \bar{y}$ and ϕ; however, its proof does not rely on these choices). □

Lemma 8. *Let $Y = \{y^0, \ldots, y^p\} \subset B(y^0; \Delta)$ be a strongly Λ-poised sample set on $B(y^0; \Delta)$. Let $\ell(x) = (\ell_0(x), \ldots, \ell_p(x))^T$, where $\ell_j(x)$, $j = 0, \ldots, p$, are the regression Lagrange polynomials or order d for Y. Let $q+1$ denote the dimension of $\mathcal{P}_n^d$. Then for any $x \in B(y^0; \Delta)$,*

$$\|\nabla\ell(x)\| \le \frac{(q+1)\hat{\theta}}{\Delta\sqrt{p+1}}\Lambda,$$

where $\hat{\theta}$ is a constant that depends on n and d but is independent of Y and Λ.

Proof. Let $\hat{Y} = \left\{\hat{y}^0, \ldots, \hat{y}^p\right\}$ be the shifted and scaled sample set defined by $\hat{y}^i = (y^i - y^0)/\Delta$.

Let $\bar{\phi} = \left\{\bar{\phi}_0(x), \ldots, \bar{\phi}_q(x)\right\}$ be a basis for $\mathcal{P}^d$, and define the basis $\hat{\phi} = \left\{\hat{\phi}_0(x), \ldots, \hat{\phi}_q(x)\right\}$, by $\hat{\phi}_i(x) = \bar{\phi}_i(\Delta x + y^0)$, $i = 0, \ldots, n$. Let $\{\ell_0(x), \ldots, \ell_p(x)\}$ be the regression Lagrange polynomials for Y, and let $\left\{\hat{\ell}_0(x), \ldots, \hat{\ell}_p(x)\right\}$ be the regression Lagrange polynomials for $\hat{Y}$.

Let $M = M(\hat{\phi}, \hat{Y})$ (defined in (2)). By the proof of [8, Lemma 4.12], $\left\|M(M^TM)^{-1}\right\| \leq \dfrac{(q+1)\bar{\theta}}{\sqrt{p+1}}\Lambda$ for some $\bar{\theta} > 0$ that depends on n and d but is independent of Y and Λ. (Note that the statement of [8, Lemma 4.12] assumes $\max_i \left\|\hat{y}^i\right\| = 1$; however, its proof does not rely on this assumption.) Thus, for any $x \in B(0;1)$,

$$\left\|\nabla\hat{\ell}(x)\right\| = \left\|M(M^TM)^{-1}\nabla\hat{\phi}(x)\right\| \leq \frac{(q+1)\bar{\theta}}{\sqrt{p+1}}\Lambda\left\|\nabla\hat{\phi}(x)\right\| \leq \frac{(q+1)\hat{\theta}}{\sqrt{p+1}}\Lambda,$$

where $\hat{\theta} = \bar{\theta}\max_{x\in B(0;1)}\left\|\nabla\hat{\phi}(x)\right\|$. By Lemma 7,

$$\|\nabla\ell(x)\| = \frac{1}{\Delta}\left\|\nabla\hat{\ell}\left(\frac{x-y^0}{\Delta}\right)\right\| \leq \frac{(q+1)\hat{\theta}}{\Delta\sqrt{p+1}}\Lambda.$$

□

We now use the preceding three results to bound the error between the function and a linear model regressing a strongly Λ-poised set of $p+1$ points.

Proposition 1. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function with Lipschitz continuous gradient with constant L_g. Let $\bar{x} \in \mathbb{R}^n$, $\Lambda > 0$ and $\Delta > 0$. Let $Y = \{y^0, \ldots, y^p\} \subset B(\bar{x};\Delta)$ be a strongly Λ-poised set of sample points with $y^0 = \bar{x}$. For $i \in \{0, \ldots, p\}$, let $\bar{f}_i = f(y^i) + \epsilon_i$, where the noise ϵ_i is sampled from a random distribution with mean 0 and variance $\sigma^2 < \infty$. Let $m : \mathbb{R}^n \to \mathbb{R}$ be the unique linear polynomial approximating the data $\{(y^i, \bar{f}_i), i = 0, \ldots, p\}$ by least-squares regression. There exist constants $\bar{C}$ and $\bar{a}$ such that for any $a > \bar{a}$ the following inequalities hold:*

1. $\mathbb{P}\left[\max_{z\in B(x;\Delta)}|m(z) - f(z)| > a\Delta^2\right] \leq \dfrac{\bar{C}\sigma^2\Lambda^2}{(a-\bar{a})^2(p+1)\Delta^4}$; *and*
2. $\mathbb{P}\left[\max_{z\in B(x;\Delta)}\|\nabla m(z) - \nabla f(z)\| > a\Delta\right] \leq \dfrac{\bar{C}\sigma^2\Lambda^2}{(a-\bar{a})^2(p+1)\Delta^4}$.

Proof. Let $\{\ell_0(x), \ldots, \ell_p(x)\}$ be the linear regression Lagrange polynomials for the set $Y = \left\{y^0, \ldots, y^p\right\}$. Define $\ell(z) = [\ell_0(x), \ldots, \ell_p(x)]^T$. Define $\epsilon = [\epsilon_0, \ldots, \epsilon_p]^T$, and note that $\mathbb{E}[\epsilon_i\epsilon_j] = 0$ for $i \neq j$. Define $Z_0(x) = \epsilon^T\ell(x)$. Then

$$\mathbb{E}\left[Z_0(\bar{x})^2\right] = \mathbb{E}\left[\left(\sum_{i=0}^p \epsilon_i\ell_i(\bar{x})\right)^2\right] = \sum_{i=0}^p \mathbb{E}\left[\epsilon_i^2\right]\ell_i(\bar{x})^2 = \sigma^2\sum_{i=0}^p \ell_i(\bar{x})^2 = \sigma^2\left\|\ell(\bar{x})\right\|^2 \leq \frac{\sigma^2(n+1)^2\Lambda^2}{p+1}, \quad (22)$$

where the last inequality follows from Definition 2 and the assumption that Y is strongly Λ-poised on $B(\bar{x};\Delta)$.

Because Z_0 is a linear function, its Jacobian matrix is constant. Denote this matrix by $Z_1 = \nabla Z_0(\bar{x}) = \epsilon^T\nabla\ell(\bar{x})$. Then,

$$\begin{aligned}\mathbb{E}\left[\|Z_1\|^2\right] &= \sum_{i=0}^p \mathbb{E}\left[\epsilon_i^2\right]\nabla\ell_i(\bar{x})^T\nabla\ell_i(\bar{x}) = \sigma^2\mathrm{Tr}\left(\nabla\ell(\bar{x})\nabla\ell(\bar{x})^T\right)\\ &= \sigma^2\left\|\nabla\ell(\bar{x})\right\|_F^2 \leq \sigma^2(n+1)\left\|\nabla\ell(\bar{x})\right\|_2^2\\ &\leq \frac{\sigma^2(n+1)^3\hat{\theta}^2\Lambda^2}{\Delta^2(p+1)}, \quad \text{by Lemma 8 ,} \end{aligned} \quad (23)$$

where $\hat{\theta}$ is a constant that depends on n but is independent of Y and Λ.

Next, observe that for any $z \in B(\bar{x};\Delta)$, $\|Z_0(z)\|^2 = \|Z_0(\bar{x}) + Z_1(z-\bar{x})\|^2 \leq (\|Z_0(\bar{x})\| + \Delta\|Z_1\|)^2 \leq 2\|Z_0(\bar{x})\|^2 + 2\Delta^2\|Z_1\|^2$. Combining this with (22) and (23) yields the inequality

$$\mathbb{E}\left[\max_{z\in B(\bar{x};\Delta)} \|Z_0(z)\|^2\right] \leq \frac{2\sigma^2(n+1)^2\Lambda^2}{p+1} + \frac{2\sigma^2(n+1)^3\hat{\theta}^2\Lambda^2}{p+1} = C_0\frac{\sigma^2\Lambda^2}{p+1},$$

where $C_0 = 2(n+1)^2\left(1+(n+1)\hat{\theta}^2\right)$.

By Markov's inequality, for any $a > 0$,

$$\mathbb{P}\left[\|Z_1\| \geq a\Delta\right] = \mathbb{P}\left[\|Z_1\|^2 \geq a^2\Delta^2\right] \leq \frac{\mathbb{E}\left[\|Z_1\|^2\right]}{a^2\Delta^2} \leq \frac{\sigma^2(n+1)^3\hat{\theta}^2\Lambda^2}{a^2\Delta^4(p+1)},$$

and

$$\mathbb{P}\left[\max_{z\in B(\bar{x};\Delta)} \|Z_0(z)\| \geq a\Delta^2\right] \leq \frac{\mathbb{E}\left[\max_{z\in B(\bar{x};\Delta)} \|Z_0(z)\|^2\right]}{a^2\Delta^4} \leq \frac{C_0\sigma^2\Lambda^2}{a^2\Delta^4(p+1)}.$$

Define $g_i(z) = G_i(z)^T(y^i - z)$, where $G_i(z)$ is defined in Lemma 6. Let $g(z) = [g_0(z), \ldots, g_p(z)]^T$. Because ∇f is Lipschitz continuous, $|g_i(z)| \leq 2L_g\Delta^2$ and $\|g(z)\| \leq 2\sqrt{p+1}L_g\Delta^2$. By Lemma 6,

$$\begin{aligned}
\|\nabla m(z) - \nabla f(z)\| &= \left\|\left(g(z)^T + \epsilon^T\right)\nabla\ell(\bar{x})\right\| \\
&\leq 2\sqrt{p+1}L_g\Delta^2\|\nabla\ell(\bar{x})\| + \|Z_1\| \\
&\leq 2L_g\Delta(n+1)\hat{\theta}\Lambda + \|Z_1\|, \quad \text{by Lemma 8.}
\end{aligned}$$

Thus,

$$\begin{aligned}
\mathbb{P}\left[\max_{z\in B(\bar{x};\Delta)} \|\nabla m(z) - \nabla f(z)\| \geq a\Delta\right] &\leq \mathbb{P}\left[\|Z_1\| \geq \left(a - 2L_g(n+1)\hat{\theta}\Lambda\right)\Delta\right] \\
&\leq \frac{\sigma^2(n+1)^3\hat{\theta}^2\Lambda^2}{\left(a - 2L_g(n+1)\hat{\theta}\Lambda\right)^2\Delta^4(p+1)} = \frac{C_1\sigma^2\Lambda^2}{(a-\bar{a}_1)^2(p+1)\Delta^4},
\end{aligned}$$

where $C_1 = (n+1)^3\hat{\theta}^2$ and $\bar{a}_1 = 2L_g(n+1)\hat{\theta}\Lambda$.

By Definition 2, $\|\ell(z)\| \leq \dfrac{n+1}{\sqrt{p+1}}\Lambda$. Thus, by Lemma 6,

$$\begin{aligned}
|m(z) - f(z)| &= \left|(g(z)+\epsilon)^T\ell(z)\right| \\
&\leq \|g(z)\|\,\|\ell(z)\| + \|Z_0(z)\| \\
&\leq 2L_g\sqrt{p+1}\Delta^2\frac{(n+1)\Lambda}{\sqrt{p+1}} + \|Z_0(z)\| = 2L_g\Delta^2(n+1)\Lambda + \|Z_0(z)\|.
\end{aligned}$$

Thus,

$$\begin{aligned}
\mathbb{P}\left[\max_{z\in B(\bar{x};\Delta)} |m(z) - f(z)| \geq a\Delta^2\right] &\leq \mathbb{P}\left[\max_{z\in B(\bar{x};\Delta)} \|Z_0(z)\| \geq (a - 2L_g(n+1)\Lambda)\Delta^2\right] \\
&\leq C_0\frac{\sigma^2\Lambda^2}{\left(a - 2L_g(n+1)\Lambda\right)^2\Delta^4(p+1)} = C_0\frac{\sigma^2\Lambda^2}{(a-\bar{a}_2)^2\Delta^4(p+1)},
\end{aligned}$$

where $\bar{a}_2 = 2L_g(n+1)\Lambda$. The result follows with $\bar{a} = \max\{\bar{a}_1, \bar{a}_2\}$ and $\bar{C} = \max\{C_0, C_1\}$. □

We now show that the models F_k^0 and F_k^s can easily be constructed to satisfy Conditions 1 and 2.

Proposition 2. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a Lipschitz continuously differentiable function. Let $\{c_k\}$, $\{\Delta_k\}$, and $\{a_k\}$ be positive sequences such that $c_k \to +\infty$ and $a_k \downarrow 0$. Let $\{x^k\}$ and $\{s^k\}$ be sequences in $\mathbb{R}^n$ with $\|s^k\| \leq \Delta_k$. Let $\Lambda > 0$, for each k define $\delta_k = \min\{\Delta_k, 1\}$, and let $Y_k^0 \subset B\left(x^k; a_k\delta_k\right)$ and $Y_k^s \subset B\left(x^k + s^k; a_k\delta_k\right)$ be strongly Λ-poised sample sets with at least $c_k/(a_k\delta_k)^4$ points each. Let m_k^0 and m_k^s be the linear polynomials fitting the computed function values on the sample sets Y_k^0 and Y_k^s, respectively, by least-squares regression. Define $F_k^0 = m_k^0\left(x^k\right)$ and $F_k^s = m_k^s\left(x^k + s^k\right)$. Then $\{F_k^0\}$ and $\{F_k^s\}$ satisfy Conditions 1 and 2.*

Proof. Let $\bar{C}$ and $\bar{a}$ be the constants defined in Proposition 1. Since $a_k \downarrow 0$ and $c_k \to +\infty$, then for any $\omega > 0$ there exists $\bar{k}(\omega)$ such that for all $k > \bar{k}(\omega)$, $2\bar{a}\left(a_k\delta_k\right)^2 < \dfrac{\beta\eta}{2}\delta_k^2 \leq \dfrac{\beta\eta}{2}\min\{\Delta_k, \Delta_k^2\}$ and $\dfrac{\bar{C}\sigma^2\Lambda^2}{\bar{a}^2\left|Y_k^0\right| a_k^4\delta_k^4} \leq \omega/2$. Thus,

$$\begin{aligned}
\mathbb{P}\left[\left|F_k^0 - f(x^k)\right| > \frac{\beta\eta}{2}\delta_k^2\right] &\leq \mathbb{P}\left[\left|m_k^0(x^k) - f(x^k)\right| > 2\bar{a}\left(a_k\delta_k\right)^2\right] \\
&\leq \frac{\bar{C}\sigma^2\Lambda^2}{\bar{a}^2\left\|Y_k^0\right\| a_k^4\delta_k^4} && \text{by Proposition 1} \\
&\leq \frac{\omega}{2}.
\end{aligned}$$

Using the same argument, we can show that

$$\mathbb{P}\left[\left|F_k^s - f(x^k + s^k)\right| > \frac{\beta\eta}{2}\delta_k^2\right] \leq \frac{\omega}{2}.$$

Thus, for $k > \bar{k}(\omega)$,

$$\mathbb{P}\left[\left|F_k^0 - f(x^k) - F_k^s + f(x^k + s^k)\right| > \beta\eta\min\left\{\Delta_k, \Delta_k^2\right\}\right] \leq \mathbb{P}\left[\left|F_k^0 - f(x^k)\right| > \frac{\beta\eta}{2}\delta_k^2\right] + \mathbb{P}\left[\left|F_k^s - f(x^k + s^k)\right| > \frac{\beta\eta}{2}\delta_k^2\right] \leq \omega,$$

so Condition 1 is satisfied.

To prove that Condition 2 holds, observe that for $k > \bar{k}(1)$, $(\beta\eta + \xi)\,\delta_k^2 > \left(2\bar{a} + \dfrac{\xi}{a_k^2}\right) a_k^2\delta_k^2$. Thus,

$$\begin{aligned}
\mathbb{P}\left[\left|F_k^0 - f(x^k)\right| > \frac{\beta\eta + \xi}{2}\delta_k^2\right] &\leq \mathbb{P}\left[\left|m_k^0(x^k) - f(x^k)\right| > \left(2\bar{a} + \frac{\xi}{a_k^2}\right)\left(a_k\delta_k\right)^2\right] \\
&\leq \frac{\bar{C}\sigma^2\Lambda^2}{\left(\bar{a} + \xi/a_k^2\right)^2\left\|Y_k^0\right\| a_k^4\delta_k^4} && \text{by Proposition 1} \\
&\leq \frac{\bar{a}^2}{2(\bar{a} + \xi/a_k^2)^2} \leq \frac{\bar{a}^2 a_k^2}{4\xi} \leq \frac{\theta}{2\xi},
\end{aligned}$$

for the constant $\theta = \max\limits_k \bar{a}a_k^2/2$. Similarly, we can show that $\mathbb{P}\left[\left|F_k^s - f(x^k + s^k)\right| > \dfrac{\beta\eta + \xi}{2}\delta_k^2\right] \leq \dfrac{\theta}{2\xi}$. It follows that

$$\begin{aligned}
\mathbb{P}\left[F_k^0 - f(x^k) + f(x^k + s^k) - F_k^s > (\beta\eta + \xi)\min\left\{\Delta_k, \Delta_k^2\right\}\right] &\leq \mathbb{P}\left[\left|F_k^0 - f(x^k)\right| > \frac{\beta\eta + \xi}{2}\delta_k^2\right] \\
&\quad + \mathbb{P}\left[\left|F_k^s - f(x^k + s^k)\right| > \frac{\beta\eta + \xi}{2}\delta_k^2\right] < \frac{\theta}{\xi}.
\end{aligned}$$

This proves that Condition 2 is satisfied. □

We next show that models built using our proposed method are α-probabilistically κ-fully linear.

Proposition 3. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function with Lipschitz continuous gradient with constant L_g. Let $\bar{x} \in \mathbb{R}^n$, $\Lambda > 0$ and $\Delta > 0$. Let $Y = \{y^0, \ldots, y^p\} \subset B(\bar{x}; \Delta)$ be a strongly Λ-poised set of sample points with $y^0 = \bar{x}$. For $i \in \{0, \ldots, p\}$, let $\bar{f}_i = f(y^i) + \epsilon_i$, where the noise ϵ_i is sampled from a random distribution with mean 0 and variance $\sigma^2 < \infty$. Let $m : \mathbb{R}^n \to \mathbb{R}$ be the unique linear polynomial approximating the data $\{(y^i, \bar{f}_i), i = 0, \ldots, p\}$ by least-squares regression. Let $\alpha \in (0, 1)$ be given. There exists a positive constant $\hat{C}$ depending only on Λ, and L_g and constants $\kappa = (\kappa_{\text{ef}}, \kappa_{\text{eg}})$ depending only on Λ, α, and L_g, such that if $p \geq \dfrac{\hat{C}}{\Delta_k^4}$, then m is α-probabilistically κ-fully linear.*

Proof. Let $\bar{C}$ and $\bar{a}$ be defined as in Proposition 1. Defining $\kappa_{\text{ef}} = \kappa_{\text{eg}} = a = \bar{a} + 1$ and $\hat{C} = \dfrac{\bar{C}\sigma^2\Lambda^2}{1 - \frac{\alpha}{2}}$, if $p \geq \dfrac{\hat{C}}{\Delta^4} - 1$, then by Proposition 1, we have

$$\mathbb{P}\left[\max_{z \in B(x;\Delta)} |m(z) - f(z)| > \kappa_{\text{ef}}\Delta^2\right] \leq \frac{\bar{C}\sigma^2\Lambda^2}{(p+1)\Delta^4} \leq 1 - \frac{\alpha}{2}$$

and

$$\mathbb{P}\left[\max_{z \in B(x;\Delta)} \|\nabla m(z) - \nabla f(z)\| > \kappa_{\text{eg}}\Delta\right] \leq \frac{\bar{C}\sigma^2\Lambda^2}{(p+1)\Delta^4} \leq 1 - \frac{\alpha}{2}.$$

Therefore, for any $y \in B(\bar{x}, \Delta)$,

$$\mathbb{P}\left[\|\nabla f(y) - \nabla m(y)\| \leq \kappa_{\text{eg}}\Delta \text{ and } |f(y) - m(y)| \leq \kappa_{\text{ef}}\Delta^2\right] \geq \alpha,$$

and the model is α-probabilistically κ-fully linear. □

One can easily test whether a set Y_k, with $|Y_k| > \dfrac{C}{\Delta_k}$ is strongly Λ-poised by using Theorem 4.12 in [8]. A partial set of points that are not strongly Λ-poised can be completed to a strongly Λ-poised set by using Algorithm 6.7 in [8].

References

[1] Amos, B.D., Easterling, D.R., Watson, L.T., Castle, B.S., Trosset, M.W., Thacker, W.I.: Fortran 95 Implementation of QNSTOP for Global and Stochastic Optimization. In: Proceedings of the High Performance Computing Symposium, pp. 15:1–15:8. Society for Computer Simulation International, San Diego, CA, USA (2014)

[2] Bandeira, A.S., Scheinberg, K., Vicente, L.N.: Convergence of Trust-Region Methods Based on Probabilistic Models. SIAM Journal on Optimization **24**(3), 1238–1264 (2014)

[3] Bastin, F., Cirillo, C., Toint, P.L.: An Adaptive Monte Carlo Algorithm for Computing Mixed Logit Estimators. Computational Management Science **3**(1), 55–79 (2006)

[4] Billups, S.C., Larson, J., Graf, P.: Derivative-Free Optimization of Expensive Functions with Computational Error Using Weighted Regression. SIAM Journal on Optimization **23**(1), 27–53 (2013)

[5] Box, G., Wilson, K.: On the Experimental Attainment of Optimum Conditions. Journal of the Royal Statistical Society. Series B (Methodological) **13**(1), 1–45 (1951)

[6] Chang, K.H., Hong, L.J., Wan, H.: Stochastic Trust-Region Response-Surface Method (STRONG)–A New Response-Surface Framework for Simulation Optimization. INFORMS Journal on Computing **25**(2), 230–243 (2012)

[7] Chen, R., Menickelly, M., Scheinberg, K.: Stochastic Optimization Using a Trust-Region Method and Random Models (2015). Preprint http://arxiv.org/abs/1504.04231

[8] Conn, A.R., Scheinberg, K., Vicente, L.N.: Introduction to Derivative-Free Optimization. Society for Industrial and Applied Mathematics (2009)

[9] Deng, G., Ferris, M.C.: Adaptation of the UOBYQA Algorithm for Noisy Functions. In: Proceedings of the 2006 Winter Simulation Conference, pp. 312–319. IEEE (2006)

[10] Deng, G., Ferris, M.C.: Extension of the Direct Optimization Algorithm for Noisy Functions. In: Proceedings of the 2007 Winter Simulation Conference, pp. 497–504. IEEE (2007)

[11] Dolan, E.D., Moré, J.J.: Benchmarking Optimization Software with Performance Profiles. Mathematical Programming **91**(2), 201–213 (2002)

[12] Dupuis, P., Simha, R.: On Sampling Controlled Stochastic Approximation. IEEE Transactions on Automatic Control **36**(8), 915–924 (1991)

[13] Jones, D.R., Perttunen, C.D., Stuckman, B.E.: Lipschitzian Optimization without the Lipschitz Constant. Journal of Optimization Theory and Applications **79**(1), 157–181 (1993)

[14] Kiefer, J., Wolfowitz, J.: Stochastic Estimation of the Maximum of a Regression Function. The Annals of Mathematical Statistics **23**(3), 462–466 (1952)

[15] Larson, J.: Derivative-free Optimization of Noisy Functions. PhD disseration, University of Colorado Denver (2012)

[16] Moré, J.J., Wild, S.M.: Benchmarking Derivative-free Optimization Algorithms. SIAM Journal on Optimization **20**(1), 172–191 (2009)

[17] Moré, J.J., Wild, S.M.: Estimating Computational Noise. SIAM Journal on Scientific Computing **33**(3), 1292–1314 (2011)

[18] Moré, J.J., Wild, S.M.: Estimating Derivatives of Noisy Simulations. ACM Transactions on Mathematical Software **38**(3), 1–21 (2012)

[19] Myers, R.H., Montgomery, D.C., Vining, G.G., Borror, C.M., Kowalski, S.M.: Response Surface Methodology: A Retrospective and Literature Survey. Journal of Quality Technology **36**(1), 53–77 (2004)

[20] Nedić, A., Bertsekas, D.P.: The Effect of Deterministic Noise in Subgradient Methods. Mathematical Programming **125**(1), 75–99 (2009)

[21] Nelder, J.A., Mead, R.: A Simplex Method for Function Minimization. The Computer Journal **7**(4), 308–313 (1965)

[22] Powell, M.: UOBYQA: Unconstrained Optimization by Quadratic Approximation. Mathematical Programming **92**(3), 555–582 (2002)

[23] Scheinberg, K.: Using Random Models in Derivative-Free Optimization (2012). Plenary presentation at "The International Symposium on Mathematical Programming"

[24] Spall, J.C.: Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation. IEEE Transactions on Automatic Control **37**(3), 332–341 (1992)

[25] Spall, J.C.: Introduction to Stochastic Search and Optimization. John Wiley & Sons, Inc., Hoboken, NJ, USA (2003)

[26] Tomick, J., Arnold, S., Barton, R.: Sample Size Selection for Improved Nelder-Mead Performance. In: Proceedings of the 1995 Winter Simulation Conference, pp. 341–345. IEEE (1995)