SIAM J. OPTIM.
Vol. 28, No. 4, pp. 3145–3176

ASTRO-DF: A CLASS OF ADAPTIVE SAMPLING TRUST-REGION ALGORITHMS FOR DERIVATIVE-FREE STOCHASTIC OPTIMIZATION*

SARA SHASHAANI†, FATEMEH S. HASHEMI‡, AND RAGHU PASUPATHY§

Abstract. We consider unconstrained optimization problems where only "stochastic" estimates of the objective function are observable as replicates from a Monte Carlo oracle. The Monte Carlo oracle is assumed to provide no direct observations of the function gradient. We present ASTRO-DF—a class of derivative-free trust-region algorithms, where a stochastic local model is constructed, optimized, and updated iteratively. Function estimation and model construction within ASTRO-DF is *adaptive* in the sense that the extent of Monte Carlo sampling is determined by continuously monitoring and balancing measures of sampling error (or variance) and structural error (or model bias) within ASTRO-DF. Such balancing of errors is designed to ensure that Monte Carlo effort within ASTRO-DF is sensitive to algorithm trajectory: sampling is higher whenever an iterate is inferred to be close to a critical point and lower when far away. We demonstrate the almost sure convergence of ASTRO-DF's iterates to first-order critical points when using stochastic polynomial interpolation models. The question of using more complicated models, e.g., regression or stochastic kriging, in combination with adaptive sampling is worth further investigation and will benefit from the methods of proof presented here. We speculate that ASTRO-DF's iterates achieve the canonical Monte Carlo convergence rate, although a proof remains elusive.

Key words. derivative-free optimization, simulation optimization, stochastic optimization, trust region

AMS subject classification. 90-XX

DOI. 10.1137/15M1042425

1. Introduction.

We consider unconstrained stochastic optimization (SO) problems, that is, optimization problems in continuous space where the objective function(s) can only be expressed implicitly via a Monte Carlo oracle. The Monte Carlo oracle is assumed to not provide any direct observations of the function derivatives even if they exist.

SO has recently gathered attention due to its versatile formulation, allowing the user to specify functions involved in an optimization problem implicitly, e.g., through a stochastic simulation. As a result, SO allows virtually any level of problem complexity to be embedded, albeit at the possible price of a computationally burdensome and slow Monte Carlo oracle. SO has seen wide recent adoption—see, for example, applications in telecommunication networks [33], traffic control [43], epidemic forecasting [42], and health care [1]. Recent editions of the Winter Simulation Conference (www.informs-sim.org) have dedicated an entire track to the SO problem and its various flavors. For a library of SO problems, see www.simopt.org and [47, 48].

*Received by the editors October 5, 2015; accepted for publication (in revised form) June 11, 2018; published electronically November 27, 2018.

http://www.siam.org/journals/siopt/28-4/M104242.html

Funding: The work of the first and second authors was supported by the National Science Foundation (NSF) Grants 1200162 and 107783 and the Office of Naval Research (ONR). The third author is thankful for the support provided by the Office of Naval Research Contract N000141712295 and the National Science Foundation Grant CMMI 1538050.

†School of Industrial Engineering, Purdue University, West Lafayette, IN 47906 (sshashaa@purdue.edu).

‡Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061 (fatemeh@vt.edu).

§Department of Statistics, Purdue University, West Lafayette, IN 47907 (pasupath@purdue.edu).

1.1. Problem statement. The SO problem we consider is formally stated as follows:

$$\text{Problem } P : \text{minimize } f(\boldsymbol{x}) \text{ subject to } \boldsymbol{x} \in \mathbb{R}^d, \tag{1.1}$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is differentiable, bounded from below, and has a Lipschitz continuous gradient. Furthermore, the function $f(\boldsymbol{x}) = \mathbb{E}[F(\boldsymbol{x})]$ is the expectation of a random function $F(\boldsymbol{x})$ that is observable through a Monte Carlo oracle. This means, for instance, that one can generate n identically distributed samples or replicates $F_j(\boldsymbol{x})$, $j = 1, 2, \ldots, n$, of $F(\boldsymbol{x})$ by "executing" the Monte Carlo oracle n times at the point $\boldsymbol{x}$. This leads to the estimator $\bar{F}(\boldsymbol{x}, n) = n^{-1} \sum_{j=1}^{n} F_j(\boldsymbol{x})$, which converges to $f(\boldsymbol{x})$ with probability one by the strong law of large numbers; furthermore, $\bar{F}(\boldsymbol{x}, n)$ is unbiased with respect to $f(\boldsymbol{x})$ and has variance $\sigma_F^2(\boldsymbol{x})/n$, where $\sigma_F^2(\boldsymbol{x})$ is estimated as $\hat{\sigma}_F^2(\boldsymbol{x}, n) = n^{-1} \sum_{j=1}^{n} (F_j(\boldsymbol{x}) - \bar{F}(\boldsymbol{x}, n))^2$. We also assume that no direct observations of the gradient $\nabla f(\cdot)$, e.g., through IPA [4, Page 214], are available via the Monte Carlo oracle. This means that optimization algorithms seeking gradient estimates need to resort to indirect gradient approximation methods such as finite differencing [4, Page 209], which lead to biased estimators.

An algorithm for solving the above problem will be evaluated based on its ability to return a (random) sequence of iterates $\{\boldsymbol{X}_k\}$ converging in some rigorously defined probabilistic metric to a first- or second-order critical point of the function f. Thus, each "run" of an SO algorithm will return a random sequence of iterates $\{\boldsymbol{X}_k\}$, and SO algorithms that return iterate sequences $\{\boldsymbol{X}_k\}$ guaranteed to converge to a critical point with probability one will be called *strongly consistent*.

1.2. Complications. The presence of a Monte Carlo oracle lends flexibility to the SO problem formulation, but it also brings with it a simply stated complication: the lack of uniform deterministic error guarantees. Specifically, suppose $\bar{F}(\boldsymbol{x}, n)$ is the Monte Carlo estimate of the unknown desired function value $f(\boldsymbol{x})$ at the point $\boldsymbol{x}$, and n represents the extent of Monte Carlo effort. Then simple probability arguments reveal that deterministic guarantees of the sort $|\bar{F}(\boldsymbol{x}, n) - f(\boldsymbol{x})| \leq \epsilon,\ \epsilon > 0$, do not hold irrespective of the size of n; instead, one has to be content with probabilistic precision guarantees of the form $\mathbb{P}\{|\bar{F}(\boldsymbol{x}, n) - f(\boldsymbol{x})| > \epsilon\} \leq \alpha$ for n sufficiently large but dependent on α. The analogous situation for function derivative estimation using Monte Carlo is worse. If the derivative estimate $\hat{\nabla} f(\boldsymbol{x}) := (\hat{\nabla}_1 f(\boldsymbol{x}), \hat{\nabla}_2 f(\boldsymbol{x}), \ldots, \hat{\nabla}_d f(\boldsymbol{x}))$ is constructed using a central-difference approximation as

$$\hat{\nabla}_i f(\boldsymbol{x}) = (2c_n)^{-1} (\bar{F}(\boldsymbol{x} + c_n \boldsymbol{e}_i, n) - \bar{F}(\boldsymbol{x} - c_n \boldsymbol{e}_i, n)), \quad i = 1, 2, \ldots, d,$$

then, as in the function estimation context, no uniform guarantees on the accuracy of $\hat{\nabla} f(\boldsymbol{x})$ are available in general. Furthermore, the rate at which $\hat{\nabla} f(\boldsymbol{x})$ converges to $\nabla f(\boldsymbol{x})$ depends crucially on the choice of $\{c_n\}$, with the best possible rate $\mathcal{O}(n^{-1/3})$ under generic Monte Carlo sampling being much slower than the corresponding $\mathcal{O}(n^{-1/2})$ rate for function estimation (see [4] for this and related results). Most importantly, implementing such finite-difference derivative estimates within an SO algorithm is well recognized to be a delicate issue, easily causing instabilities [4, Page 210]. In any event, the lack of uniform deterministic guarantees in the SO context means that estimation error inevitably accumulates across iterations of an algorithm, thereby threatening convergence guarantees of the resulting iterates. Algorithms for solving SO have to somehow contend with such potential nonconvergence due to mischance, either through the introduction of gain sequences as in stochastic

approximation [36] or through appropriate sampling as in sample average approximation or retrospective approximation [35, 44, 56].

A second complication within SO, but one that it partially shares with black-box deterministic optimization contexts, is the lack of information about function structure. Structural properties such as convexity, uni-modality, and differentiability, if known to be present, can be exploited when designing optimization algorithms. Such properties, when appropriate, are usually assumed within the deterministic context, and an appropriate solution algorithm devised. In SO, however, structural assumptions about the underlying true objective and constraint function, even if correct, may not provide as much leverage during algorithm development. This is because, due to the presence of stochastic error, the true objective and constraint functions are never directly observed, and making structural assumptions about their observed sample paths is far more suspect.

Another aspect that is unique to SO is noteworthy. Monte Carlo oracle calls are typically the most computation-intensive operations within SO contexts. Depending on the nature of the SO algorithm, different numbers of Monte Carlo oracle calls may be expended across iterations, e.g., constant in stochastic approximation (SA) [36], varying but predetermined in retrospective approximation (RA) [44, 49], or random as in sampling controlled stochastic recursions [46]. This means that the elemental measure of effort in SO—the number of Monte Carlo oracle calls—may not have a simple relationship with the notion of "iterations" defined within the specific SO algorithm, forcing a need for more careful bookkeeping. This is why iterative SO algorithms are well advised to measure convergence and convergence rates not in terms of the number of iterations, but rather in terms of the total number of Monte Carlo oracle calls.

1.3. ASTRO-DF and overview of contribution. Our particular focus in this research is that of developing a class of algorithms for solving low to moderate dimensional SO problems that have no readily discernible structure. We are inspired by the analogous problem in the deterministic context that has spurred the development of a special and arguably very useful class of optimization methods called model-based trust-region derivative-free (TRO-DF) algorithms [26, 51, 25, 5]. TRO-DF algorithms are typified by two aspects: (i) they eschew the direct computation and use of derivatives for searching, and instead rely on constructed models of guaranteed accuracy in specified "trust regions;" (ii) the algorithmic search evolves by repeatedly constructing and optimizing a local model within a dynamic trust region, explicitly restricting the distance between the successive iterates returned by the algorithm. The aspect in (i) is particularly suited for adaptation to SO contexts where direct derivative estimation can be delicate and unstable, requiring careful choice of step sizes (discussed in section 1.2). The aspect in (i) also aids efficiency because models constructed in previous iterations can be reused with some updating, and no effort is expended for explicit estimation of derivatives. The aspect in (ii) runs counter to efficiency, but is designed to reduce variance in the algorithm's iterates, through steps that are more circumspect.

We construct a family of adaptive sampling trust-region optimization derivative-free (ASTRO-DF) algorithms for the SO context. In their most rudimentary form, ASTRO-DF algorithms follow a familiar idea for iteratively estimating the first-order critical points of a function. Given a current random iterate $\boldsymbol{X}_k$ that approximates the first-order critical point of interest, ASTRO-DF constructs a tractable local stochastic model using Monte Carlo observations of the objective function at carefully chosen

points around $\boldsymbol{X}_k$. The constructed model is then optimized within the local region in which it is constructed to obtain a candidate solution $\tilde{\boldsymbol{X}}_{k+1}$. Next, the objective function is *observed* (using Monte Carlo) at $\tilde{\boldsymbol{X}}_{k+1}$ and compared to the value *predicted* by the model at $\tilde{\boldsymbol{X}}_{k+1}$. If the observed decrease in function values from $\boldsymbol{X}_k$ to $\tilde{\boldsymbol{X}}_{k+1}$ exceeds a fraction of the decrease predicted by the constructed model, the candidate $\tilde{\boldsymbol{X}}_{k+1}$ is accepted as the next iterate $\boldsymbol{X}_{k+1}$. As a vote of confidence on the constructed model, the trust-region radius then remains the same or is expanded by a factor. Otherwise, that is, if the predicted decrease is much more than the observed decrease, the candidate $\tilde{\boldsymbol{X}}_{k+1}$ is rejected and the local model is updated within the shrunk trust region in an attempt to improve accuracy. This iterative process then repeats to produce a random sequence of iterates $\{\boldsymbol{X}_k\}$ that is realized in each run of ASTRO-DF.

Remark 1. Throughout this paper, we use the term "sampling" to refer to the act of obtaining replicates using multiple runs of the Monte Carlo oracle at a fixed point. This is not to be confused with sampling design points in the search region. So, when we say that the sample size is n, we mean that n amount of Monte Carlo effort was expended to obtain the function estimate at a fixed point.

The above ideas for model construction, trust-region management, and candidate point acceptance say nothing about how much Monte Carlo sampling to expend. Since all observations for function estimation and model construction are based on Monte Carlo, the resulting accuracy estimates are at best probabilistic, leading us to the question of how much to sample. Too little Monte Carlo sampling threatens convergence due to accumulated stochastic and deterministic errors, and too much Monte Carlo sampling means reduced overall efficiency. Identifying the correct Monte Carlo sampling trade-off is more than a theoretical question, and answering it adequately entails more than broad prescriptions on sampling rates. To produce good implementations, sampling prescriptions ought to be automatic, that is, there should be no need for a user to choose sample sizes based on the problem at hand.

To resolve the issue of how much to sample, we propose that a simple strategy called *adaptive sampling* be incorporated within derivative-free trust-region algorithms in the SO context. Recognizing that the error in function and model estimation can be decomposed into error due to sampling or variance and error due to structure or bias, adaptive sampling seeks to ensure that "just adequate" Monte Carlo sampling is performed by balancing these errors. For example, when constructing a local model, Monte Carlo sampling in ASTRO-DF is adaptive in the sense that sampling continues until a certain continuously monitored measure of model quality exceeds a measure of sampling variability. A similar rule is employed when estimating the objective function at a point for purposes of candidate point acceptance. We believe that such adaptive sampling paves the way for efficiency because it reacts to the observed algorithm trajectory and, as we shall see, keeps the different sources of error within the algorithm commensurate. The resulting algorithm will likely remain practical because of the simplicity of the proposed adaptive sampling rule—sample until the estimated standard error falls below a certain specified exponent of the prevailing trust-region radius.

While invaluable as an implementation idea, adaptive sampling introduces substantial complications when analyzing algorithm behavior. Akin to what happens during sequential sampling in the context of confidence interval construction [21, 32], the explicit dependence of the extent of Monte Carlo sampling on algorithm trajectory causes systematic early stopping and consequent bias in the function estimates

obtained within ASTRO-DF. In other words, $\mathbb{E}[\bar{F}(\boldsymbol{x}, n)] \neq f(\boldsymbol{x})$ when using adaptive sampling since the sample size n is a *stopping time* that will depend on the algorithmic history (see Definition 2.6 and [29, Chapter 5]).

Demonstrating that ASTRO-DF's iterates converge to a first-order critical point with probability one entails demonstrating that the bias effects of adaptive sampling, especially when used within the derivative-free trust-region context, wear away asymptotically. We accomplish this by first generically characterizing a relationship between the moments of the adaptive sample size and the function estimates at stopping, and then showing that the errors induced due to model construction, algorithm recursion, and function estimation remain in *lock-step* throughout ASTRO-DF's evolution.

We note that ASTRO-DF, as presented here, assumes that, during the model construction step of the algorithm, a linear or quadratic model is built by interpolation of Monte Carlo observations. While such models are reasonable and have seen wide use in the analogous deterministic context, other possibly more powerful model construction techniques such as regression or stochastic kriging [3] could be considered in place of interpolation models, especially alongside adaptive sampling. Ongoing research investigates this question and it is our belief that the proof techniques that we present in this paper will carry over, albeit with some changes.

2. Preliminaries.

In this section we list notation, key definitions, standing assumptions, and some basic results that will be invoked throughout the rest of the document.

2.1. Notation and convention.

We use bold font for vectors, script font for sets, lower case for real numbers and upper case for random variables. Hence $\{\boldsymbol{X}_k\}$ denotes a sequence of random vectors in $\mathbb{R}^d$, $\boldsymbol{x} = (x^1, x^2, \ldots, x^d)$ denotes a d-dimensional vector of real numbers, $\mathcal{Y} := \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_p\}$ denotes a set of p real vectors, and $\mathcal{Y} := \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_p\}$ denotes a set of p random vectors. The set $\mathcal{B}(\boldsymbol{x}; r) = \{\boldsymbol{y} \in \mathbb{R}^d : \|\boldsymbol{y} - \boldsymbol{x}\| \leq r\}$ is the closed ball of radius $r > 0$ with center $\boldsymbol{x}$. For sequences $\{a_k\}$, $\{b_k\}$ of nonnegative reals we say $a_k \sim b_k$ if $\lim_{k\to\infty} a_k/b_k = 1$. For a sequence of random vectors $\{\boldsymbol{X}_k\}$, $\boldsymbol{X}_k \xrightarrow{\text{w.p.1}} \boldsymbol{X}$ denotes convergence with probability one or almost sure convergence. For a sequence of sets $\{A_n\}$ defined on a probability space, the set $\{A_n \text{ i.o.}\} := \bigcap_{n=1}^{\infty} \bigcup_{m=n}^{\infty} A_m$ refers to the event that "A_n happens infinitely often." The term "iid," also used multiple times throughout the paper, abbreviates independent and identically distributed. For a sequence of random variables $\{\boldsymbol{X}_k\}$, we say $\boldsymbol{X}_k = \mathcal{O}_p(1)$ if $\{\boldsymbol{X}_k\}$ is stochastically bounded, that is, given any $\epsilon > 0$ there exists a finite $M(\epsilon) > 0$ such that $\mathbb{P}\{\boldsymbol{X}_k \in (-M(\epsilon), M(\epsilon))\} \geq 1 - \epsilon$ for all $k \geq K(\epsilon) < \infty$.

2.2. Key definitions.

The following definitions will be invoked heavily during our exposition and analysis of ASTRO-DF. For further details on these definitions, consult [23, 24] and [26].

DEFINITION 2.1 (poised and Λ-poised sets). *Given $\boldsymbol{x} \in \mathbb{R}^d$ and $\Delta > 0$, let $\mathcal{Y}(\boldsymbol{x}, \Delta) = \{\boldsymbol{Y}_i \in \mathcal{B}(\boldsymbol{x}; \Delta),\ i = 1, 2, \ldots, p\}$ be a finite set of points in a closed ball of radius Δ around $\boldsymbol{x}$ and $\Phi(\boldsymbol{z}) = (\phi^1(\boldsymbol{z}), \phi^2(\boldsymbol{z}), \ldots, \phi^q(\boldsymbol{z}))$ be a polynomial basis on $\mathbb{R}^d$. Define*

$$P(\Phi, \mathcal{Y}(\boldsymbol{x}, \Delta)) = \begin{bmatrix} \phi^1(\boldsymbol{Y}_1) & \phi^2(\boldsymbol{Y}_1) & \ldots & \phi^q(\boldsymbol{Y}_1) \\ \phi^1(\boldsymbol{Y}_2) & \phi^2(\boldsymbol{Y}_2) & \ldots & \phi^q(\boldsymbol{Y}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi^1(\boldsymbol{Y}_p) & \phi^2(\boldsymbol{Y}_p) & \ldots & \phi^q(\boldsymbol{Y}_p) \end{bmatrix}. \tag{2.1}$$

Then, $\mathcal{Y}$ is said to be a "poised set" in $\mathcal{B}(\boldsymbol{x};\Delta)$ if the matrix $P(\Phi,\mathcal{Y})$ is nonsingular. A poised set $\mathcal{Y}$ is said to be "Λ-poised" in $\mathcal{B}(\boldsymbol{x};\Delta)$ if

$$\Lambda \geq \max_{j=1,\ldots,p} \max_{\boldsymbol{z}\in\mathcal{B}(\boldsymbol{x};\Delta)} |\ell_j(\boldsymbol{z})|,$$

where $\ell_j(\boldsymbol{z})$ are the Lagrange polynomials associated with $\boldsymbol{Y}_i \in \mathcal{Y}(\boldsymbol{x},\Delta)$.

DEFINITION 2.2 (polynomial interpolation models). *Let $f : \mathbb{X} \subseteq \mathbb{R}^d \to \mathbb{R}$ be a real-valued function, $\Delta > 0$, and let $\mathcal{Y}$ and Φ be as defined in Definition* 2.1 *with $p = q$. Suppose we can find $\boldsymbol{\alpha} = (\alpha^1, \alpha^2, \ldots, \alpha^p)$ such that*

$$P(\Phi,\mathcal{Y})\,\boldsymbol{\alpha} = \left(f(\boldsymbol{Y}_1),\ldots,f(\boldsymbol{Y}_p)\right)^T. \tag{2.2}$$

Note that such $\boldsymbol{\alpha}$ is guaranteed to exist if $\mathcal{Y}$ is poised. Then the function $m(\boldsymbol{z}) : \mathcal{B}(\boldsymbol{x};\Delta) \to \mathbb{R}$ given by

$$m(\boldsymbol{z}) = \sum_{j=1}^{p} \alpha^j \phi^{(j)}(\boldsymbol{z}) \tag{2.3}$$

is said to be a polynomial interpolation model of f on $\mathcal{B}(\boldsymbol{x};\Delta)$. As a special case, $m(\boldsymbol{z})$ is said to be a linear interpolation model of f on $\mathcal{B}(\boldsymbol{x};\Delta)$ if

$$\Phi(\boldsymbol{z}) := (1, z^1, z^2, \ldots, z^d),$$

and a quadratic interpolation model of f on $\mathcal{B}(\boldsymbol{x};\Delta)$ if

$$\Phi(\boldsymbol{z}) := (1, z^1, z^2, \ldots, z^d, \tfrac{1}{2}(z^1)^2, z^1z^2, \ldots, \tfrac{1}{2}(z^2)^2, \ldots, \tfrac{1}{2}(z_d)^2).$$

Hence, for the former $p = d+1$ and for the latter $p = (d+1)(d+2)/2$ number of points are required.

DEFINITION 2.3 (stochastic polynomial interpolation models). *A model constructed as described in Definition* 2.2 *but with Monte Carlo sampled function estimates will henceforth be called a stochastic interpolation model. Analogous to* (2.2)*, suppose $\hat{\boldsymbol{\alpha}} = (\hat{\alpha}^1, \hat{\alpha}^2, \ldots, \hat{\alpha}^p)$ is such that*

$$P(\Phi,\mathcal{Y})\,\hat{\boldsymbol{\alpha}} = \left(\bar{F}(\boldsymbol{Y}_1, n(\boldsymbol{Y}_1)), \bar{F}(\boldsymbol{Y}_2, n(\boldsymbol{Y}_2)), \ldots, \bar{F}(\boldsymbol{Y}_p, n(\boldsymbol{Y}_p))\right)^T.$$

Then the stochastic function $M(\boldsymbol{z}) : \mathcal{B}(\boldsymbol{x};\Delta) \to \mathbb{R}$ given as $M(\boldsymbol{z}) = \sum_{j=1}^{p} \hat{\alpha}^j \phi^{(j)}(\boldsymbol{z})$ is said to be a stochastic polynomial interpolation model of f on $\mathcal{B}(\boldsymbol{x};\Delta)$, where $\Phi(\boldsymbol{z}) = (\phi^1(\boldsymbol{z}), \phi^2(\boldsymbol{z}), \ldots, \phi^q(\boldsymbol{z}))$ and $P(\Phi,\mathcal{Y})$ are as in Definition 2.1*.*

DEFINITION 2.4 (fully linear models). *Given $\boldsymbol{x} \in \mathbb{R}^d$ and $\Delta > 0$, $m(\boldsymbol{z}) : \mathcal{B}(\boldsymbol{x};\Delta) \to \mathbb{R}$, $m \in \mathcal{C}^1$, is said to be a $(\kappa_{ef},\kappa_{eg})$-fully linear model of f on $\mathcal{B}(\boldsymbol{x};\Delta)$ if it has a Lipschitz continuous gradient with Lipschitz constant ν_{gL}^m, and there exist constants κ_{ef} and κ_{eg} (not dependent on $\boldsymbol{z}$ and Δ) such that*

$$\begin{aligned} |f(\boldsymbol{z}) - m(\boldsymbol{z})| &\leq \kappa_{ef}\Delta^2, \\ \|\nabla f(\boldsymbol{z}) - \nabla m(\boldsymbol{z})\| &\leq \kappa_{eg}\Delta. \end{aligned} \tag{2.4}$$

A linear interpolation model of the function f constructed using a poised set $\mathcal{Y}$ of points can be shown to be $(\kappa_{ef},\kappa_{eg})$-fully linear when the function f is continuously differentiable having Lipschitz gradients [26]. Numerous other types of interpolation and regression models can be constructed to satisfy the full-linearity condition in (2.4). We will not go into further detail on different model constructions considering the different focus of this paper.

DEFINITION 2.5 (Cauchy reduction). *Step $\boldsymbol{s}$ is said to achieve a κ_{fcd} fraction of Cauchy reduction for $m(\cdot)$ on $\mathcal{B}(\boldsymbol{x};\Delta)$ with some $\Delta > 0$ if*

$$m(\boldsymbol{x}) - m(\boldsymbol{x}+\boldsymbol{s}) \geq \frac{\kappa_{fcd}}{2} \|\nabla m(\boldsymbol{x})\| \min\left\{ \frac{\|\nabla m(\boldsymbol{x})\|}{\|\nabla^2 m(\boldsymbol{x})\|}, \Delta \right\}, \tag{2.5}$$

where $\nabla m(\boldsymbol{x})$ and $\nabla^2 m(\boldsymbol{x})$ are the model gradient and the model Hessian at point $\boldsymbol{x}$. We assume $\|\nabla m(\boldsymbol{x})\|/\|\nabla^2 m(\boldsymbol{x})\| = +\infty$ when $\nabla^2 m(\boldsymbol{x}) = \boldsymbol{0}$. A Cauchy step with $\kappa_{fcd} = 1$ is obtained by minimizing the model $m(\cdot)$ along the steepest descent direction within $\mathcal{B}(\boldsymbol{x};\Delta)$ [26, Page 175]. A Cauchy step is easy to obtain when $m(\cdot)$ is quadratic; for linear $m(\cdot)$, the model is minimized at $\boldsymbol{x}+\boldsymbol{s}$, $\boldsymbol{s} = -\Delta \frac{\nabla m(\boldsymbol{x})}{\|\nabla m(\boldsymbol{x})\|}$ and thus $m(\boldsymbol{x}) - m(\boldsymbol{x}+\boldsymbol{s}) = -\boldsymbol{s}^T \nabla m(\boldsymbol{x}) = \|\nabla m(\boldsymbol{x})\|\Delta$.

DEFINITION 2.6 (sample path, filtration, and stopping time). *For the stochastic process $\{\boldsymbol{X}_k : k \in \mathbb{N}\}$ defined on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$, we refer to the mapping $\boldsymbol{X}_\cdot(\omega) : \mathbb{N} \to \mathbb{R}$ (for fixed $\omega \in \Omega$) as a "sample path." Since the sample path $\boldsymbol{X}_\cdot(\omega)$ is fixed for each ω, we will sometimes refer to ω itself as the sample path.*

A "filtration" $\{\mathcal{F}_k\}$ is an increasing family of σ-algebras of $\mathcal{F}$, that is, $\mathcal{F}_k \subseteq \mathcal{F}_{k+1} \subseteq \mathcal{F}$ for all k. We can think of $\mathcal{F}_k$ as the information available at time k.

We say that the random variable N is a "stopping time" with respect to the filtration $\{\mathcal{F}_k\}$ if $\{N = n\} \in \mathcal{F}_k$, that is, the event $\{N = n\}$ for any n can be judged to have happened (or not) based on "current information." See [12] *for more on these notions.*

2.3. Standing assumptions. We now list two standing assumptions relating to the problem. Three additional assumptions relating to the algorithm will be made in section 5.

Assumption 1. The function $f : \mathbb{R}^d \to \mathbb{R}$ is bounded from below and has gradient $\nabla f(\cdot)$ that is Lipschitz continuous, that is, there exists $\nu_{gL} < \infty$ such that

$$\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\| \leq \nu_{gL} \|\boldsymbol{x} - \boldsymbol{y}\|$$

for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$.

Assumption 2. When executed at $\boldsymbol{X}_k$, the Monte Carlo oracle generates iid random variates $F_j(\boldsymbol{X}_k) = f(\boldsymbol{X}_k) + \xi_j \,|\, \mathcal{F}_k$, where $\mathcal{F}_k$ is the filtration formed by all visited points (and their associated function estimates) up to iteration k. Furthermore, $\mathbb{E}[\xi_j \,|\, \mathcal{F}_k] = 0$, $\mathbb{E}[\xi_j^2 \,|\, \mathcal{F}_k] = \sigma^2 < \infty$ for all k, and $\sup_k \mathbb{E}[|\xi_j|^{4v} \,|\, \mathcal{F}_k] < \infty$ for some $v \geq 2$.

Since the "simulation error" $\xi_j | \mathcal{F}_k$ has zero mean, it allows the direct application of Theorem 2.7 for characterizing the behavior of the Monte Carlo function estimates in ASTRO-DF. We believe that a bound on the $4v$th moment of the error $\xi_j | \mathcal{F}_k$, as stipulated through Assumption 2, is reasonable. This assumption is less stringent (for any $v < \infty$) than assuming that the moment generating function of the error exists in a neighborhood of zero, which implies the existence of all moments. Commonly encountered distributions, e.g., normal, gamma, beta, have all finite moments; any distribution with bounded support has all finite moments.

Remark 2. The uniform (across $\boldsymbol{x}$) error variance assumption $\mathbb{E}[\xi_j^2 \,|\, \mathcal{F}_k] = \sigma^2$ in Assumption 2 may be violated in settings where the error is proportional to the objective function value and the objective function $f(\boldsymbol{x}) \to \infty$ as $\|\boldsymbol{x}\| \to \infty$. Such contexts could be addressed partially through a more general theory that assumes $\mathbb{E}[\xi_j^2 | \mathcal{F}_k] \leq \sigma^2(1 + \|\boldsymbol{x}\|^2)$ akin to [15] and [16].

2.4. Useful results. We will now state some basic results that will be used at various points in the paper.

The first result (Theorem 2.7) is a variation of Lemma 2 and Theorem 1 in [31], which originally appeared in the context of sequential confidence intervals. The result characterizes the behavior of the sample mean $\bar{X}_n$ of a random number of iid random variables with mean zero. The sample size $n = N(\lambda) \geq \lambda$ governing $\bar{X}_n$ is such that sampling continues until the standard error of $\bar{X}_n$ falls below a specific threshold $\kappa/\sqrt{\lambda}$. The adaptive sampling rule we use in this paper broadly conforms to this sampling scheme, and we will hence find extensive use for Theorem 2.7 in our analysis. The postulates and the setting of Theorem 2.7 differ only trivially from the setting of Theorem 1 in [31]; for this reason, we have chosen not to include a proof.

THEOREM 2.7. *Suppose random variables X_i, $i = 1, 2, \ldots$, are iid with $\mathbb{E}[X_1] = 0$, $\mathbb{E}[X_1^2] = \sigma^2 > 0$, and $\mathbb{E}[|X_1|^{4v}] < \infty$ for some $v \geq 2$. Let*

$$\hat{\sigma}_n^2 = n^{-1} \sum_{i=1}^{n} (X_i - \bar{X}_n)^2,$$

where $\bar{X}_n = n^{-1} \sum_{i=1}^{n} X_i$. If

$$N(\lambda) = \inf \left\{ n \geq \lambda : \frac{\hat{\sigma}_n}{\sqrt{n}} \leq \frac{\kappa}{\sqrt{\lambda}} \right\},$$

then $N(\lambda) \xrightarrow{w.p.1} \infty$ and $\mathbb{E}[\bar{X}_{N(\lambda)}^2] \sim \kappa^2 \lambda^{-1}$ as $\lambda \to \infty$.

The next result is from [21], again in the sequential confidence interval context, and shows for a specified sampling rule the estimated standard deviation of the sample mean converges to the true standard deviation almost surely.

THEOREM 2.8. *Suppose random variables X_i, $i = 1, 2, \ldots$, are iid with variance $\sigma^2 < \infty$, $\bar{X}_n = n^{-1} \sum_{i=1}^{n} X_i$, $\hat{\sigma}_n^2 = n^{-1} \sum_{i=1}^{n} (X_i - \bar{X}_n)^2$, and $\{a_n\}$ a sequence of positive constants such that $a_n \to a$ as $n \to \infty$. If*

$$N(d) = \inf \left\{ n \geq 2 : \frac{\hat{\sigma}_n}{\sqrt{n}} \leq \frac{d}{a_n} \right\},$$

then $\hat{\sigma}_{N(d)}/\sigma \xrightarrow{w.p.1} 1$ as $d \to 0$.

We next state Lemma 2.9, which characterizes the error in the stochastic polynomial interpolation model introduced in Definition 2.3. Lemma 2.9 is essentially a stochastic variant of a result that appears in Chapter 3 of [26]. We provide a sketch of the proof of Lemma 2.9 in the appendix.

LEMMA 2.9. *Let Assumption 1 hold and let $\mathcal{Y} = \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_p\}$ be a Λ-poised set on $\mathcal{B}(\boldsymbol{Y}_1; \Delta)$. Let $m(\boldsymbol{z})$ be a polynomial interpolation model of f on $\mathcal{B}(\boldsymbol{Y}_1; \Delta)$. Let $M(\boldsymbol{z})$ be the corresponding stochastic polynomial interpolation model of f on $\mathcal{B}(\boldsymbol{Y}_1; \Delta)$ constructed using observations $\bar{F}(\boldsymbol{Y}_i, n(\boldsymbol{Y}_i)) = f(\boldsymbol{Y}_i) + E_i$ for $i = 1, 2, \ldots, p$.*

(i) *For all $\boldsymbol{z} \in \mathcal{B}(\boldsymbol{Y}_1; \Delta)$,*

$$|M(\boldsymbol{z}) - m(\boldsymbol{z})| \leq p\Lambda \max_{i=1,2,\ldots,p} \left| \bar{F}(\boldsymbol{Y}_i, n(\boldsymbol{Y}_i)) - f(\boldsymbol{Y}_i) \right|.$$

(ii) *If $M(\boldsymbol{z})$ is a stochastic linear interpolation model (augmented with a quadratic term) or a stochastic quadratic interpolation model of f on $\mathcal{B}(\boldsymbol{Y}_1; \Delta)$, then*

there exist positive constants κ_{eg1} and κ_{eg2} (that depend on Λ and ν_{gL}) such that for $\boldsymbol{z} \in \mathcal{B}(\boldsymbol{Y}_1; \Delta)$,

$$\|\nabla M(\boldsymbol{z}) - \nabla f(\boldsymbol{z})\| \leq \kappa_{eg1}\Delta + \kappa_{eg2}\frac{\sqrt{\sum_{i=2}^{p}(E_i - E_1)^2}}{\Delta}.$$

We end this section by stating two other basic results that we repeatedly invoke, and which can be found in most standard treatments of probability such as [12]. The first of these is used to provide an upper bound to the probability of the union of events; the second provides sufficient conditions to ensure that the probability of an infinite number of a collection of events $\{A_n\}$ happening is zero.

LEMMA 2.10 (Boole's inequality). *Let $A_1, A_2, \ldots$ be a countable set of events defined on a probability space. Then $\mathbb{P}(\bigcup_i A_i) \leq \sum_i \mathbb{P}(A_i)$. We thus see that if the random variables X and X_i, $i = 1, 2, \ldots, q$, satisfy $X \leq X_1 + X_2 + \cdots + X_q$, then*

$$\mathbb{P}\{X > c\} \leq \mathbb{P}\left\{\bigcup_{i=1}^{q}\left(X_i > \frac{c}{q}\right)\right\} \leq \sum_{i=1}^{q}\mathbb{P}\left\{X_i > \frac{c}{q}\right\}.$$

LEMMA 2.11 (the first Borel–Cantelli lemma). *For a sequence $A_1, A_2, \ldots$ of events defined on a probability space, if $\sum_{n=1}^{\infty}\mathbb{P}\{A_n\} < \infty$, then the probability $\mathbb{P}\{A_n \text{ i.o.}\}$ of A_n happening "infinitely often" is zero.*

3. Related work. Much progress has been made on solving various flavors of the SO problem. The predominant solution methods in the simulation literature fall into two broad categories called stochastic approximation (SA) and sample-average approximation (SAA). SA and SAA have enjoyed a long history with mature literature in theory and algorithms. More recently, newer classes of algorithms that can be described as "stochastic versions" of iterative structures in the deterministic context have emerged.

3.1. SA and SAA. Virtually all stochastic-approximation-type methods are subsumed by the following generic form:

$$\boldsymbol{X}_{k+1} = \Pi_{\mathcal{D}}(\boldsymbol{X}_k - a_k\boldsymbol{G}_k), \tag{3.1}$$

where $\Pi_{\mathcal{D}}(\boldsymbol{x})$ is the projection of the point $\boldsymbol{x}$ onto a set $\mathcal{D}$, $\{a_k\}$ is a user-chosen positive-valued scalar sequence, and $\boldsymbol{G}_k = (G_k^1, G_k^2, \ldots, G_k^d)$ is an estimator of the gradient $\nabla f(\boldsymbol{X}_k)$ of the function f at the point $\boldsymbol{X}_k$. When direct Monte Carlo observations of the objective function f are available, the most common expression for $\boldsymbol{G}_k$ is either the central-difference approximation

$$G_k^i = (2c_k)^{-1}(F(\boldsymbol{X}_k + c_k\boldsymbol{e}_i) - F(\boldsymbol{X}_k - c_k\boldsymbol{e}_i))$$

or the forward-difference approximation $G_k^i = c_k^{-1}(F(\boldsymbol{X}_k + c_k\boldsymbol{e}_i) - F(\boldsymbol{X}_k))$ of the gradient $\nabla f(\boldsymbol{X}_k)$, where $\{c_k\}$ is a positive-valued sequence and $F : \mathbb{R}^d \to \mathbb{R}$ is the observable estimator of the objective function $f : \mathbb{R}^d \to \mathbb{R}$. The resulting recursion is the famous Kiefer–Wolfowitz process [14, 34]. More recent recursions include an estimated Hessian $H_k(\cdot)$ of the function f at the point $\boldsymbol{X}_k$,

$$\boldsymbol{X}_{k+1} = \Pi_{\mathcal{D}}\left(\boldsymbol{X}_k - a_k H_k^{-1}\boldsymbol{G}_k\right), \tag{3.2}$$

making the resulting recursion look closer to the classical Newton iteration in the deterministic context. The Hessian estimator $H_k(\cdot)$ has d^2 entries, and hence, most

methods that use (3.2) estimate $H_k(\cdot)$ either by using a parsimonious design (e.g., [57, 58]) or by constructing it from the history of observed points.

As can be seen in (3.1), the SA recursion is simply stated and implemented, and little has changed in its basic structure since 1951, when it was first introduced by Robbins and Monro [52] in the context of finding a zero of a "noisy" vector function. Instead, much of the research over the ensuing decades has focused on questions such as convergence and convergence rates of SA-type algorithms, the effect of averaging on the consistency and convergence rates of the iterates, and efforts to choose the sequence $\{a_k\}$ in an adaptive fashion. Some good entry points into the broad SA literature include [36, 40, 45, 50]. Recent attempts [17, 18, 60] address the persistent dilemma of choosing the step-size sequence $\{a_k\}$ to ensure good practical performance.

SAA, in contrast to SA, is more a framework than an algorithm to solve SO problems. Instead of solving Problem P, SAA asks to solve a "sample-path" problem P_n (to optimality) to obtain a solution estimator $\boldsymbol{X}_n^*$. Formally, in the unconstrained context SAA seeks to solve

$$\text{Problem } P_n: \text{ minimize } f(\boldsymbol{x}, n) \text{ subject to } \boldsymbol{x} \in \mathbb{R}^d, \tag{3.3}$$

where $f(\boldsymbol{x}, n)$ is computed using a "fixed" sample of size n.

SAA is attractive in that problem P_n becomes a *deterministic* optimization problem and SAA can bring to bear all of the advances in deterministic nonlinear programming methods [11, 41] of the last few decades. SAA has been the subject of a tremendous amount of theoretical and empirical research over the last two decades. For example the conditions that allow the transfer of structural properties from the sample path to the limit function $f(\boldsymbol{x})$ [35, Propositions 1, 3, and 4], the sufficient conditions for the consistency of the optimal value and solution of problem P_n assuming the numerical procedure in use within SAA can produce global optima [56, Theorem 5.3], consistency of the set of stationary points of problem P_n [6, 56], convergence rates for the optimal value [56, Theorem 5.7] and optimal solution [35, Theorem 12], expressions for the minimum sample size m that provides probabilistic guarantees on the optimality gap of the sample-path solution [54, Theorem 5.18], methods for estimating the accuracy of an obtained solution [8, 9, 39], and quantifications of the trade-off between searching and sampling [53] have all been thoroughly studied. SAA is usually not implemented in the vanilla form P_n due to known issues relating to an appropriate choice of the sample size n. There have been recent advances [8, 9, 10, 28, 45, 46] aimed at defeating the issue of sample-size choice.

3.2. Stochastic TRO. Two algorithms that are particularly noteworthy competitors to what we propose here are STORM [20] and the recently proposed algorithm by Larson and Billups [37] (henceforth LB2014). While the underlying logic in both of these algorithms is similar, differences arise in terms of what has been assumed about the quality of the constructed models and how such quality can be achieved in practice. For instance, STORM treats the context of biased estimators, that is, contexts where $\mathbb{E}[\bar{F}(\boldsymbol{x}, n)] \neq f(\boldsymbol{x})$. Consistency in STORM relies on models assumed to be constructed of a specified quality (characterized through the notion of probabilistic full linearity) with a probability exceeding a fixed threshold. Crucially, the sample means for such a model construction use a sample size that is derived using the Chebyshev inequality with an assumed upper bound on the variance. By contrast, the sample sizes in ASTRO-DF are determined adaptively by balancing squared bias and variance estimates for the function estimator. While this makes the sample size in ASTRO-DF a stopping time [12], thereby complicating proofs, such adaptive sam-

pling enables ASTRO-DF to differentially sample across the search space, leading to efficiency. Like ASTRO-DF, STORM exhibits almost sure convergence to a first-order critical point, but with certain stipulations on algorithmic parameters that are not immediately verifiable.

LB2014, like STORM, uses random models. Unlike STORM, however, the sequence of models constructed in LB2014 is assumed to be accurate (in a certain sense) across iterations with a probability sequence that converges to one. A related version [13] of LB2014 addresses the case of differing levels of (spatial) stochastic error through the use of weighted regression schemes, where the weights are chosen heuristically.

Another noteworthy algorithm for the context we consider in this paper is VNSP proposed by Deng and Ferris [27, 28]. VNSP uses a quadratic interpolation model within a trust-region optimization framework, and is derivative free in the sense that only function estimates are assumed to be available. Model construction, inference, and improvement, along with (nondecreasing) sample-size updates happen within a Bayesian framework with an assumed Gaussian conjugate prior. Convergence theory for VNSP is accordingly within a Bayesian setting.

In the slightly more tangential context where unbiased gradient estimates are assumed to be available, a number of trust-region-type algorithms have emerged in the last decade or so. STRONG or the stochastic trust-region response-surface method [19], for instance, is an adaptive sampling trust-region algorithm for solving SO problems that is in the spirit of what we propose here. A key feature of STRONG is local model construction through a design of experiments combined with a hypothesis testing procedure. STRONG assumes that the error in the derivative observations are additive and have a Gaussian distribution. Amos et al. [2] and Bastin, Cirillo, and Toint [7] provide two other examples of algorithms that treat the setting where unbiased observations of the gradient are assumed to be available. (The former, in fact, assumes that unbiased estimates of the Hessian of the objective function are available.) Bastin, Cirillo, and Toint [7] is specific to the problem of estimation within mixed-logit models.

4. ASTRO-DF overview and algorithm listing. ASTRO-DF is an adaptive sampling trust-region derivative-free algorithm whose essence is encapsulated within four repeating stages: (a) local stochastic model construction and certification through adaptive sampling, (b) optimization of the constructed model in a trust region for identifying the next candidate solution, (c) estimation of the objective function at the next candidate solution through adaptive sampling, and (d) iterate and trust-region update based on a (stochastic) sufficient decrease check. These stages appear with italic labels in Algorithm 1. In what follows, we describe each step of Algorithm 1 in further detail.

During step 2, Algorithm 1 executes `AdaptiveModelConstruction` to obtain a $(\kappa_{ef}, \kappa_{eg})$ stochastic linear model $M_k(\boldsymbol{z})$, $\boldsymbol{z} \in \mathcal{B}(\boldsymbol{X}_k; \Delta_k)$, that satisfies the error bound outlined in part (ii) of Lemma 2.9. To obtain a quadratic model, a quadratic term can be augmented to the stochastic linear model constructed through `AdaptiveModelConstruction`, or a stochastic quadratic model can be constructed in a fashion similar to what is outlined in `AdaptiveModelConstruction`. In all these cases, the error bound outlined in part (ii) of Lemma 2.9 is satisfied and Algorithm 2 terminates in finitely many steps with probability one—see Lemma C.1 in the appendix.

During the jth iteration of the contraction loop (steps 2–9) in Algorithm 2, (a) a Λ-poised set $\mathcal{Y}_k^{(j)} \triangleq \{\boldsymbol{Y}_{k,1}^{(j)}, \boldsymbol{Y}_{k,2}^{(j)}, \ldots, \boldsymbol{Y}_{k,p}^{(j)}\}$ in the "candidate" trust region on a radius

Algorithm 1 ASTRO-DF main algorithm.

Require: Initial guess $\boldsymbol{x}_0 \in \mathbb{R}^d$, initial trust-region radius $\Delta_0 > 0$ and maximum radius $\Delta_{\max} > 0$, model "fitness" threshold $\eta_1 > 0$, trust-region expansion constant $\gamma_1 \geq 1$ and contraction constant $\gamma_2 \in (0,1)$, initial sample size n_0, sample size lower bound sequence $\{\lambda_k\}$ such that $k^{(1+\epsilon)} = \mathcal{O}(\lambda_k)$ for some $\epsilon > 0$, initial sample set $\mathcal{Y}_0$, and outer adaptive sampling constant κ_{oas}.

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

2: *Model construction.* Construct the model at $\boldsymbol{X}_k$ by calling Algorithm 2 with the candidate trust-region radius Δ_k and candidate set of design points $\mathcal{Y}_k$,

$$[M_k(\boldsymbol{X}_k + \boldsymbol{s}), \check{\Delta}_k, \check{\mathcal{Y}}_k, \check{N}_k] = \texttt{AdaptiveModelConstruction}(\Delta_k, \mathcal{Y}_k, N_k, \lambda_k).$$

3: *TR subproblem.* Approximate the kth step by minimizing the model in the trust region, $\boldsymbol{S}_k = \operatorname{argmin}_{\|\boldsymbol{s}\| \leq \check{\Delta}_k} M_k(\boldsymbol{X}_k + \boldsymbol{s})$, and set the new candidate point $\tilde{\boldsymbol{X}}_{k+1} = \boldsymbol{X}_k + \boldsymbol{S}_k$.

4: *Evaluate.* Estimate the function at the candidate point using adaptive sampling to obtain $\bar{F}(\tilde{\boldsymbol{X}}_{k+1}, \widetilde{N}_{k+1})$, where

$$\widetilde{N}_{k+1} = \max\left\{\lambda_k, \min\left\{n : \frac{\hat{\sigma}_F(\tilde{\boldsymbol{X}}_{k+1}, n)}{\sqrt{n}} \leq \frac{\kappa_{oas}\check{\Delta}_k^2}{\sqrt{\lambda_k}}\right\}\right\}. \tag{4.1}$$

Update.

5: Compute the success ratio $\hat{\rho}_k$ as

$$\hat{\rho}_k = \frac{\bar{F}(\boldsymbol{X}_k, \check{N}_k) - \bar{F}(\tilde{\boldsymbol{X}}_{k+1}, \widetilde{N}_{k+1})}{M_k(\boldsymbol{X}_k) - M_k(\tilde{\boldsymbol{X}}_{k+1})}.$$

6: **if** $\hat{\rho}_k \geq \eta_1$ **then**

7: $\boldsymbol{X}_{k+1} = \tilde{\boldsymbol{X}}_{k+1}$, $\Delta_{k+1} = \min\{\gamma_1\check{\Delta}_k, \Delta_{\max}\}$, $N_{k+1} = \widetilde{N}_{k+1}$. Update the sample set $\mathcal{Y}_{k+1}$ to include the new iterate.

8: **else**

9: $\boldsymbol{X}_{k+1} = \boldsymbol{X}_k$, $\Delta_{k+1} = \gamma_2\check{\Delta}_k$, $N_{k+1} = \check{N}_k$. Update $\mathcal{Y}_{k+1}$, if needed, to include the rejected candidate point.

10: **end if**

11: **end for**

$\Delta_k^{(j)}$ and center $\boldsymbol{Y}_{k,1}^{(j)} := \boldsymbol{X}_k$ is chosen; (b) Monte Carlo function estimates are obtained at each $\boldsymbol{Y}_{k,i}^{(j)} \in \mathcal{Y}_k^{(j)}$ with sample size $N(\boldsymbol{Y}_{k,i}^{(j)})$, with the center $\boldsymbol{Y}_{k,1}^{(j)} = \boldsymbol{X}_k$ using existing observations if any; and (c) a stochastic model is constructed by interpolating the obtained function estimates.

Sampling at each point in $\mathcal{Y}_k^{(j)}$ is adaptive and continues (steps 4–6) until the estimated standard errors $N(\boldsymbol{Y}_{k,i}^{(j)})^{-1/2}\hat{\sigma}_F(\boldsymbol{Y}_{k,i}^{(j)}, N(\boldsymbol{Y}_{k,i}^{(j)}))$ of the function estimates $\bar{F}(\boldsymbol{Y}_{k,i}^{(j)}, N(\boldsymbol{Y}_{k,i}^{(j)}))$ come within a factor of the squared candidate trust-region radius. If the resulting model $M_k^{(j)}(\boldsymbol{z}), \boldsymbol{z} \in \mathcal{B}(\boldsymbol{X}_k; \Delta_k^{(j)})$ is such that the candidate trust-region radius $\Delta_k^{(j)}$ is too large compared to the norm of the model gradient $\|\nabla M_k^{(j)}(\boldsymbol{X}_k)\|$, that is, if $\Delta_k^{(j)} > \mu\|\nabla M_k^{(j)}(\boldsymbol{X}_k)\|$, then the candidate trust-region radius is shrunk by a factor w and control is returned back to step 3. On the other hand, if the candidate trust-region radius is smaller than the product of μ and the norm of the model gradient, then the resulting stochastic model is accepted but over an updated incumbent trust-region radius given by step 11. Step 11 of Algorithm 2, akin to [25], updates the incumbent trust-region radius $\check{\Delta}$ to the point between $\Delta_k^{(j)}$ and $\beta\|\nabla M_k^{(j)}(\boldsymbol{X}_k)\|$ that is closest to Δ_k.

Step 3 in Algorithm 1 then approximately solves the constrained optimization problem $\boldsymbol{S}_k = \arg\min_{\|\boldsymbol{s}\| \leq \check{\Delta}_k} M_k(\boldsymbol{X}_k + \boldsymbol{s})$ to obtain a candidate point $\tilde{\boldsymbol{X}}_{k+1} = \boldsymbol{X}_k + \boldsymbol{S}_k$ satisfying the κ_{fcd}-Cauchy decrease as defined in Definition 2.5. This will later be specified in Assumption 3.

Algorithm 2 $[M_k(\boldsymbol{X}_k+\boldsymbol{s}), \check{\Delta}_k, \check{\mathcal{Y}}_k, \check{N}_k] =$ `AdaptiveModelConstruction`$(\Delta_k, \mathcal{Y}_k, N_k, \lambda_k)$

Require: *Parameters from ASTRO-DF.* Candidate trust-region radius Δ_k and candidate sample set $\mathcal{Y}_k$ (possibly with cardinality $< p$).
Parameters specific to `AdaptiveModelConstruction`. Trust-region contraction factor $w \in (0,1)$, trust-region and gradient balance constant μ, gradient inflation constant β with $0 < \beta < \mu$, and inner adaptive sampling constant κ_{ias}.

1: Initialize $j = 1$, set $\mathcal{Y}_k^{(j)} = \mathcal{Y}_k$, and set $\boldsymbol{Y}_{k,1} = \boldsymbol{X}_k$, where $\boldsymbol{X}_k$ is the first member of $\mathcal{Y}_k$.
Contraction loop.
2: **repeat**
3: Improve $\mathcal{Y}_k^{(j)} = \{\boldsymbol{Y}_{k,1}^{(j)}, \boldsymbol{Y}_{k,2}^{(j)}, \ldots, \boldsymbol{Y}_{k,p}^{(j)}\}$ by appropriately choosing points that form a Λ-poised set in $\mathcal{B}(\boldsymbol{X}_k; \Delta_k^{(j)})$, where $\Delta_k^{(j)} = \Delta_k w^{j-1}$.
4: **for** $i = 1$ to p **do**
5: Estimate $\bar{F}(\boldsymbol{Y}_{k,i}^{(j)}, N(\boldsymbol{Y}_{k,i}^{(j)}))$, where

$$N\left(\boldsymbol{Y}_{k,i}^{(j)}\right) = \max\left\{\lambda_k, \min\left\{n : \frac{\hat{\sigma}_F\left(\boldsymbol{Y}_{k,i}^{(j)}, n\right)}{\sqrt{n}} \le \frac{\kappa_{ias}(\Delta_k^{(j)})^2}{\sqrt{\lambda_k}}\right\}\right\}. \tag{4.2}$$

6: **end for**
7: Construct the model $M_k^{(j)}(\boldsymbol{X}_k+\boldsymbol{s})$ via interpolation.
8: Set $j = j+1$.
9: **until** $\Delta_k^{(j)} \le \mu\|\nabla M_k^{(j)}(\boldsymbol{X}_k)\|$.
10: Set $M_k(\boldsymbol{X}_k+\boldsymbol{s}) = M_k^{(j)}(\boldsymbol{X}_k+\boldsymbol{s})$ and $\nabla M_k(\boldsymbol{X}_k) = \nabla M_k^{(j)}(\boldsymbol{X}_k)$.
11: **return** $M_k(\boldsymbol{X}_k+\boldsymbol{s})$, $\check{\Delta}_k = \max\{\beta\|\nabla M_k(\boldsymbol{X}_k)\|, \Delta_k^{(j)}\}$, $\check{\mathcal{Y}}_k = \mathcal{Y}_k^{(j)}$, and $\check{N}_k = N(\boldsymbol{Y}_{k,1})$ from step 5.

In preparation for checking if the candidate solution $\tilde{\boldsymbol{X}}_{k+1}$ provides sufficient decrease, step 4 of Algorithm 1 obtains Monte Carlo samples of the objective function at $\tilde{\boldsymbol{X}}_{k+1}$ until the estimated standard error $\widetilde{N}_{k+1}^{-1/2}\hat{\sigma}_F(\tilde{\boldsymbol{X}}_{k+1}, \widetilde{N}_{k+1})$ of $\bar{F}(\tilde{\boldsymbol{X}}_{k+1}, \widetilde{N}_{k+1})$ is within a factor of the squared trust-region radius, subject to the sample size being at least as big as λ_k.

In step 5 of Algorithm 1, the obtained function estimate is used to check if the so-called *success ratio* $\hat{\rho}_k$, that is, the ratio of the predicted to the observed function decrease at the point $\tilde{\boldsymbol{X}}_{k+1}$, exceeds a fixed threshold η_1. The denominator of the success ratio $\hat{\rho}_k$ is calculated by evaluating the constructed model at $\tilde{\boldsymbol{X}}_{k+1}$ (using the analytical form listed in Definition 2.3). If $\hat{\rho}_k$ exceeds the threshold η_1, the candidate $\tilde{\boldsymbol{X}}_{k+1}$ is accepted as the new iterate $\boldsymbol{X}_{k+1}$, the iteration is deemed successful, and the trust region is expanded (step 6). If $\hat{\rho}_k$ falls below the specified threshold η_1, the candidate $\tilde{\boldsymbol{X}}_{k+1}$ is rejected (though it may remain in the sample set), the iteration is deemed unsuccessful, and the trust region is shrunk (step 9). In either case, Δ_{k+1} is set as the incumbent trust-region radius, N_{k+1} is set as the current sample size of $\boldsymbol{X}_{k+1}$, and $\mathcal{Y}_k$ is set as the interpolation set for the next iteration. Note that in the next iteration the sample size of $\boldsymbol{X}_{k+1}$ is subject to change through step 2 again.

We emphasize the following four issues pertaining to Algorithm 1 and the model resulting from the application of Algorithm 2.

(i) The (hypothetical) deterministic model $m_k(\boldsymbol{X}_k)$ constructed from true function observations on the poised set $\check{\mathcal{Y}}_k$ will be $(\kappa_{ef}, \kappa_{eg})$-fully linear on the updated trust region $\mathcal{B}(\boldsymbol{X}_k; \check{\Delta}_k)$. The model $m_k(\boldsymbol{X}_k)$ is, of course, unavailable since true function evaluations are unavailable.

(ii) The trust region resulting from the application of Algorithm 2 has a radius that is at least β times the model gradient norm $\|\nabla M_k(\boldsymbol{X}_k)\|$.

(iii) The structure of adaptive sampling in step 5 of Algorithm 2 is identical to that appearing for estimation in step 4 of Algorithm 1. The adaptive sampling step

simply involves sampling until the estimated standard error of the function estimate comes within a factor of the squared incumbent trust-region radius. As our convergence proofs will reveal, balancing the estimated standard error to any lower power of the incumbent trust-region radius will threaten the consistency of ASTRO-DF's iterates.

(iv) The sequence $\{\lambda_k\}$ appearing as the first argument of the "max" function in the expression for the adaptive sample size (in step 4 of Algorithm 1 and step 5 of Algorithm 2) is standard for all adaptive sampling contexts (see, e.g., [21, 32]) and is intended to nullify the possibility of the sample-size sequence not tending to infinity due to mischance. However, this lower bound on the sample size does not explicitly participate in the asymptotic limit. Specifically, from (4.1), $\widetilde{N}_{k+1} \geq \lambda_k$ and $\widetilde{N}_{k+1}^{-1/2}\hat{\sigma}_F(\boldsymbol{X}_k, \widetilde{N}_{k+1}) \leq \lambda_k^{-1/2}\kappa_{oas}\check{\Delta}_k^2$. In other words,

$$\widetilde{N}_{k+1} = \max\left\{\lambda_k, \lambda_k \frac{\hat{\sigma}_F^2(\tilde{\boldsymbol{X}}_{k+1}, \widetilde{N}_{k+1})}{\kappa_{oas}^2 \check{\Delta}_k^4}\right\}.$$

Since $\mathbb{P}\{\lim_{k\to\infty} \hat{\sigma}_F^2(\boldsymbol{X}_k, N_k) = \sigma > 0\} = 1$ by [21], and $\mathbb{P}\{\lim_{k\to\infty} \Delta_k = 0\} = 1$ (a result we will prove in section 5.1), we observe that the probability of the first argument in the expression for the adaptive sample size being binding decays to zero as $k \to \infty$.

(v) As can be seen from the algorithm listings, the quality of the constructed model in ASTRO-DF is ensured at every iteration; as will be evident, such stringency eases the theoretical analysis. This is in contrast to many modern implementations of derivative-free trust-region algorithms where model quality is checked and improved (if necessary) through a "criticality step" that is triggered only when the norm of the model gradient falls below a threshold. This selective model quality assurance makes the algorithm more numerically efficient at the price of a more complicated trust-region management and convergence analysis (see acceptable iterations and model improving iterations in [26, Page 185]). We believe that incorporating a similar criticality step in ASTRO-DF will lighten the computational burden during implementation.
We also note here that towards improving implementation without affecting asymptotics, a more stringent condition for contracting the trust-region radius could be introduced within steps 6–9 of Algorithm 1. This is consistent with standard trust-region implementations [22, Page 116] that usually use three levels (*very successful*, *successful*, and *unsuccessful*) to describe an iteration, depending on whether the trust-region radius is expanded, remains unchanged, or is contracted.

In summary, at the beginning of each iteration k, Algorithm 1 starts with $\boldsymbol{X}_k$, Δ_k, N_k, and $\mathcal{Y}_k$ as inputs. In step 2, Δ_k and $\mathcal{Y}_k$ are updated to $\check{\Delta}_k$ and $\check{\mathcal{Y}}_k$ and a new $\check{N}_k$ for $\boldsymbol{X}_k$ is obtained. A candidate point $\tilde{\boldsymbol{X}}_{k+1}$ is then found in step 3 and its sample size $\widetilde{N}_{k+1}$ is found in step 4. At the end of iteration k, Algorithm 1 chooses the next solution $\boldsymbol{X}_{k+1}$ and its sample size N_{k+1} by choosing between the incumbent pair $(\boldsymbol{X}_k, \check{N}_k)$ and the candidate pair $(\tilde{\boldsymbol{X}}_{k+1}, \widetilde{N}_{k+1})$. Lastly Δ_{k+1} and $\mathcal{Y}_{k+1}$ are updated from $\check{\Delta}_k$ and $\check{\mathcal{Y}}_k$ in preparation to start the next iteration.

5. Convergence analysis of ASTRO-DF. As noted in section 4, the convergence behavior of ASTRO-DF depends crucially on the behavior of three error terms

expressed through the following decomposition:

$$\begin{aligned}\left|\bar{F}\left(\tilde{\boldsymbol{X}}_{k+1},\widetilde{N}_{k+1}\right)-M_k\left(\tilde{\boldsymbol{X}}_{k+1}\right)\right| \leq & \left|\bar{F}\left(\tilde{\boldsymbol{X}}_{k+1},\widetilde{N}_{k+1}\right)-f\left(\tilde{\boldsymbol{X}}_{k+1}\right)\right| \\ & +\left|f\left(\tilde{\boldsymbol{X}}_{k+1}\right)-m_k\left(\tilde{\boldsymbol{X}}_{k+1}\right)\right| \\ & +\left|m_k\left(\tilde{\boldsymbol{X}}_{k+1}\right)-M_k\left(\tilde{\boldsymbol{X}}_{k+1}\right)\right|. \end{aligned} \tag{5.1}$$

The three terms appearing on the right-hand side of (5.1) can be interpreted, respectively, as follows: (i) the *stochastic sampling error* $|\bar{F}(\tilde{\boldsymbol{X}}_{k+1},\widetilde{N}_{k+1})-f(\tilde{\boldsymbol{X}}_{k+1})|$ arising due to the fact that function evaluations are estimated using Monte Carlo, (ii) the *deterministic model error* $|f(\tilde{\boldsymbol{X}}_{k+1})-m_k(\tilde{\boldsymbol{X}}_{k+1})|$ arising due to the choice of local model, and (iii) the *stochastic interpolation error* $|m_k(\tilde{\boldsymbol{X}}_{k+1})-M_k(\tilde{\boldsymbol{X}}_{k+1})|$ arising due to the fact that model prediction at unobserved points is a combination of the model bias and the error in (i). The analysis in the deterministic context involves only the error in (ii). Accordingly, driving the errors in (i) and (iii) to zero sufficiently fast, while ensuring the full linearity of the unknown deterministic model guarantees almost sure convergence.

Driving the errors in (i) and (iii) to zero sufficiently fast is accomplished by forcing the sample sizes to increase across iterations at a sufficiently fast rate, by keeping the estimated standard error of all function estimates in lock-step with the square of the trust-region radius. The trust-region radius is also kept in lock-step with the model gradient through Algorithm 2. Such a deliberate lock-step between the model error, trust-region radius, and the model gradient is aimed at efficiency without sacrificing consistency.

In what follows, we provide a formal proof of the almost sure convergence of ASTRO-DF's iterates to first-order critical points of the function f. As will become evident, our proof assumes that the constructed models are either stochastic linear interpolation (augmented with a quadratic term) or stochastic quadratic interpolation models, with extensions to other stochastic polynomial interpolation models following in a straightforward fashion. We first list three additional assumptions related to the nature of the proposed algorithm that are assumed to hold along with those listed in section 2.3 for the ensuing results.

Assumption 3. The minimizer obtained in the trust-region subproblem (step 3 of Algorithm 1) satisfies a κ_{fcd}-Cauchy decrease with $\kappa_{fcd}>0$, that is,

$$M_k\left(\boldsymbol{X}_k\right)-M_k\left(\tilde{\boldsymbol{X}}_{k+1}\right) \geq \frac{\kappa_{fcd}}{2}\left\|\nabla M_k\left(\boldsymbol{X}_k\right)\right\| \min\left\{\frac{\left\|\nabla M_k\left(\boldsymbol{X}_k\right)\right\|}{\left\|\nabla^2 M_k\left(\boldsymbol{X}_k\right)\right\|}, \check{\Delta}_k\right\}.$$

Assumption 4. There exists a positive constant κ_{bhm} such that, for every iteration k and every sample path ω, the model Hessian norm is bounded above by κ_{bhm}, that is, $\|\nabla^2 M_k(\boldsymbol{X}_k(\omega))\| \leq \kappa_{bhm}$.

Assumption 5. The "lower-bound sequence" $\{\lambda_k\}$ is chosen to satisfy $k^{(1+\epsilon)} = \mathcal{O}(\lambda_k)$ for some $\epsilon>0$.

Assumption 3 is usually easily ensured through an appropriate choice in the trust-region subproblem (step 3 of Algorithm 1). For example, Assumption 3 is satisfied if the optimization resulting from step 3 of Algorithm 1 yields a solution that is at least as good as the Cauchy step $t_C = \operatorname{argmin}_{\alpha\in[0,\Delta_k]} M_k(\boldsymbol{X}_k - \alpha\nabla M_k(\boldsymbol{X}_k))$ (see section 10.1 in [26] for additional details). Likewise, Assumption 4 can be enforced

through a check and an explicit readjustment of the model parameters ($\hat{\boldsymbol{\alpha}}$ in Definition 2.3). Assumption 5 imposes a weak minimum increase on the sample sizes for estimation and model construction operations within ASTRO-DF (see item (iv) in section 4).

Remark 3. It is our view that the minimum rate of increase on the lower bound sequence $\{\lambda_k\}$ can be reduced to a logarithmic increase, that is, $\log k^{1+\epsilon} = \mathcal{O}(\lambda_k)$, instead of what has been assumed in Assumption 5. Using the notation of Theorem 2.7, this will require a large-deviation-type bound on the tail probability $\mathbb{P}\{|\bar{X}_{N(\lambda)}| > t\}$ after assuming the existence of the moment-generating function of X_i's.

5.1. Main results. We are now ready to establish our *strong consistency* result, that is, the almost sure convergence of the iterates generated by ASTRO-DF to a first-order critical point. We start with Lemma 5.1, which asserts that ASTRO-DF's sample paths are bounded below with probability one. Lemma 5.1 is then used to prove Lemma 5.2 and Theorem 5.3, which, respectively, show that iterations whose trust-region radii are smaller than a certain factor of the model gradient are eventually successful with probability one, and that ASTRO-DF's trust-region radii converge to zero with probability one. Lemma 5.2 and Theorem 5.3 together establish a global almost sure lim-inf convergence result in Theorem 5.5, which in turn leads to a global almost sure lim convergence result in Theorem 5.6.

Lemma 5.1 first establishes that the sequence of function estimates at the iterates generated by ASTRO-DF has to remain bounded from below almost surely.

LEMMA 5.1. *Let Assumptions* 1, 2, *and* 5 *hold. Then*
(a) $\mathbb{P}\{\liminf_{k\to\infty} \bar{F}(\boldsymbol{X}_k, \check{N}_k) = -\infty\} = 0$, *and*
(b) $\mathbb{P}\{|\bar{F}(\boldsymbol{X}_k, \check{N}_k) - f(\boldsymbol{X}_k)| \geq c\check{\Delta}_k^s \text{ i.o.}\} = 0$ *for* $s = 0, 1, 2$ *and any* $c > 0$.
(c) $\mathbb{P}\{|\bar{F}(\boldsymbol{X}_{k+1}, N_{k+1}) - f(\boldsymbol{X}_{k+1})| \geq c\check{\Delta}_k^s \text{ i.o.}\} = 0$ *for* $s = 0, 1, 2$ *and any* $c > 0$.

Proof. For part (a) we know from Assumption 1 that f is bounded from below. Hence we can write for any $c > 0$,

$$\mathbb{P}\left\{\liminf_{k\to\infty} \bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right) = -\infty\right\} \leq \mathbb{P}\left\{\left|\bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right) - f\left(\boldsymbol{X}_k\right)\right| \geq c \text{ i.o.}\right\}. \tag{5.2}$$

However by the law of total probability,

$$\begin{aligned}\mathbb{P}\left\{\left|\bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right) - f\left(\boldsymbol{X}_k\right)\right| \geq c\right\} &= \mathbb{E}\left[\mathbb{P}\left\{\left|\bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right) - f\left(\boldsymbol{X}_k\right)\right| \geq c \,|\, \mathcal{F}_k\right\}\right] \\ &\leq \mathbb{E}[c^{-2}\mathbb{E}[\left(\bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right) - f\left(\boldsymbol{X}_k\right)\right)^2 \,|\, \mathcal{F}_k]], \end{aligned} \tag{5.3}$$

where the last inequality follows from Chebyshev's inequality [12]. Now invoke Theorem 2.7 along with Assumption 2 and the sample-size expression in (4.1) to notice that

$$\mathbb{E}[\left(\bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right) - f\left(\boldsymbol{X}_k\right)\right)^2 \,|\, \mathcal{F}_k] \sim \kappa_{ias}^2 \check{\Delta}_k^4 \lambda_k^{-1} \tag{5.4}$$

as $\lambda_k \to \infty$; that is, for large enough k, we can write for any $\delta > 0$ that

$$\begin{aligned}\mathbb{E}[\left(\bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right) - f\left(\boldsymbol{X}_k\right)\right)^2 \,|\, \mathcal{F}_k] &\leq (1+\delta)\kappa_{ias}^2 \check{\Delta}_k^4 \lambda_k^{-1} \\ &\leq (1+\delta)\kappa_{ias}^2 \Delta_{\max}^4 \lambda_k^{-1}. \end{aligned} \tag{5.5}$$

Now use (5.3) and (5.5) to write

$$\begin{aligned}\mathbb{P}\left\{\left|\bar{F}\left(\boldsymbol{X}_k,\check{N}_k\right)-f\left(\boldsymbol{X}_k\right)\right|\geq c\right\} &\leq \mathbb{E}[c^{-2}\mathbb{E}[\left(\bar{F}\left(\boldsymbol{X}_k,\check{N}_k\right)-f\left(\boldsymbol{X}_k\right)\right)^2\mid\mathcal{F}_k]]\\ &\leq \mathbb{E}[c^{-2}(1+\delta)\kappa_{ias}^2\Delta_{\max}^4\lambda_k^{-1}]\\ &= c^{-2}(1+\delta)\kappa_{ias}^2\Delta_{\max}^4\lambda_k^{-1}. \end{aligned}\tag{5.6}$$

The right-hand side of (5.6) is summable since $k^{(1+\epsilon)}=\mathcal{O}(\lambda_k)$ for some $\epsilon>0$; we can thus invoke the first Borel–Cantelli lemma (Lemma 2.11) and conclude that the right-hand side of (5.2) is zero. This proves part (a) and the $s=0$ case in part (b).

To prove the $s=2$ case in part (b), similar to (5.6), we write

$$\begin{aligned}\mathbb{P}\left\{\left|\bar{F}\left(\boldsymbol{X}_k,\check{N}_k\right)-f\left(\boldsymbol{X}_k\right)\right|\geq c\check{\Delta}_k^2\right\} &\leq \mathbb{E}[c^{-2}\check{\Delta}_k^{-4}\mathbb{E}[\left(\bar{F}\left(\boldsymbol{X}_k,\check{N}_k\right)-f\left(\boldsymbol{X}_k\right)\right)^2\mid\mathcal{F}_k]]\\ &\leq \mathbb{E}[c^{-2}\check{\Delta}_k^{-4}(1+\delta)\kappa_{ias}^2\check{\Delta}_k^4\lambda_k^{-1}]\\ &= c^{-2}(1+\delta)\kappa_{ias}^2\lambda_k^{-1}, \end{aligned}\tag{5.7}$$

which is again summable and hence allows the invocation of the first Borel–Cantelli lemma (Lemma 2.11). This proves the $s=2$ case in part (b).

The $s=1$ case in part (b) follows along lines identical to the $s=2$ case. Finally part (c) follows along similar lines after noting the relationship between the sample size N_{k+1} and the trust-region radius $\check{\Delta}_k$, for both successful k where $(\boldsymbol{X}_{k+1},N_{k+1})=(\tilde{\boldsymbol{X}}_{k+1},\tilde{N}_{k+1})$ as well as unsuccessful k where $(\boldsymbol{X}_{k+1},N_{k+1})=(\boldsymbol{X}_k,\check{N}_k)$. ∎

We next prove a lemma that asserts that iterations where the trust-region radius (upon exiting Algorithm 2) drops below a certain factor of the model gradient are eventually successful with probability one.

Lemma 5.2. *Let Assumptions* 1, 2, 3, 4, *and* 5 *hold. Define the set*

$$\mathcal{V}:=\left\{\omega:\exists\ \textit{inf. subseq. }\{k_j\}\ \textit{s.t.}\left(\frac{\check{\Delta}_{k_j}(\omega)}{\|\nabla M_{k_j}(\boldsymbol{X}_{k_j}(\omega))\|}\leq\frac{1}{2}\frac{\eta'}{\kappa_{ef}}\right)\bigcap\left(\hat{\rho}_{k_j}(\omega)<\eta_1\right)\right\},\tag{5.8}$$

where $\eta'=6^{-1}(1-\eta_1)\kappa_{fcd}\min\{(\mu\kappa_{bhm})^{-1},1\}$. *Then,* $\mathbb{P}\{\mathcal{V}\}=0$.

Proof. For every $\omega\in\mathcal{V}$, Assumption 3 on Cauchy decrease in the minimization problem implies

$$\begin{aligned} &M_k\left(\boldsymbol{X}_k(\omega)\right)-M_k(\tilde{\boldsymbol{X}}_{k+1}(\omega))\\ &\quad\geq\frac{\kappa_{fcd}}{2}\left\|\nabla M_k\left(\boldsymbol{X}_k(\omega)\right)\right\|\min\left\{\frac{\left\|\nabla M_k\left(\boldsymbol{X}_k(\omega)\right)\right\|}{\left\|\nabla^2 M_k\left(\boldsymbol{X}_k(\omega)\right)\right\|},\check{\Delta}_k(\omega)\right\}\\ &\quad\geq\frac{\kappa_{fcd}}{2}\left\|\nabla M_k\left(\boldsymbol{X}_k(\omega)\right)\right\|\min\left\{\frac{\check{\Delta}_k(\omega)}{\mu\kappa_{bhm}},\check{\Delta}_k(\omega)\right\}\\ &\quad=\frac{\kappa_{fcd}}{2}\min\{(\mu\kappa_{bhm})^{-1},1\}\check{\Delta}_k(\omega)\left\|\nabla M_k\left(\boldsymbol{X}_k(\omega)\right)\right\|, \end{aligned}\tag{5.9}$$

where the second inequality follows from both $\check{\Delta}_k(\omega)\leq\mu\|\nabla M_k(\boldsymbol{X}_k(\omega))\|$, ensured by Algorithm 2, and Assumption 4. Now, recall that

$$\hat{\rho}_k(\omega):=\frac{\bar{F}\left(\boldsymbol{X}_k(\omega),\check{N}_k\right)-\bar{F}(\tilde{\boldsymbol{X}}_{k+1}(\omega),\widetilde{N}_{k+1})}{M_k(\boldsymbol{X}_k(\omega))-M_k(\tilde{\boldsymbol{X}}_{k+1}(\omega))},$$

and that $\bar{F}(\boldsymbol{X}_k(\omega), \check{N}_k) = M_k(\boldsymbol{X}_k(\omega))$. Now, since $\hat{\rho}_{k_j}(\omega) < \eta_1$, $|1 - \hat{\rho}_{k_j}(\omega)| > 1 - \eta_1$ and we can write

$$|1 - \hat{\rho}_{k_j}(\omega)| = \frac{|\bar{F}(\tilde{\boldsymbol{X}}_{k_j+1}(\omega), \widetilde{N}_{k_j+1}(\omega)) - M_{k_j}(\tilde{\boldsymbol{X}}_{k_j+1}(\omega))|}{M_{k_j}(\boldsymbol{X}_{k_j}(\omega)) - M_{k_j}(\tilde{\boldsymbol{X}}_{k_j+1}(\omega))} > 1 - \eta_1,$$

and hence

$$|\bar{F}(\tilde{\boldsymbol{X}}_{k_j+1}(\omega), \widetilde{N}_{k_j+1}(\omega)) - M_{k_j}(\tilde{\boldsymbol{X}}_{k_j+1}(\omega))| > 3\eta' \check{\Delta}_{k_j}(\omega) \|\nabla M_{k_j}(\boldsymbol{X}_{k_j}(\omega))\|.$$

Making use of the decomposition in (5.1), we obtain

$$\begin{aligned}\mathcal{V} &= \left\{\omega : \exists \text{ inf. subseq. } \{k_j\} \text{ s.t. } \left(\frac{\check{\Delta}_{k_j}(\omega)}{\|\nabla M_{k_j}(\boldsymbol{X}_{k_j}(\omega))\|} \le \frac{1}{2}\frac{\eta'}{\kappa_{ef}}\right)\right. \\ &\qquad \left. \bigcap \left(|1 - \hat{\rho}_{k_j}(\omega)| > 1 - \eta_1\right) \right\} \\ &\subseteq \mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{V}_3,\end{aligned}$$

where

$$\begin{aligned}\mathcal{V}_1 &:= \left\{\omega : \exists \text{ inf. subseq. } \{k_j\} \text{ s.t. } Err_{1,k_j}(\omega) > \eta' \check{\Delta}_{k_j}(\omega) \left\|\nabla M_{k_j}\left(\boldsymbol{X}_{k_j}(\omega)\right)\right\|\right\}, \\ \mathcal{V}_2 &:= \left\{\omega : \exists \text{ inf. subseq. } \{k_j\} \text{ s.t. } \left(\frac{\check{\Delta}_{k_j}(\omega)}{\|\nabla M_{k_j}(\boldsymbol{X}_{k_j}(\omega))\|} \le \frac{1}{2}\frac{\eta'}{\kappa_{ef}}\right)\right. \\ &\qquad \left. \bigcap \left(Err_{2,k_j}(\omega) > \eta' \check{\Delta}_{k_j}(\omega) \left\|\nabla M_{k_j}\left(\boldsymbol{X}_{k_j}(\omega)\right)\right\|\right) \right\}, \\ \mathcal{V}_3 &:= \left\{\omega : \exists \text{ inf. subseq. } \{k_j\} \text{ s.t. } Err_{3,k_j}(\omega) > \eta' \check{\Delta}_{k_j}(\omega) \left\|\nabla M_{k_j}\left(\boldsymbol{X}_{k_j}(\omega)\right)\right\|\right\},\end{aligned}$$

and

$$\begin{aligned}Err_{1,k}(\omega) &:= |\bar{F}(\tilde{\boldsymbol{X}}_{k+1}(\omega), \widetilde{N}_{k+1}(\omega)) - f(\tilde{\boldsymbol{X}}_{k+1}(\omega))|, \\ Err_{2,k}(\omega) &:= |f(\tilde{\boldsymbol{X}}_{k+1}(\omega)) - m_k(\tilde{\boldsymbol{X}}_{k+1}(\omega))|, \\ Err_{3,k}(\omega) &:= |m_k(\tilde{\boldsymbol{X}}_{k+1}(\omega)) - M_k(\tilde{\boldsymbol{X}}_{k+1}(\omega))|.\end{aligned}$$

The rest of the proof will demonstrate that each of the sets $\mathcal{V}_1$, $\mathcal{V}_2$, and $\mathcal{V}_3$ has measure zero, thereby establishing the result.

Let us show that the set $\mathcal{V}_1$ has measure zero. (In what follows, we are careful to suppress the argument ω when making probabilistic arguments leading to the invocation of the first Borel–Cantelli lemma.) Use $\check{\Delta}_{k_j}(\omega) \le \mu \|\nabla M_{k_j}(\boldsymbol{X}_{k_j}(\omega))\|$ in the expression for $\mathcal{V}_1$, and then part (b) of Lemma 5.1 with $s = 2$ to see that

$$\mathbb{P}\{\mathcal{V}_1\} \le \mathbb{P}\{|\bar{F}(\tilde{\boldsymbol{X}}_{k+1}, \widetilde{N}_{k+1}) - f(\tilde{\boldsymbol{X}}_{k+1})| \ge \eta' \mu^{-1} \check{\Delta}_k^2 \text{ i.o.}\} = 0. \tag{5.10}$$

Let us next show that the set $\mathcal{V}_3$ has measure zero. Notice, using part (i) of Lemma 2.9, and relabeling $\boldsymbol{X}_k$ to $\boldsymbol{Y}_{k,1}$ for readability, that

$$\begin{aligned}\mathbb{P}\{Err_{3,k} > \eta'\mu^{-1}\check{\Delta}_k^2\} &\leq \mathbb{P}\left\{\max_{\substack{\boldsymbol{Y}_{k,i}\in\mathcal{Y}_k,\\ i=1,2,\ldots,p}} \left|\bar{F}\left(\boldsymbol{Y}_{k,i}, N\left(\boldsymbol{Y}_{k,i}\right)\right) - f\left(\boldsymbol{Y}_{k,i}\right)\right| > \frac{\eta'\mu^{-1}\check{\Delta}_k^2}{p\Lambda}\right\} \\ &\leq \sum_{i=1}^{p} \mathbb{P}\left\{\left|\bar{F}\left(\boldsymbol{Y}_{k,i}, N\left(\boldsymbol{Y}_{k,i}\right)\right) - f\left(\boldsymbol{Y}_{k,i}\right)\right| > \frac{\eta'\mu^{-1}\check{\Delta}_k^2}{p^2\Lambda}\right\} \\ &= \sum_{i=1}^{p} \mathbb{E}\left[\mathbb{P}\left\{\left|\bar{F}\left(\boldsymbol{Y}_{k,i}, N\left(\boldsymbol{Y}_{k,i}\right)\right) - f\left(\boldsymbol{Y}_{k,i}\right)\right| > \frac{\eta'\mu^{-1}\check{\Delta}_k^2}{p^2\Lambda} \mid \mathcal{F}_k\right\}\right]. \end{aligned} \tag{5.11}$$

Now using (5.11) and part (b) of Lemma 5.1 with $s = 2$, we can then say for large enough k and some $\delta > 0$ that

$$\mathbb{P}\left\{Err_{3,k} > \eta'\mu^{-1}\check{\Delta}_k^2\right\} \leq \frac{p^5\Lambda^2\mu^2}{\lambda_k(\eta')^2}(1+\delta)\kappa_{oas}^2. \tag{5.12}$$

Since λ_k is chosen so that $k^{1+\epsilon} = \mathcal{O}(\lambda_k)$ for some $\epsilon > 0$, we see that (5.12) implies, from the first Borel–Cantelli lemma (Lemma 2.11), that

$$\mathbb{P}\left\{Err_{3,k} > \eta'\mu^{-1}\check{\Delta}_k^2 \text{ i.o.}\right\} = 0. \tag{5.13}$$

Noting the definition of the set $\mathcal{V}_3$, and since $\check{\Delta}_k(\omega) \leq \mu\|\nabla M_k(\boldsymbol{X}_k(\omega))\|$ is ensured by Algorithm 2, we then see that

$$\mathbb{P}\{\mathcal{V}_3\} \leq \mathbb{P}\left\{Err_{3,k} > \eta'\mu^{-1}\check{\Delta}_k^2 \text{ i.o.}\right\} = 0. \tag{5.14}$$

Let's next show that the set $\mathcal{V}_2$ has measure zero. At the end of step 9 of Algorithm 2 let $m_k^{(j)}(\boldsymbol{z})$ be the interpolation model of f constructed on the Λ-poised set $\mathcal{Y}_k^{(j)}$ (we cannot construct $m_k(\cdot)$ explicitly because the true function values are unknown). Then $m_k^{(j)}(\boldsymbol{z})$ is a $(\kappa_{ef}, \kappa_{eg})$-fully linear model of f on $\mathcal{B}(\boldsymbol{X}_k(\omega); \Delta_k^{(j)}(\omega))$, and since $\check{\Delta}_k(\omega) \geq \Delta_k(\omega)w^{j-1}$, by Lemma 10.25 in [26, Page 200] we have that $m_k(\cdot)$ is a $(\kappa_{ef}, \kappa_{eg})$-fully linear model of f on $\mathcal{B}(\boldsymbol{X}_k(\omega); \check{\Delta}_k(\omega))$, implying that

$$|f(\tilde{\boldsymbol{X}}_{k+1}(\omega)) - m_k(\tilde{\boldsymbol{X}}_{k+1}(\omega))| \leq \kappa_{ef}\check{\Delta}_k^2(\omega).$$

Therefore, whenever $\|\nabla M_{k_j}(\boldsymbol{X}_{k_j}(\omega))\|^{-1}\check{\Delta}_{k_j}(\omega) \leq (2\kappa_{ef})^{-1}\eta'$ as stipulated by the first condition in the definition of the set $\mathcal{V}_2$, we see that

$$\begin{aligned} Err_{2,k_j}(\omega) := |f(\tilde{\boldsymbol{X}}_{k_j+1}(\omega)) - m_{k_j}(\tilde{\boldsymbol{X}}_{k_j+1}(\omega))| &\leq \frac{1}{2}\eta'\check{\Delta}_{k_j}(\omega)\|\nabla M_{k_j}\left(\boldsymbol{X}_{k_j}(\omega)\right)\| \\ &< \eta'\check{\Delta}_{k_j}(\omega)\|\nabla M_{k_j}\left(\boldsymbol{X}_{k_j}(\omega)\right)\|, \end{aligned}$$

implying that $\mathcal{V}_2 = \emptyset$, and hence

$$\mathbb{P}\{\mathcal{V}_2\} = 0. \tag{5.15}$$

From (5.10), (5.14), and (5.15), we see that $\mathbb{P}\{\mathcal{V}\} \leq \mathbb{P}\{\mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{V}_3\} = 0$ and the assertion of the lemma holds. ∎

Lemma 5.2 essentially states that iterations wherein the trust-region radius becomes small compared to the model gradient will eventually be successful with probability one. This result will become important later when we demonstrate that the limit infimum of true gradients at ASTRO-DF's iterates converges to zero with probability one.

Remark 4. Lemma 5.2 does not guarantee that all iterations past a threshold are successful iterations. In other words, ASTRO-DF can generate sample paths that contain an infinite subsequence of unsuccessful iterations. Interestingly, since we know from steps 9 and 11 of Algorithm 2 that the ratio $\check{\Delta}_k \|\nabla M_k(\boldsymbol{X}_k)\|^{-1} \in [\beta, \mu]$, the subsequence defined in $\mathcal{V}$ never materializes if $(2\kappa_{ef})^{-1}\eta' < \beta$, implying that the iterations of ASTRO-DF are eventually successful almost surely. However, the condition $(2\kappa_{ef})^{-1}\eta' < \beta$ cannot be assured in practice since the constant κ_{ef} is unknown.

We next prove a theorem that plays a crucial role in proving the overall convergence of ASTRO-DF iterates. Recall that even in deterministic derivative-free trust-region algorithms, unlike trust-region algorithms where derivative observations are available, the trust-region radius necessarily needs to decay to zero to ensure convergence. Theorem 5.3 states that this is indeed the case for ASTRO-DF. The proof rests on Lemma 5.1 and the assumed sufficient Cauchy decrease guarantee during step 3 of Algorithm 1.

THEOREM 5.3. *Let Assumptions* 1, 2, 3, 4, *and* 5 *hold. Then,*

$$\Delta_k \xrightarrow{w.p.1} 0 \text{ as } k \to \infty.$$

Proof. Define the set

$$\begin{aligned}\mathcal{D}_1 := \{\omega : \exists K_1(\omega) \text{ s.t. } & |\bar{F}(\boldsymbol{X}_k, \check{N}_k) - f(\boldsymbol{X}_k)| \le c'\check{\Delta}_k^2 \\ & \text{and } |\bar{F}(\boldsymbol{X}_{k+1}, N_{k+1}) - f(\boldsymbol{X}_{k+1})| \le c'\check{\Delta}_k^2 \text{ for all } k \ge K_1(\omega)\},\end{aligned} \tag{5.16}$$

where $c' = (3+2\gamma_1^2)^{-1}(2\mu)^{-1}\eta_1\kappa_{fcd}\min\{(\mu\kappa_{bhm})^{-1}, 1\}$. Due to part (b) of Lemma 5.1 and the definition of N_{k+1}, we note that $\mathbb{P}\{\mathcal{D}_1\} = 1$. Additionally, define the set $\mathcal{D}_2 := \{\omega : \liminf_{k\to\infty} \bar{F}(\boldsymbol{X}_k, \check{N}_k) > -\infty\}$. Part (a) of Lemma 5.1 also implies that $\mathbb{P}\{\mathcal{D}_2\} = 1$. Select $\omega \in \mathcal{D}_1 \cap \mathcal{D}_2$ for the ensuing arguments.

Suppose the number of successful iterations in ω is finite; then since unsuccessful iterations are necessarily contracting iterations, the trust-region radius has to tend to zero and the statement of the theorem holds.

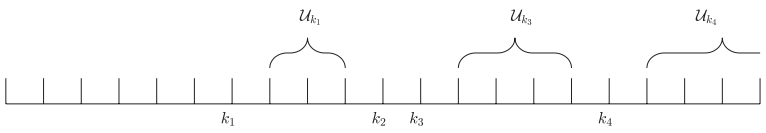

FIG. 1. *The iterations in ASTRO-DF can be divided into blocks of successful and unsuccessful iterations. In the illustration, the successful iterations are* k_1, k_2, k_3, *and* k_4, *and the block of unsuccessful iterations that follows the successful iteration* k_i *is denoted* $\mathcal{U}_{k_i}$ *with* $\mathcal{U}_{k_2} = \emptyset$.

Now suppose that the number of successful iterations in ω is infinite, and let $\{k_j\}$ be the sequence of successful iterations in ω. For convenience of analysis, divide the

iterations in ω into "blocks" of successful and unsuccessful iterations as depicted in Figure 1 and write (after suppressing ω for notational simplicity)

$$\begin{aligned}\bar{F}\left(\boldsymbol{X}_{k_n}, \check{N}_{k_n}\right) &= \bar{F}\left(\boldsymbol{X}_{k_1}, \check{N}_{k_1}\right) + \sum_{k=k_1}^{k_{n-1}} \left(\bar{F}\left(\boldsymbol{X}_{k+1}, \check{N}_{k+1}\right) - \bar{F}\left(\boldsymbol{X}_k, \check{N}_k\right)\right) \\ &= \bar{F}\left(\boldsymbol{X}_{k_1}, \check{N}_{k_1}\right) + \sum_{j=1}^{n-1} \left(\left(A_{k_j} + B_{k_j}\right) + \sum_{i \in \mathcal{U}_{k_j}} \left(A_i + B_i\right) \right), \end{aligned} \tag{5.17}$$

where $\{k_j\}$ are the successful iterations, $\mathcal{U}_{k_j} := \{k_j + 1, \ldots, k_{j+1} - 1\}$ is the "block" of unsuccessful iterations that follows the successful iteration k_j (with the understanding that $\mathcal{U}_{k_j}$ can be the empty set),

$$A_i = \bar{F}(\boldsymbol{X}_{i+1}, \check{N}_{i+1}) - \bar{F}(\boldsymbol{X}_{i+1}, N_{i+1}), \quad \text{and} \quad B_i = \bar{F}(\boldsymbol{X}_{i+1}, N_{i+1}) - \bar{F}(\boldsymbol{X}_i, \check{N}_i).$$

In words, A_i represents the stochastic fluctuation of the function estimate at $\boldsymbol{X}_{i+1}$ before entering and upon exiting `AdaptiveModelConstruction` and B_i represents the change in the function estimate after updating the iterate at the end of Algorithm 1.

Also, henceforth we suppose that the cardinality $|\mathcal{U}_{k_j}| < \infty$, because, if $|\mathcal{U}_{k_j}| = \infty$ for some j, since the unsuccessful iterations are contracting, the statement of the theorem will hold trivially.

We make some observations to be used in analyzing the behavior of the summations appearing in (5.17).

(a) We write

$$\begin{aligned} A_i &\le \left|\bar{F}(\boldsymbol{X}_{i+1}, \check{N}_{i+1}) - f(\boldsymbol{X}_{i+1})\right| + \left|f(\boldsymbol{X}_{i+1}) - \bar{F}\left(\boldsymbol{X}_{i+1}, N_{i+1}\right)\right| \\ &\le c' \check{\Delta}_{i+1}^2 + c' \check{\Delta}_i^2 \\ &\le (\gamma_1^2 + 1) c' \check{\Delta}_i^2, \end{aligned} \tag{5.18}$$

where the first inequality in (5.18) follows from the triangle inequality, the second inequality in (5.18) follows since $\omega \in \mathcal{D}_1$, letting $c = c'$ in part (b) and (c) of Lemma 5.1, and the third inequality in (5.18) follows noting that $\check{\Delta}_{i+1} \le \gamma_1 \check{\Delta}_i$.

(b) For the unsuccessful iteration i, $N_{i+1} = \check{N}_i$ while $\boldsymbol{X}_{i+1} = \boldsymbol{X}_i$. Hence $B_i = 0$.

(c) For the successful iteration k_j, we know by definition that $\hat{\rho}_{k_j} \ge \eta_1$, $\boldsymbol{X}_{k_j+1} = \widetilde{\boldsymbol{X}}_{k_j+1}$, and $N_{k_j+1} = \widetilde{N}_{k_j+1}$. Then by Assumptions 3 and 4 and by the assurance in Algorithm 2 that $\check{\Delta}_{k_j} \le \mu \|\nabla M_{k_j}(\boldsymbol{X}_k)\|$, we have

$$\begin{aligned} B_{k_j} &\le \eta_1 \left(M_{k_j}\left(\boldsymbol{X}_{k_j+1}\right) - M_{k_j}\left(\boldsymbol{X}_{k_j}\right)\right) \\ &\le -\frac{\eta_1}{2} \kappa_{fcd} \left\|\nabla M_{k_j}\left(\boldsymbol{X}_{k_j}\right)\right\| \min\left\{ \frac{\left\|\nabla M_{k_j}\left(\boldsymbol{X}_{k_j}\right)\right\|}{\left\|\nabla^2 M_{k_j}\left(\boldsymbol{X}_{k_j}\right)\right\|}, \check{\Delta}_{k_j} \right\} \\ &\le -(3 + 2\gamma_1^2) c' \check{\Delta}_{k_j}^2. \end{aligned} \tag{5.19}$$

Now, towards analyzing (5.17), we see that there exists a j_0 such that, for $j \ge j_0$,

$$\begin{aligned} A_{k_j} + B_{k_j} + \sum_{i \in \mathcal{U}_{k_j}} (A_i + B_i) &= A_{k_j} + B_{k_j} + \sum_{i \in \mathcal{U}_{k_j}} A_i \\ &\le A_{k_j} + B_{k_j} + \bar{F}\left(\boldsymbol{X}_{(k_j)+1}, \check{N}_{k_{j+1}}\right) \\ &\quad - \bar{F}\left(\boldsymbol{X}_{(k_j)+1}, N_{(k_j)+1}\right) \\ &< (\gamma_1^2 + 1 - (3 + 2\gamma_1^2) + (\gamma_1^2 + 1)) c' \check{\Delta}_{k_j}^2 = -c' \check{\Delta}_{k_j}^2, \end{aligned} \tag{5.20}$$

where the first equality follows from observation (b), the first inequality follows from the definition of A_i and after observing that $\boldsymbol{X}_{(k_j)+1} = \boldsymbol{X}_{(k_j)+2} = \cdots = \boldsymbol{X}_{k_{j+1}}$ and $N_{i+1} = \check{N}_i$ for $i \in \mathcal{U}_{k_j}$, and finally the second inequality follows from observation (a), (5.19), and knowing that $\check{\Delta}_{k_{j+1}} < \check{\Delta}_{(k_j)+1} \leq \gamma_1 \check{\Delta}_{k_j}$ for $i \in \mathcal{U}_{k_j}$. Note that when $k_{j+1} = k_j + 1$, that is, two successful iterations occur consecutively, the set $\mathcal{U}_{k_j}$ is empty and the inequality in (5.20) still holds.

Use (5.20) in (5.17) to see that, for $n \geq j_0$,

$$\begin{aligned} \bar{F}\left(\boldsymbol{X}_{k_n}, \check{N}_{k_n}\right) &\leq \bar{F}\left(\boldsymbol{X}_{k_1}, \check{N}_{k_1}\right) \\ &\quad + \sum_{j=1}^{j_0-1} \left(\left(A_{k_j} + B_{k_j}\right) + \sum_{i \in \mathcal{U}_{k_j}} \left(A_i + B_i\right) \right) - c' \sum_{j=j_0}^{n} \check{\Delta}_{k_j}^2. \end{aligned} \tag{5.21}$$

Thus, since $\omega \in \mathcal{D}_1 \cap \mathcal{D}_2$ and (5.21) holds, we see that $\sum_{j=j_0}^{n} \check{\Delta}_{k_j}^2 < \infty$ and therefore $\check{\Delta}_{k_j} \to 0$ as $j \to \infty$. Furthermore, from $\check{\Delta}_i \leq \check{\Delta}_{k_j}$ for $i \in \mathcal{U}_{k_j}$, we note that $\{\check{\Delta}_k\} \to 0$. And, since $\Delta_{k+1} \leq \gamma_1 \check{\Delta}_k$, we conclude that the assertion of the theorem holds. ∎

Relying on Theorem 5.3, we show through Lemma 5.4 that the model gradient converges to the true gradient almost surely when the model is obtained through stochastic linear interpolation (with or without a quadratic term) or stochastic quadratic interpolation. Lemma 5.4 does not imply that the true gradient itself converges to zero—a fact that will be established subsequently. Again, implicit in the proof of Theorem 5.3 is the requirement that Algorithm 2 terminates in finite time with probability one, a fact that we establish through Lemma C.1 in the appendix.

LEMMA 5.4. *Let Assumptions* 1, 2, 3, 4, *and* 5 *hold. Suppose the model $M_k(\cdot)$ is obtained through stochastic linear or stochastic quadratic interpolation. Then*

$$\|\nabla M_k\left(\boldsymbol{X}_k\right) - \nabla f\left(\boldsymbol{X}_k\right)\| \xrightarrow{w.p.1} 0 \text{ as } k \to \infty.$$

Proof. In step 3 of Algorithm 2, $\Delta_k^{(j)}$ denotes the trust-region radius over which the model is constructed. Note that due to step 11 of Algorithm 2, $\Delta_k^{(j)}$ may or may not equal the ending trust-region radius $\check{\Delta}_k$ upon completion of k iterations of ASTRO-DF. Then, from part (ii) of Lemma 2.9 we know that except for a set of sample paths of measure zero,

$$\|\nabla M_k\left(\boldsymbol{X}_k\right) - \nabla f\left(\boldsymbol{X}_k\right)\| \leq \kappa_{eg1} \Delta_k^{(j)} + \kappa_{eg2} \frac{\sqrt{\sum_{i=2}^{p} \left(E_{k,i}^{(j)} - E_{k,1}^{(j)}\right)^2}}{\Delta_k^{(j)}},$$

where $E_{k,i}^{(j)} = \bar{F}(\boldsymbol{Y}_{k,i}^{(j)}, N(\boldsymbol{Y}_{k,i}^{(j)})) - f(\boldsymbol{Y}_{k,i}^{(j)})$ for $i = 1, \ldots, p$ denotes the error due to sampling at point $\boldsymbol{Y}_{k,i}^{(j)}$ after the jth iteration of the contraction loop (recall that $\boldsymbol{Y}_{k,1}^{(j)} = \boldsymbol{X}_k$).

From Theorem 5.3, $\Delta_k \xrightarrow{\text{w.p.1}} 0$, and hence $\Delta_k^{(j)} \xrightarrow{\text{w.p.1}} 0$ as $k \to \infty$. Also,

$$\sqrt{\sum_{i=2}^{p} \left(E_{k,i}^{(j)} - E_{k,1}^{(j)}\right)^2} \leq \sum_{i=2}^{p} \sqrt{\left(E_{k,i}^{(j)} - E_{k,1}^{(j)}\right)^2} = \sum_{i=2}^{p} \left|E_{k,i}^{(j)} - E_{k,1}^{(j)}\right|.$$

Considering these two observations, it suffices to show that as $k \to \infty$,

$$\left(\Delta_k^{(j)}\right)^{-1} \sum_{i=2}^{p} \left|E_{k,i}^{(j)} - E_{k,1}^{(j)}\right| \xrightarrow{\text{w.p.1}} 0. \tag{5.22}$$

Towards this, we write for any $c > 0$, large enough k, some $\delta > 0$, and by Boole's inequality (Lemma 2.10),

$$\begin{aligned}
\mathbb{P}\left\{\frac{\sum_{i=2}^{p}\left|E_{k,i}^{(j)}-E_{k,1}^{(j)}\right|}{\Delta_k^{(j)}} \geq c\right\} &\leq \sum_{i=2}^{p}\mathbb{E}\left[\mathbb{P}\left\{\left|E_{k,i}^{(j)}-E_{k,1}^{(j)}\right| \geq \frac{c\Delta_k^{(j)}}{p-1} \mid \mathcal{F}_k\right\}\right] \\
&\leq \sum_{i=2}^{p}\left(\mathbb{E}\left[\mathbb{P}\left\{\left|E_{k,i}^{(j)}\right| \geq \frac{c\Delta_k^{(j)}}{2(p-1)} \mid \mathcal{F}_k\right\}\right]\right. \\
&\qquad \left.+\mathbb{E}\left[\mathbb{P}\left\{\left|E_{k,1}^{(j)}\right| \geq \frac{c\Delta_k^{(j)}}{2(p-1)} \mid \mathcal{F}_k\right\}\right]\right) \\
&\leq (p-1)\mathbb{E}\left[2\frac{4(p-1)^2(1+\delta)\kappa_{ias}^2}{\left(c\Delta_k^{(j)}\right)^2}(\Delta_k^{(j)})^4\lambda_k^{-1}|\mathcal{F}_k\right] \\
&\leq 8(p-1)^3c^{-2}(1+\delta)\kappa_{ias}^2\Delta_{\max}^2\lambda_k^{-1}, \qquad (5.23)
\end{aligned}$$

where the second inequality in (5.23) follows from the application of Boole's inequality (see Lemma 2.10), and the penultimate inequality in (5.23) follows from arguments identical to those leading to (5.6) in the proof of Lemma 5.1 after using the adaptive sample-size expression in (4.2), the final inequality in (5.23) follows since $\check{\Delta}_k \geq \Delta_k^{(j)}$, and c is arbitrary. Since the right-hand side of (5.23) is summable, we can invoke Lemma 2.11 to conclude that (5.22) holds. ∎

The almost sure liminf convergence now follows.

THEOREM 5.5. *Let Assumptions* 1, 2, 3, 4, *and* 5 *hold. Then*

$$\liminf \|\nabla f(\boldsymbol{X}_k)\| \xrightarrow{w.p.1} 0 \text{ as } k \to \infty.$$

Proof. For the purpose of arriving at a contradiction, suppose that

$$\mathcal{D}_g = \{\omega : \exists\ \kappa_{lbg}(\omega), K_g(\omega) > 0 \text{ s.t. } \|\nabla M_k(\boldsymbol{X}_k)\| \geq \kappa_{lbg}(\omega)\ \forall k > K_g(\omega)\}$$

and that the set $\mathcal{D}_g$ has positive measure.

Due to the assumptions of the theorem, we see that Lemma 5.2 and Theorem 5.3 hold. Hence, we can find a set $\mathcal{D}_d$ of sample paths such that $\mathbb{P}\{\mathcal{D}_d\} = 1$, and such that for each $\omega \in \mathcal{D}_d$, the sequence $\{\check{\Delta}_k(\omega)\} \to 0$ and $\omega \in \mathcal{V}^c$, where the set $\mathcal{V}$ is given in (5.8). Recall that $\check{\Delta}_k$ is the trust-region radius upon exiting Algorithm 2.

Choose $\omega \in \mathcal{D}_g \cap \mathcal{D}_d$. Then by Lemma 5.2, we see that either $\|\nabla M_k(\boldsymbol{X}_k(\omega))\| < 2\kappa_{ef}(\eta')^{-1}\check{\Delta}_k(\omega)$ (with η' given in the postulate of Lemma 5.2) or $\hat{\rho}_k(\omega) \geq \eta_1$ for large enough k. But $\|\nabla M_k(\boldsymbol{X}_k(\omega))\| < 2\kappa_{ef}(\eta')^{-1}\check{\Delta}_k(\omega)$ cannot be true for large enough k since $\{\check{\Delta}_k(\omega)\} \to 0$. Therefore, for the chosen $\omega \in \mathcal{D}_g \cap \mathcal{D}_d$, it must be true that $\hat{\rho}_k(\omega) \geq \eta_1$ for large enough k. In other words, the iterations in sample path $\omega \in \mathcal{D}_g \cap \mathcal{D}_d$ are eventually successful.

Now let $K_s(\omega) > 0$ be such that $K_s(\omega) - 1$ is the last unsuccessful iteration in $\omega \in \mathcal{D}_g \cap \mathcal{D}_d$, that is, k is a successful iteration if $k \geq K_s(\omega)$. Consider $k \geq \max\{K_g(\omega), K_s(\omega)\} + 1$ and the following two (mutually exclusive and collectively exhaustive) cases when Algorithm 2 starts.

(i) $\Delta_k(\omega) \geq \mu\|\nabla M_k(\boldsymbol{X}_k(\omega))\|$: the inner loop of Algorithm 2 is executed more than once, implying that

$$\check{\Delta}_k(\omega) \geq \beta\|\nabla M_k(\boldsymbol{X}_k(\omega))\| \geq \beta\kappa_{lbg}(\omega).$$

(ii) $\Delta_k(\omega) < \mu\|\nabla M_k(\boldsymbol{X}_k(\omega))\|$: the inner loop of Algorithm 2 is executed only once, meaning that the trust-region radius attempts to expand from the previous iteration, that is,

$$\check{\Delta}_k(\omega) = \Delta_k(\omega) = \min(\gamma_1 \check{\Delta}_{k-1}(\omega), \Delta_{\max}).$$

We see from (i) and (ii) that $\check{\Delta}_k(\omega) \geq \min\{\beta\kappa_{lbg}(\omega), \Delta_{\max\{K_g(\omega),K_s(\omega)\}}\}$, thus contradicting the observation $\{\check{\Delta}_k(\omega)\} \to 0$. We conclude, hence, that $\mathbb{P}\{\mathcal{D}_g\} = 0$ and that

$$\liminf_{k\to\infty} \|\nabla M_k(\boldsymbol{X}_k)\| = 0 \tag{5.24}$$

almost surely. This, along with Lemma 5.4, implies $\liminf_{k\to\infty} \|\nabla f(\boldsymbol{X}_k)\| = 0$ almost surely. ∎

An important observation from the Algorithms 1 and 2 is that the sequence $\{\bar{F}(\boldsymbol{X}_k, N_k)\}$ of function estimates is not necessarily monotone decreasing. When iteration k is unsuccessful, that is, $\boldsymbol{X}_k = \boldsymbol{X}_{k+1}$, it is possible that $\bar{F}(\boldsymbol{X}_k, N_k) < \bar{F}(\boldsymbol{X}_{k+1}, N_{k+1})$; when iteration k is successful, it must be true that $\bar{F}(\boldsymbol{X}_k, \check{N}_k) > \bar{F}(\boldsymbol{X}_{k+1}, N_{k+1})$ but it is still possible that $\bar{F}(\boldsymbol{X}_k, N_k) < \bar{F}(\boldsymbol{X}_{k+1}, N_{k+1})$ because $\bar{F}(\boldsymbol{X}_k, N_k) \neq \bar{F}(\boldsymbol{X}_k, \check{N}_k)$. This fact somewhat complicates the next result on almost sure convergence.

THEOREM 5.6. *Let Assumptions* 1, 2, 3, 4, *and* 5 *hold. Then*

$$\|\nabla f(\boldsymbol{X}_k)\| \xrightarrow{w.p.1} 0 \text{ as } k \to \infty.$$

Proof. We first observe that Lemmas 5.1 and 5.2, Theorem 5.3, Lemma 5.4, and Theorem 5.5 all hold with probability one. We can hence assume without loss of generality that in a set $\mathcal{D}$ of measure one, the assertions in each of Lemmas 5.1 and 5.2, Theorem 5.3, Lemma 5.4, and Theorem 5.5 hold sample-pathwise for all $\omega \in \mathcal{D}$.

Towards a contradiction, suppose that the assertion of the theorem does not hold on a set $\tilde{\mathcal{D}} \subset \mathcal{D}$ of positive measure, and choose $\omega \in \tilde{\mathcal{D}}$. The chosen sample path ω may either have an infinite number of successful iterations or a finite number of successful iterations. In what follows we suppress ω from the notation for ease of exposition.

Let us first consider the case where the chosen sample path ω has a finite number of successful iterations. Then there exists $K_0 < \infty$ such that sequence $\{\boldsymbol{X}_k\}$ of iterates remains fixed for $k \geq K_0$, that is, $\boldsymbol{X}_k = \boldsymbol{X}_{k+1} = \boldsymbol{X}_{k+2} = \cdots$ for $k \geq K_0$. Also there exists a subsequence $\{k_i\}$ and $\epsilon > 0$ such that $\|\nabla f(\boldsymbol{X}_{k_i})\| \geq \epsilon$. Then, we see that

$$0 < \epsilon \leq \|\nabla f(\boldsymbol{X}_{k_i})\| \leq \|\nabla f(\boldsymbol{X}_{k_i}) - \nabla M_{k_i}(\boldsymbol{X}_{k_i})\| + \|\nabla M_{k_i}(\boldsymbol{X}_{k_i})\|. \tag{5.25}$$

However, since we know from Lemma 5.4 that $\|\nabla f(\boldsymbol{X}_{k_i}) - \nabla M_{k_i}(\boldsymbol{X}_{k_i})\| \to 0$, we see that for large enough i, $\|\nabla M_{k_i}(\boldsymbol{X}_{k_i})\| \geq \epsilon/2$. Next, we observe from Lemma 5.2 that for every subsequence $\{k_j\}$ of iterations in the sample path ω, either the iteration is successful or $\check{\Delta}_{k_j} > \frac{1}{2}\frac{\eta'}{\kappa_{ef}}\|\nabla M_{k_j}(\boldsymbol{X}_{k_j})\|$. Then since our successful iterations were finite, it must be true for large enough i that $\check{\Delta}_{k_i} > \frac{1}{2}\frac{\eta'}{\kappa_{ef}}\|\nabla M_{k_i}(\boldsymbol{X}_{k_i})\| \geq \frac{1}{4}\frac{\eta'}{\kappa_{ef}}\epsilon$. This result contradicts Theorem 5.3, allowing us to conclude that the assertion of the theorem holds.

Now suppose the sample path ω has an infinite number of successful iterations. We can find a subsequence of iterations $\{t_i\}$ satisfying $\|\nabla f(\boldsymbol{X}_{t_i})\| > 3\epsilon$ for some

$\epsilon > 0$. Due to Theorem 5.5, corresponding to each element t_i, there exists $\ell_i = \ell(t_i)$, that is the first iteration after t_i with $\|\nabla f(\boldsymbol{X}_{\ell_i})\| < 2\epsilon$. Furthermore, because there exists an infinite number of successful iterations, and since the iterates do not change after an unsuccessful iteration, the iteration t_i and the iteration ℓ_i can be chosen to be *successful* iterations. Defining $\mathcal{K}_i := \{k : t_i \le k < \ell_i\}$, we then see that

$$\|\nabla f\left(\boldsymbol{X}_{\ell_i}\right)\| < 2\epsilon, \quad \|\nabla f\left(\boldsymbol{X}_{t_i}\right)\| > 3\epsilon, \quad \|\nabla f\left(\boldsymbol{X}_k\right)\| \ge 2\epsilon \quad \text{if } k \in \mathcal{K}_i, \tag{5.26}$$

and that iteration t_i is successful, iteration $k \in \mathcal{K}_i \setminus \{t_i\}$ may or may not be successful, and iteration ℓ_i is successful.

Adopting the notation from the proof of Theorem 5.3 and denoting the first success of the set $\mathcal{K}_i$ as the $j(i)$th overall success, we note that k_j, $j \in \mathcal{J}_i := \{j(i), j(i)+1, \ldots, j(i)+n(i)-1\}$ are the successful iterations in the set $\mathcal{K}_i$ with $n(i)$ total successes. Observe that $k_{j(i)} = t_i$ and $k_{j(i)+n(i)-1} < \ell_i$.

The following further observations hold for $\omega \in \tilde{\mathcal{D}}$, and $i \ge i_1$, where $i_1 := i_1(\omega) < \infty$ is some threshold.

(a) For $k \in \mathcal{K}_i$, by Lemma 5.4 and since we have argued that $\|\nabla f(\boldsymbol{X}_k)\| \ge 2\epsilon$, we see that $\|\nabla M_k(\boldsymbol{X}_k)\| \ge \epsilon$.

(b) For $k \in \mathcal{K}_i$, by Theorem 5.3, $\check{\Delta}_k \le \kappa_{bhm}^{-1}\epsilon$.

(c) For $k \in \mathcal{K}_i$, by parts (b) and (c) of Lemma 5.1 with $c = \frac{1}{4(2+\gamma_1)}\eta_1\epsilon\kappa_{fcd}$ and $s = 1$,

$$|\bar{F}(\boldsymbol{X}_{k+1}, N_{k+1}) - f(\boldsymbol{X}_{k+1})| \le \frac{1}{4(2+\gamma_1)}\eta_1\epsilon\kappa_{fcd}\check{\Delta}_k$$

and

$$|\bar{F}(\boldsymbol{X}_k, \check{N}_k) - f(\boldsymbol{X}_k)| \le \frac{1}{4(2+\gamma_1)}\eta_1\epsilon\kappa_{fcd}\check{\Delta}_k.$$

(d) For any successful iteration k_j with $j \in \mathcal{J}_i$,

$$\begin{aligned} B_{k_j} := \bar{F}\left(\boldsymbol{X}_{k_j+1}, N_{k_j+1}\right) - \bar{F}\left(\boldsymbol{X}_{k_j}, \check{N}_{k_j}\right) &\le -\eta_1\frac{\kappa_{fcd}}{2}\epsilon\min\left\{\frac{\epsilon}{\kappa_{bhm}}, \check{\Delta}_{k_j}\right\} \\ &= -\frac{1}{2}\eta_1\epsilon\kappa_{fcd}\check{\Delta}_{k_j}, \end{aligned} \tag{5.27}$$

where we have followed the notation in (5.17) for B_{k_j}, the second inequality in (5.19), and our observations in (a) and (b).

Identical to the arguments leading to (5.17) (see Figure 1), we write

$$\begin{aligned} \bar{F}\left(\boldsymbol{X}_{\ell_i}, \check{N}_{\ell_i}\right) &= \bar{F}\left(\boldsymbol{X}_{t_i}, \check{N}_{t_i}\right) + \sum_{j\in\mathcal{J}_i}\left(\left(A_{k_j}+B_{k_j}\right) + \sum_{u\in\mathcal{U}_{k_j}}\left(A_u+B_u\right)\right) \\ &= \bar{F}\left(\boldsymbol{X}_{t_i}, \check{N}_{t_i}\right) \\ &\quad + \sum_{j\in\mathcal{J}_i} A_{k_j} + B_{k_j} + \bar{F}\left(\boldsymbol{X}_{(k_j)+1}, \check{N}_{k_j+1}\right) - \bar{F}\left(\boldsymbol{X}_{(k_j)+1}, N_{(k_j)+1}\right), \end{aligned} \tag{5.28}$$

where, following the notation in (5.17), $A_i = \bar{F}(\boldsymbol{X}_{i+1}, \check{N}_{i+1}) - \bar{F}(\boldsymbol{X}_{i+1}, N_{i+1})$ and $B_i = \bar{F}(\boldsymbol{X}_{i+1}, N_{i+1}) - \bar{F}(\boldsymbol{X}_i, \check{N}_i)$. Applying the observation in (c) and noting $\check{\Delta}_{k+1} \le \gamma_1\check{\Delta}_k$, we see that

$$|\bar{F}\left(\boldsymbol{X}_{(k_j)+1}, \check{N}_{k_j+1}\right) - \bar{F}\left(\boldsymbol{X}_{(k_j)+1}, N_{(k_j)+1}\right)| \le \frac{\gamma_1+1}{4(2+\gamma_1)}\eta_1\epsilon\kappa_{fcd}\check{\Delta}_{k_j}. \tag{5.29}$$

Also, again using the observation in (c), we see that

$$|A_{k_j}| \leq \frac{1}{4(2+\gamma_1)}\eta_1\epsilon\kappa_{fcd}\check{\Delta}_{k_j}. \tag{5.30}$$

Now use (5.27), (5.30), and (5.29) in (5.28) to get

$$\bar{F}\left(\boldsymbol{X}_{\ell_i}, \check{N}_{\ell_i}\right) - \bar{F}\left(\boldsymbol{X}_{t_i}, \check{N}_{t_i}\right) \leq -\frac{1}{4}\eta_1\epsilon\kappa_{fcd}\sum_{j\in\mathcal{J}_i}\check{\Delta}_{k_j}. \tag{5.31}$$

Furthermore, since $\boldsymbol{X}_{j+1} = \boldsymbol{X}_j$ when j is unsuccessful, we have

$$\begin{aligned}\|\boldsymbol{X}_{\ell_i} - \boldsymbol{X}_{t_i}\| &\leq \sum_{j=t_i}^{\ell_i-1}\|\boldsymbol{X}_{j+1} - \boldsymbol{X}_j\| \leq \sum_{j\in\mathcal{J}_i}\check{\Delta}_{k_j} \\ &\leq \frac{4}{\eta_1\epsilon\kappa_{fcd}}\left(\bar{F}\left(\boldsymbol{X}_{t_i}, \check{N}_{t_i}\right) - \bar{F}(\boldsymbol{X}_{\ell_i}, \check{N}_{\ell_i})\right), \end{aligned} \tag{5.32}$$

where the second inequality in (5.32) follows from (5.31).

Now suppose $\{k_j\}$ is the sequence of all successful iterations in $\omega \in \tilde{\mathcal{D}}$. From our arguments leading to (5.31), we know that for large enough j,

$$A_{k_j} + B_{k_j} + \sum_{u\in\mathcal{U}_{k_j}} A_u < 0;$$

hence the sequence $\{\bar{F}(\boldsymbol{X}_{k_j}, \check{N}_{k_j})\}$ of function estimates at the successful iterations is decreasing for large enough j. And, we know from part (a) of Lemma 5.1 that the sequence $\{\bar{F}(\boldsymbol{X}_{k_j}, \check{N}_{k_j})\}$ of function estimates is bounded below. The right-hand side of (5.32) thus tends to zero as $i \to \infty$, leading us to conclude that

$$\|\boldsymbol{X}_{\ell_i} - \boldsymbol{X}_{t_i}\| \to 0 \text{ as } i \to \infty. \tag{5.33}$$

The conclusion in (5.33), along with Assumption 1 on Lipschitz continuity of the gradients, implies that

$$\|\nabla f\left(\boldsymbol{X}_{\ell_i}\right) - \nabla f\left(\boldsymbol{X}_{t_i}\right)\| \to 0 \text{ as } i \to \infty. \tag{5.34}$$

However, (5.34) contradicts (5.26), leading us to conclude that the assertion of the theorem holds. ∎

6. Further remarks and discussion. Over the last decade or so, derivative-free trust-region algorithms have deservedly enjoyed great attention and success in the deterministic optimization context. Analogous algorithms for the now widely prevalent and important Monte Carlo stochastic optimization context, where only stochastic function oracles are available, are poorly studied. This paper develops adaptive sampling derivative-free trust-region algorithms (called ASTRO-DF) for solving stochastic optimization problems. The key idea within ASTRO-DF is to endow a derivative-free trust-region algorithm with an adaptive sampling strategy for function estimation. The extent of such sampling at a visited point depends on the estimated proximity of the point to a solution, calculated by balancing the estimated standard error of the function estimate with a certain power of the incumbent trust-region

radius. So, just as one might expect of efficient algorithms, Monte Carlo sampling in ASTRO-DF tends to be low during the early iterations compared to later iterations, when the visited points are more likely to be closer to a first-order critical point. More importantly, however, the schedule of sampling is not predetermined (as in most stochastic approximation and sample-average approximation algorithms) but instead adapts to the prevailing algorithm trajectory and the needed precision of the function estimates.

We show that the norm of the gradient at the sequence of iterates returned by ASTRO-DF converges to zero with probability one. While the proofs are detailed, convergence follows from two key features of ASTRO-DF: (i) the stochastic interpolation models are constructed across iterates in such a way that the error in the stochastic interpolation model is guaranteed to remain in lock-step with (a certain power of) the trust-region radius, and (ii) the optimization within the trust-region step is performed in such a way as to guarantee Cauchy reduction and then the objective function is evaluated at the resulting candidate point. Remarkably, features (i) and (ii) together ensure that the sequence of trust-region radii necessarily need to converge to zero with probability one, and that the model gradient, the true gradient, and the trust-region radius all have to remain in lock-step, thus guaranteeing that the sequence of true gradient norms at the incumbent solutions converges to zero with probability one. The key driver for efficiency is adaptive sampling, making all sample sizes within ASTRO-DF stopping times that are explicitly dependent on algorithm trajectory.

Four other points are worthy of mention.

(i) Our proofs demonstrate global convergence to first-order critical points. Corresponding proofs of convergence to a second-order critical point can be obtained in an identical fashion by driving some measure of second-order stationarity to zero instead of the model gradient.

(ii) The adaptive sampling ideas and the ensuing proofs we have presented in this paper are for the specific case of stochastic interpolation models. It seems to us, however, that the methods of proof presented in this paper can be co-opted (with care) into other potentially more powerful model construction ideas such as regression [13] and kriging [3, 59].

(iii) We have presented no proof that ASTRO-DF's iterates achieve the Monte Carlo *canonical rate* [4], which in a sense represents the fastest achievable convergence rate. Formally, this means demonstrating that $\sqrt{W_k}\,\nabla f(\mathbf{X}_k) = \mathcal{O}_p(1)$, where W_k is the total simulation effort after k iterations and $\mathcal{O}_p(\cdot)$ connotes stochastic bounding [55]. Demonstrating that ASTRO-DF's iterates achieve the canonical rate will rely on rate results for derivative-free trust-region algorithms in the deterministic context, some of which are only now appearing [30].

(iv) The asymptotic sampling rate within ASTRO-DF is approximately $\mathcal{O}(\Delta_k^{-4})$, where Δ_k is the incumbent trust-region radius (see (5.4) in the proof of Lemma 5.1). This sampling stipulation is comparable to that prescribed in two other prominent recent studies [20, 38]. The $\mathcal{O}(\Delta_k^{-4})$ sampling appears to be the minimum needed to guarantee convergence in derivative-free trust-region methods *without assumptions on the tail behavior of the error driving simulation observations.* A question of interest is whether the $\mathcal{O}(\Delta_k^{-4})$ sampling stipulation can be relaxed by assuming that the simulation error is light-tailed, that is, the errors have a well-defined moment-generating function.

Appendix A. Proof of part (i) of Lemma 2.9. We know that for all $\boldsymbol{z} \in \mathcal{B}(\boldsymbol{Y}_1; \Delta)$,

$$m(\boldsymbol{z}) = \sum_{i=1}^{p} \ell_i(\boldsymbol{z}) f(\boldsymbol{Y}_i), \qquad M(\boldsymbol{z}) = \sum_{i=1}^{p} \ell_i(\boldsymbol{z}) \bar{F}(\boldsymbol{Y}_i, n(\boldsymbol{Y}_i)),$$

where $\ell_i(\boldsymbol{z})$ are the Lagrange polynomials associated with the set $\mathcal{Y}$. Since $\mathcal{Y}$ is Λ-poised in $\mathcal{B}(\boldsymbol{Y}_1; \Delta)$, we know (see Chapter 3 in [26]) that

$$\Lambda \geq \Lambda_\ell = \max_{i=1,2,\ldots,p} \max_{\boldsymbol{z} \in \mathcal{B}(\boldsymbol{Y}_1; \Delta)} |\ell_i(\boldsymbol{z})|. \tag{A.1}$$

Now write, for $\boldsymbol{z} \in \mathcal{B}(\boldsymbol{Y}_1; \Delta)$,

$$\begin{aligned} |M(\boldsymbol{z}) - m(\boldsymbol{z})| &= \left| \sum_{i=1}^{p} \ell_i(\boldsymbol{z}) \left(\bar{F}(\boldsymbol{Y}_i, n(\boldsymbol{Y}_i)) - f(\boldsymbol{Y}_i) \right) \right| \\ &\leq p\Lambda_\ell \max_{i \in \{1,2,\ldots,p\}} \left| \bar{F}(\boldsymbol{Y}_i, n(\boldsymbol{Y}_i)) - f(\boldsymbol{Y}_i) \right| \\ &\leq p\Lambda \max_{i \in \{1,2,\ldots,p\}} \left| \bar{F}(\boldsymbol{Y}_i, n(\boldsymbol{Y}_i)) - f(\boldsymbol{Y}_i) \right|, \end{aligned}$$

where the last inequality follows from (A.1).

Appendix B. Proof of part (ii) of Lemma 2.9. If the model $M(\cdot)$ is a stochastic linear or a stochastic quadratic interpolation model, we see that, for $i = 1, 2, 3, \ldots, p$,

$$\begin{aligned} &(\boldsymbol{Y}_i - \boldsymbol{Y}_1)^T \nabla M(\boldsymbol{Y}_1) \\ &\quad = M(\boldsymbol{Y}_i) - M(\boldsymbol{Y}_1) + \frac{1}{2}(\boldsymbol{Y}_i - \boldsymbol{Y}_1)^T \nabla^2 M(\boldsymbol{Y}_1)(\boldsymbol{Y}_i - \boldsymbol{Y}_1) \\ &\quad = (f(\boldsymbol{Y}_i) - f(\boldsymbol{Y}_1)) + (E_i - E_1) + \frac{1}{2}(\boldsymbol{Y}_i - \boldsymbol{Y}_1)^T \nabla^2 M(\boldsymbol{Y}_1)(\boldsymbol{Y}_i - \boldsymbol{Y}_1). \end{aligned} \tag{B.1}$$

Now retrace the steps of the proof on pages 26 and 27 of [26], while carrying the additional term $E_i - E_1$ appearing on the right-hand side of (B.1) to obtain

$$\Delta \|\nabla M(z) - \nabla f(z)\| \leq \kappa_1 \Delta^2 + \kappa_2 \sqrt{\sum_{i=2}^{p} (E_i - E_1)^2}, \tag{B.2}$$

where κ_1 and κ_2 are determined by the scaled *geometry matrix*, comprised of, for $i = 2, \ldots, p$, $\Delta^{-1}(\boldsymbol{Y}_i - \boldsymbol{Y}_1)$, and the Lipschitz constant of the gradient ν_{gL}.

Appendix C. Model construction algorithm termination. We demonstrate through the following result that the model construction algorithm (Algorithm 2) terminates with probability one whenever the incumbent solution $\boldsymbol{X}_k$ is not a first-order critical point.

LEMMA C.1. *Define*

$$\mathcal{W} := \{\omega : \exists k \text{ such that } \mathbb{I}_{NC_k}(\omega) = 1,\ \mathbb{I}_{NT_k}(\omega) = 1\},$$

where NC_k is the event that $\nabla f(\boldsymbol{X}_k) \neq 0$ and NT_k is the event that Algorithm 2 does not terminate. (The set $\mathcal{W}$ contains sample paths that each have at least one iteration k where the solution $\boldsymbol{X}_k$ is noncritical but Algorithm 2 does not terminate.) Then $\mathbb{P}\{\mathcal{W}_k\} = 0$.

Proof. Write $\mathcal{W} = \bigcup_{i=1}^{\infty} \mathcal{W}_k$, where

$$\mathcal{W}_k := \{\omega : \mathbb{I}_{NC_k}(\omega) = 1,\ \mathbb{I}_{NT_k}(\omega) = 1\}.$$

We will now demonstrate that each set $\mathcal{W}_k$ is of measure zero.

First, we notice that the contraction loop (steps 3–9) in Algorithm 2 is executed only once if $\mu\|\nabla M_k(\boldsymbol{X}_k)\| \geq \Delta_k$, in which case Algorithm 2 terminates trivially. Hence, the set $\mathcal{W}_k$ contains only sample paths ω such that $\mu\|\nabla M_k(\boldsymbol{X}_k)\| < \Delta_k$ and Algorithm 2 does not terminate, that is, the contraction loop in steps 3–9 of Algorithm 2 during iteration k is infinite.

Let $\nabla M_k^{(j)}(\boldsymbol{X}_k) \,|\, \mathcal{F}_k$ denote the model gradient during the jth iteration of the contraction loop, given the filtration until that iteration. Then, due to the previous argument, $\mathbb{I}_{NT_k}\mu\|\nabla M_k^{(j)}(\boldsymbol{X}_k)|\mathcal{F}_k\| < \Delta_k^{(j)}$ for all $j \geq 1$. Recall from step 3 of Algorithm 2 that $\Delta_k^{(j)} = \Delta_k w^{j-1}$. Then, since $w < 1$, and since we have supposed that there are an infinite number of contractions,

$$\mathbb{I}_{NT_k}\|\nabla M_k^{(j)}\left(\boldsymbol{X}_k\right) \,|\, \mathcal{F}_k\| \xrightarrow{\text{w.p.1}} 0 \text{ as } j \to \infty. \tag{C.1}$$

Due to the sampling rule in (4.2) and by Theorem 2.7, we know that $N(\boldsymbol{Y}_{k,i}^{(j)})|\mathcal{F}_k \xrightarrow{\text{w.p.1}} \infty$ as $j \to \infty$. Now, if

$$E_{k,i}^{(j)} \,|\, \mathcal{F}_k = \bar{F}(\boldsymbol{Y}_{k,i}^{(j)}, N(\boldsymbol{Y}_{k,i}^{(j)})) - f(\boldsymbol{Y}_{k,i}^{(j)}) \,|\, \mathcal{F}_k,$$

then we can write for large enough j and some $\delta > 0$,

$$\mathbb{P}\left\{\mathbb{I}_{NT_k}\frac{\sum_{i=2}^{p}|E_{k,i}^{(j)} - E_{k,1}^{(j)}|}{\Delta_k^{(j)}} \geq c \,|\, \mathcal{F}_k\right\} \leq 8(p-1)^3 c^{-2}(1+\delta)\kappa_{ias}^2 \Delta_{\max}^2 \lambda_k^{-1} w^{2j-2}, \tag{C.2}$$

which follows from arguments identical to (5.23) in the proof of Lemma 5.4. Since the right-hand side of (C.2) is summable in j, we conclude by the first Borel–Cantelli lemma (Lemma 2.11) that as $j \to \infty$,

$$\mathbb{I}_{NT_k}\left(\frac{\sum_{i=2}^{p}|E_{k,i}^{(j)} - E_{k,1}^{(j)}|}{\Delta_k^{(j)}} \,|\, \mathcal{F}_k\right) \xrightarrow{\text{w.p.1}} 0.$$

This implies, from Lemma 2.9 and since Algorithm 2 maintains fully linear models, that as $j \to \infty$,

$$\begin{aligned}
&\mathbb{I}_{NT_k}\left\|\nabla f\left(\boldsymbol{X}_k\right) - \nabla M_k^{(j)}\left(\boldsymbol{X}_k\right)|\mathcal{F}_k\right\| \\
&\quad \leq \kappa_{eg1}\mathbb{I}_{NT_k}\Delta_k^{(j)} + \kappa_{eg2}\mathbb{I}_{NT_k}\left(\frac{\sum_{i=2}^{p}|E_{k,i}^{(j)} - E_{k,1}^{(j)}|}{\Delta_k^{(j)}} \,|\, \mathcal{F}_k\right) \\
&\quad \xrightarrow{\text{w.p.1}} 0.
\end{aligned} \tag{C.3}$$

Use (C.3) and (C.1) to conclude that $\mathcal{W}_k$ has to be a measure zero set. ∎

Acknowledgments. The authors owe a debt of gratitude to the two referees and the Associate Editor for their excellent suggestions and detailed technical feedback throughout the editorial process.

REFERENCES

[1] O. Alagoz, A. J. Schaefer, and M. S. Roberts, *Optimization in organ allocation*, in Handbook of Optimization in Medicine, P. M. Pardalos and E. Romeijn, eds., Kluwer Academic Publishers, Norwell, MA, 2009, pp. 1–28.

[2] B. D. Amos, D. R. Easterling, L. T. Watson, W. I. Thacker, B. S. Castle, and M. W. Trosset, *Algorithm XXX: QNSTOP—Quasi Newton Algorithm for Stochastic Optimization*, preprint, https://vtechworks.lib.vt.edu/bitstream/handle/10919/49672/qnTOMS14.pdf, 2014.

[3] B. Ankenman, B. L. Nelson, and J. Staum, *Stochastic kriging for simulation metamodeling*, Oper. Res., 58 (2010), pp. 371–382.

[4] S. Asmussen and P. W. Glynn, *Stochastic Simulation: Algorithms and Analysis*, Stoch. Model. Appl. Probab. 57, Springer, New York, 2007.

[5] A. S. Bandeira, K. Scheinberg, and L. N. Vicente, *Convergence of trust-region methods based on probabilistic models*, SIAM J. Optim., 24 (2014), pp. 1238–1264.

[6] F. Bastin, C. Cirillo, and P. L. Toint, *Convergence theory for nonconvex stochastic programming with an application to mixed logit*, Math. Program., 108 (2006), pp. 207–234.

[7] F. Bastin, C. Cirillo, and Ph. L. Toint, *An adaptive Monte Carlo algorithm for computing mixed logit estimators*, Comput. Manag. Sci., 3 (2006), pp. 55–79.

[8] G. Bayraksan and D. P. Morton, *Assessing solution quality in stochastic programs*, Math. Program., 108 (2007), pp. 495–514.

[9] G. Bayraksan and D. P. Morton, *A sequential sampling procedure for stochastic programming*, Oper. Res., 59 (2011), pp. 898–913.

[10] G. Bayraksan and P. Pierre-Louis, *Fixed-width sequential stopping rules for a class of stochastic programs*, SIAM J. Optim., 22 (2012), pp. 1518–1548.

[11] M. S. Bazaara, H. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*, John Wiley, New York, 2006.

[12] P. Billingsley, *Probability and Measure*, John Wiley, New York, 1995.

[13] S. C. Billups, J. Larson, and P. Graf, *Derivative-free optimization of expensive functions with computational error using weighted regression*, SIAM J. Optim., 23 (2013), pp. 27–53.

[14] J. Blum, *Approximation methods which converge with probability one*, Ann. Math. Stat., 25 (1954), pp. 382–386.

[15] V. S. Borkar, *Stochastic Approximation: A Dynamical Systems Viewpoint*, Cambridge University Press, Cambridge, 2008.

[16] L. Bottou, F. Curtis, and J. Nocedal, *Optimization methods for large-scale machine learning*, SIAM Rev., 60 (2018), pp. 223–311.

[17] M. Broadie, D. M. Cicek, and A. Zeevi, *An adaptive multidimensional version of the Kiefer–Wolfowitz stochastic approximation algorithm*, in Proceedings of the 2009 Winter Simulation Conference, M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, eds., IEEE Press, Piscataway, NJ, 2009, pp. 601–612.

[18] M. Broadie, D. M. Cicek, and A. Zeevi, *General bounds and finite-time improvement for the Kiefer–Wolfowitz stochastic approximation algorithm*, Oper. Res., 59 (2011), pp. 1211–1224.

[19] K. Chang, L. J. Hong, and H. Wan, *Stochastic trust-region response-surface method (STRONG)—A new response-surface framework for simulation optimization*, INFORMS J. Comput., 25 (2013), pp. 230–243.

[20] R. Chen, M. Menickelly, and K. Scheinberg, *Stochastic optimization using a trust-region method and random models*, Math. Program., 169 (2018), pp. 447–487.

[21] T. S. Chow and H. Robbins, *On the asymptotic theory of fixed-width sequential confidence intervals for the mean*, Ann. Math. Statist., 36 (1965), pp. 457–462.

[22] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust-Region Methods*, MOS-SIAM Ser. Optim., SIAM, Philadelphia, PA, 2000.

[23] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Geometry of interpolation sets in derivative free optimization*, Math. Program., 111 (2008), pp. 141–172.

[24] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Geometry of sample sets in derivative-free optimization: Polynomial regression and undetermined interpolation*, IMA J. Numer. Anal., 28 (2008), pp. 721–748.

[25] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Global convergence of general derivative-free trust-region algorithms to first-and second-order critical points*, SIAM J. Optim., 20 (2009), pp. 387–415.

[26] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*, MOS-SIAM Ser. Optim. 8, SIAM, Philadelphia, PA, 2009.

[27] G. Deng and M. C. Ferris, *Adaptation of the UOBYQA Algorithm for Noisy Functions*, in Proceedings of the 38th Conference on Winter Simulation, Monterey, CA, 2006, pp. 312–319.
[28] G. Deng and M. C. Ferris, *Variable-number sample-path optimization*, Math. Program., 117 (2009), pp. 81–109.
[29] R. Durrett, *Probability: Theory and Examples*, Cambridge University Press, New York, 2010.
[30] R. Garmanjani, D. Júdice, and L. N. Vicente, *Trust-Region Methods Without Using Derivatives: Worst Case Complexity and the Non-Smooth Case*, preprint, https://www.mat.uc.pt/~lnv/papers/str.pdf, 2015.
[31] M. Ghosh and N. Mukhopadhyay, *Sequential point estimation of the mean when the distribution is unspecified*, Comm. Statist. Theory Methods, 8 (1979), pp. 637–652.
[32] M. Ghosh, N. Mukhopadhyay, and P. K. Sen, *Sequential Estimation*, Wiley Ser. Probab. Stat., 1997.
[33] Y. T. Hou, Y. Shi, and H. D. Sherali, *Applied Optimization Methods for Wireless Networks*, Cambridge University Press, Cambridge, 2014.
[34] J. Kiefer and J. Wolfowitz, *Stochastic estimation of the maximum of a regression function*, Ann. Math. Stat., 23 (1952), pp. 462–466.
[35] S. Kim, R. Pasupathy, and S. G. Henderson, *A Guide to sample average approximation*, in Handbook of Simulation Optimization, Internat. Ser. Oper. Res. Management Sci. 216, Fu M. (ed.), Springer, New York, NY, 2015.
[36] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, Springer, New York, 2003.
[37] J. Larson and S. C. Billups, *Stochastic derivative-free optimization using a trust-region framework*, Comput. Optim. Appl., 64 (2016), pp. 619–645.
[38] J. M. Larson, *Derivative-Free Optimization of Noisy Functions*, Ph.D. thesis, Department of Applied Mathematics, University of Colorado, Denver, CO, 2012.
[39] W. K. Mak, D. P. Morton, and R. K. Wood, *Monte Carlo bounding techniques for determining solution quality in stochastic programs*, Oper. Res. Lett., 24 (1999), pp. 47–56.
[40] A. Mokkadem and M. Pelletier, *A generalization of the averaging procedure: The use of two-time-scale algorithms*, SIAM J. Control Optim., 49 (2011), p. 1523.
[41] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, Berlin, 2006.
[42] E. O. Nsoesie, R. J. Beckman, S. Shashaani, K. S. Nagaraj, and M. V. Marathe, *A simulation optimization approach to epidemic forecasting*, PLOS One, 8 (2013), https://doi.org/10.1371/journal.pone.0067164.
[43] C. Osorio and M. Bierlaire, *A Surrogate Model for Traffic Optimization of Congested Networks: An Analytic Queueing Network Approach*, Report TRANSP-OR 90825, École Polytechnique Fédérale de Lausanne, France, 2009.
[44] R. Pasupathy, *On Choosing parameters in retrospective-approximation algorithms for stochastic root finding and simulation optimization*, Oper. Res., 58 (2010), pp. 889–901.
[45] R. Pasupathy and S. Ghosh, *Simulation optimization: A concise overview and implementation guide*, in Theory Driven by Influential Applications, INFORMS Tutor., INFORMS, Catonsville, MD, 2013, pp. 122–150.
[46] R. Pasupathy, P. W. Glynn, S. G. Ghosh, and F. S. Hahemi, *On Sampling Rates in Simulation-Based Recursions*, preprint, http://www.optimization-online.org/DB_HTML/2016/03/5361.html, 2017.
[47] R. Pasupathy and S. G. Henderson, *A testbed of simulation-optimization problems*, in Proceedings of the 2006 Winter Simulation Conference, IEEE Press, Piscataway, NJ, pp. 255–263.
[48] R. Pasupathy and S. G. Henderson, *SimOpt: A library of simulation optimization problems*, in Proceedings of the 2011 Winter Simulation Conference, S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu, eds., IEEE Press, Piscataway, NJ, 2011, pp. 4075–4085.
[49] R. Pasupathy and B. W. Schmeiser, *Retrospective-approximation algorithms for multidimensional stochastic root-finding problems*, ACM Trans. Model. Comput. Simul., 19 (2009), No. 5.
[50] B. T. Polyak and A. B. Juditsky, *Acceleration of stochastic approximation by averaging*, SIAM J. Control Optim., 30 (1992), pp. 838–855.
[51] M. J. D. Powell, *UOBYQA: Unconstrained optimization by quadratic approximation*, Math. Program., 92 (2002), pp. 555–582.
[52] H. Robbins and S. Monro, *A stochastic approximation method*, Ann. Math. Stat., 22 (1951), pp. 400–407.
[53] J. O. Royset and R. Szechtman, *Optimal budget allocation for sample average approximation*, Oper. Res., 61 (2013), pp. 762–776.

[54] A. Ruszczynski and A. Shapiro, eds., *Stochastic Programming*, Handbooks Oper. Res. Management Sci. 10, Elsevier, New York, 2003.
[55] R. J. Serfling, *Approximation Theorems of Mathematical Statistics*, John Wiley, New York, 1980.
[56] A. Shapiro, D. Dentcheva, and A. Ruszczynski, *Lectures on Stochastic Programming: Modeling and Theory*, SIAM, Philadelphia, PA, 2009.
[57] J. C. Spall, *Adaptive stochastic approximation by the simultaneous perturbation method*, IEEE Trans. Automat. Control, 45 (2000), pp. 1839–1853.
[58] J. C. Spall, *Introduction to Stochastic Search and Optimization*, John Wiley, Hoboken, NJ, 2003.
[59] M. L. Stein, *Interpolation of Spatial Data: Some Theory for Kriging*, Springer, New York, 1999.
[60] F. Yousefian, A Nedić, U. V., and Shanbhag, *On stochastic gradient and subgradient methods with adaptive step length sequences*, Automatica, 48 (2012), pp. 56–67.