A SMOOTHING DIRECT SEARCH METHOD FOR MONTE CARLO-BASED BOUND CONSTRAINED COMPOSITE NONSMOOTH OPTIMIZATION

XIAOJUN CHEN*, C. T. KELLEY†, FENGMIN XU‡, AND ZAIKUN ZHANG§

January 31, 2018

Abstract. We propose and analyze a smoothing direct search algorithm for finding a minimizer of a nonsmooth nonconvex function over a box constraint set, where the objective function values cannot be computed directly, but are approximated by Monte Carlo simulation. In the algorithm, we adjust the stencil size, the sample size, and the smoothing parameter simultaneously so that the stencil size goes to zero faster than the smoothing parameter and the square root of the sample size goes to infinity faster than the reciprocal of the stencil size. We prove that with probability one any accumulation point of the sequence generated by the algorithm is a Clarke stationary point. We report on numerical results from statistics and financial applications.

Key words. Sampling methods, direct search algorithm, Monte Carlo simulation, non-smooth optimization, smoothing functions, Clarke stationarity

AMS subject classifications. 65K05, 65K10, 90C30

1. Introduction. Let $G : R^{n+m} \to R$ be a locally Lipschitz continuous function, $F : R^n \to R^m$ be a continuously differentiable function and $f : R^n \to R$ be a composite nonsmooth function of the form

$$f(x) = G(x, F(x)). \tag{1.1}$$

In this paper, we consider the nonsmooth minimization problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in X := \{x \in R^n \mid \ell \le x \le u\}, \end{array} \tag{1.2}$$

where $\ell, u \in R^n$, $-\infty < \ell_i < u_i < \infty$, $i = 1, \ldots, n$ and the function value $F(x)$ is not computed exactly, but rather approximated by a Monte Carlo simulation F^N, where N is the sample size of the Monte Carlo simulation.

We assume that the function G admits a smoothing approximation in the sense of [10]. By this we mean that there is a family of differentiable functions $\hat{G}(\cdot, \mu)$ which converges to G as $\mu \to 0$. We make this precise in Definition 2.3. In this paper we use the smoothing parameter μ to design an algorithm which takes gradients of $\hat{G}$ to determine descent directions and then decreases μ as the iteration progresses.

Our use of the term nonsmooth in this context is standard [14] and refers to functions which, while not differentiable (smooth), are locally Lipschitz continuous and the generalized derivatives are well-defined in the sense of [14].

In [11] we considered a similar problem in the more general situation where the objective function was not everywhere defined and capturing the domain of f was part of the problem. In this paper the objective function is everywhere defined and can be approximated by a smoothing approach. The results in this paper exploit the structure of that special case to simplify the analysis and improve the efficiency of the method via smoothing.

A large class of nonsmooth functions have the form (1.1). The box constraint does not restrict applications where the objective functions have minimizers in a compact set. In other words, if the level set

*Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China (maxjchen@polyu.edu.hk). The work of this author was partially supported by Hong Kong Research Grant Council grant PolyU153000/15p.

†North Carolina State University, Department of Mathematics, Box 8205, Raleigh, NC 27695-8205, USA (Tim_Kelley@ncsu.edu). The work of this author was partially supported by the Consortium for Advanced Simulation of Light Water Reactors (www.casl.gov), and Simulation of Nuclear Reactors under U.S. Department of Energy Contract No. DE-AC05-00OR22725, Army Research Office grant #W911NF-16-1-0504 and National Science Foundation Grant DMS-1406349.

‡ School of Economics and Finance, Xi'an Jiaotong University, Xi'an, 710049, China (fengminxu@mail.xjtu.edu.cn). The work of this author was partially supported by the Chinese Natural Science Foundation 11571271, 71331001.

§Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China (zaikun.zhang@polyu.edu.hk). The work of this author was supported by the start-up grant 1-ZVHT from The Hong Kong Polytechnic University and the Early Career Scheme grant PolyU 253012/17P from RGC Hong Kong.

$\{x : f(x) \leq f(x_0)\}$ of f with a point $x_0 \in R^n$ is bounded, then minimizing f over R^n can be equivalently written as a box constrained optimization problem (1.2). One example (for which $m = n$) is the expected value version of the stochastic variational inequality problem [12, 49]: Given the induced probability space $(\Xi \subset R^\ell, \mathcal{A}, \mathcal{P})$ and a convex set $\Omega \subseteq R^n$, find $x^* \in \Omega$ such that

$$(x - x^*)^T F(x^*) \geq 0, \quad \forall x \in \Omega, \tag{1.3}$$

where $F(x) := E[\phi(\xi, x)]$, and $\phi : \Xi \times R^n \to R^n$ is continuously differentiable with respect to x for almost all $\xi \in \Xi$ and $\mathcal{A}$-measurable with respect to ξ. The stochastic variational inequality problem (1.3) reduces to the stochastic complementarity problem:

$$x \geq 0, \quad F(x) \geq 0, \quad x^T F(x) = 0, \tag{1.4}$$

when $\Omega = R^n_+ = \{x \in R^n \,|\, x \geq 0\}$, and the system of stochastic nonsmooth equations:

$$F(x) = 0$$

when $\Omega = R^n$. In this case, the approximation is via Monte Carlo simulation

$$F(x) = E[\phi(\xi, x)] \approx F^N(x) := \frac{1}{N} \sum_{i=1}^{N} \phi(\xi^i, x),$$

where N is the sample size and ξ^i, $i = 1, \ldots, N$ are observations of $\xi \in \Xi$.

We can express problem (1.3) as a minimization problem [20]

$$\min_{x \in R^n} \|x - \mathrm{Proj}_\Omega(x - F(x))\|_2^2 \tag{1.5}$$

where Proj_Ω is the projection onto the set Ω. In this formulation the optimal function value is zero.

Another example of problem (1.2) is the ℓ_1-norm regularized minimization problem

$$\min_{x \in R^n} \|F(x)\|_1 + \lambda \|x\|_1, \tag{1.6}$$

where $F(x)$ is approximated by a Monte Carlo simulation.

If problems (1.5) and (1.6) have minimizers in a compact set, then there is a box constraint set X, which might be difficult to determine a priori, such that problems (1.5) and (1.6) can be equivalently written as problem (1.2) with $f(x) = \|x - \mathrm{Proj}_\Omega(x - F(x))\|_2^2$ and $f(x) = \|F(x)\|_1 + \lambda\|x\|_1$, respectively.

We will exploit the composition structure by using a smoothing function for f. We will show that if f is replaced by the outcome of Monte Carlo simulation and one has full knowledge of the nonsmoothness, we can develop a smoothing direct search method with Monte Carlo simulation, which has global convergence to a Clarke stationary point of problem (1.2) with probability one (w.p.1.).

For example, if f has the form (1.1), we can define the smoothing function for f as

$$\hat{f}(x, \mu) = \hat{G}(x, F(x), \mu) \tag{1.7}$$

and, when $F(x)$ is replaced by the Monte Carlo outcome $F^N(x)$, we set the smoothing Monte Carlo simulation as

$$\tilde{f}(x, \mu, N) = \hat{G}(x, F^N(x), \mu). \tag{1.8}$$

We consider stencil-based direct search methods in this paper. By this we mean that at each step in the optimization the function is evaluated at a set of points of the form

$$\{x \pm h v_i\}_{i=1}^n \cup \{x\}$$

where x is the current point and h is the stencil size. The directions v can be quite general [3, 28, 31, 32]. In this paper we will use the positive and negative coordinate directions, which is sufficient for our application.

For any fixed smoothing parameter $\mu > 0$, the function $\hat{f}$ is continuously differentiable with respect to x. The main contribution of this paper is to propose a smoothing direct search algorithm with Monte Carlo simulation for solving problem (1.2) and prove the convergence of the algorithm when the stencil size h and smoothing parameter μ go to zero with the rate $h/\mu \to 0$, and the sample size N goes to infinity with the rate $(h\sqrt{N})^{-1} \to 0$.

Convergence analysis of direct search algorithms for smooth optimization problems where function values can be computed exactly have been well studied in [15, 16, 23, 28, 47]. Nonsmooth problems have been considered in [2–4, 11, 28]. Theory and algorithms for problems where the function evaluations require embedded Monte Carlo simulations have been carefully considered for optimization problems [11, 29, 33, 45, 46, 49] and for nonlinear equations [55, 58]. The new algorithm in this paper exploits the structure of the problem and properties of smoothing methods to allow for using coordinate basis as fixed stencil search directions, simplifying the approaches of [3, 11, 28] for nonsmooth problems while preserving the convergence results.

Direct search methods have been coupled with randomized methods in [52] where the randomization was in the sampling and the optimization problem itself was deterministic. In [48] a generalized pattern search algorithm was applied to a problem where the objective function f was an expectation. The objective was a function of continuous and categorical variables and was assumed to be a smooth function of the continuous variables. Neither of these papers consider nonsmooth problems.

This paper is organized as follows. In section 2, we present a smoothing direct search algorithm for problem (1.2) where the function values $f(x)$ can be computed directly, and prove the convergence of the algorithm. In section 3, we extend the algorithm and convergence analysis to a smoothing direct search algorithm for (1.2) where the function values $f(x)$ cannot be computed directly, but are approximated by Monte Carlo simulation. In section 4, we present numerical experiments which include examples from statistical learning, and portfolio selection using test problems from the OR-Library [5] and real data from the Shanghai-Shenzhen stock market.

2. A smoothing direct search method. We begin by reviewing sampling direct search methods in the context of the smooth optimization problem. Let the set of search directions be an orthonormal basis $V = \{v_1, v_2, \ldots, v_n\}$. Let h be the stencil size along those search directions. A stencil centered at x with h is the set of points $\{x \pm hv_i\}_{i=1}^n \cup \{x\}$. More general stencils can be used [3, 28, 31, 32] but are not needed for the applications in this paper.

The concept of stencil failure is important in both the algorithms and the analysis.

DEFINITION 2.1. *We say that* stencil failure *has occurred if*

$$f(x) \le f(x \pm hv_i) \quad \text{for } i = 1, \ldots, n. \tag{2.1}$$

For simplicity, in this paper we will use

$$V = \{e_1, e_2, \ldots, e_n\}, \tag{2.2}$$

for each iteration. Here e_i is the i-th coordinate vector. The algorithms and convergence analysis can be extended to an orthonormal basis.

It is easy to show [16, 27, 28] that if f is Lipschitz continuously differentiable in X, then (2.1) implies that

$$\|\nabla f(x)\| = O(h) \tag{2.3}$$

uniformly for $x \in X$. To see this note that Lipschitz continuous differentiability of f in X and (2.1) imply that

$$\frac{\partial f(x)}{\partial x_i} h + O(h^2) = f(x + he_i) - f(x) \ge 0, \quad i = 1, \ldots, n$$

and

$$-\frac{\partial f(x)}{\partial x_i} h + O(h^2) = f(x - he_i) - f(x) \ge 0, \quad i = 1, \ldots, n$$

uniformly in X. Hence $\|\nabla f(x)\| = O(h)$, uniformly in X.

Sampling methods evaluate the objective function at the points of the stencil. If the current point is the best (stencil failure at the current point), then the stencil size is reduced. If the current point is not the best on the stencil, then the new best point becomes the current point. Algorithm `direct_search` is a version of the method for minimizing a continuously differentiable objective function f within a convex set X.

Algorithm `direct_search` (x, f, h)

for forever **do**
  $f_{base} = f(x)$
  $f_{min} = \min\{f(y) \,|\, y = x \pm hv,\ v \in V \text{ and } y \in X\}$
  $\hat{y} \in \{y \,|\, f(y) = f_{min},\ y = x \pm hv,\ v \in V \text{ and } y \in X\}$
  if $f_{min} \geq f_{base}$ **then**
    $h \leftarrow h/2$
  else
    $x \leftarrow \hat{y}$
  end if
end for

Algorithm `direct_search` generates two sequences $\{x_k\}$ and $\{h_k\}$. The convergence analysis is based on their subsequences $\{\tilde{x}_t\}$ and $\{\tilde{h}_t\}$, whose generation is described in the following box with initial points x_0 and h_0, and $t = 0$.

Sequences generated by Algorithm `direct_search`
for $k \geq 0$
  $\hat{y} \in \operatorname{argmin}\{f(y) \,|\, y = x_k \pm h_k v,\ v \in V \text{ and } y \in X\}$
  if $f(\hat{y}) \geq f(x_k)$ (**stencil failure**)
    set $x_{k+1} = x_k$, $h_{k+1} = h_k/2$, $t = t + 1$, $\tilde{x}_t = x_{k+1}$, $\tilde{h}_t = h_{k+1}$
  else
    $x_{k+1} = \hat{y}$, $h_{k+1} = h_k$
  end if
end for

In Algorithm `direct_search`, we must choose an initial point $x \in X$. This requirement for a feasible starting point in a box constraint set X is easy to satisfy. The worst case cost of a sweep through the stencils for a fixed $h > 0$ is sampling every point on the finite set

$$\Omega_h(x) = \{x_0 + mhv \ \mid \ m = 1, 2, \dots \text{ and } v \in V\} \cup X,$$

where x_0 is either the initial point or the first point after h has been reduced. This worst case is, in our experience, very unlikely.

In our formulation the search is non-opportunistic. By this we mean that the minimization is done over the entire stencil. The analysis is the same for the opportunistic version, where the first point with a smaller function value than f_{base} is used. The reason is that the stencil size is only reduced when the stencil fails. Stencil failure can only take place if the entire stencil is sampled. Before then, it does not matter if the search is opportunistic or not.

The convergence proof of Algorithm `direct_search` is based on the stencil directions such that if stencil failure happens at the current point, then some type of approximate necessary condition holds. This idea can be made very general with different stencils and different smoothness requirements on the objective function f [2–4, 11, 28].

We consider the following first-order stationarity measure

$$\chi(x) \ = \ \max_{x+d \in X, \|d\| \leq 1} [-d^T \nabla f(x)]. \tag{2.4}$$

It is easy to check that, if $x \in X$ is a local minimizer of (1.2), then $\chi(x) = 0$.

PROPOSITION 2.2. *Assume that f is Lipschitz continuously differentiable. Let $\{x_k\}$ with $x_0 \in X$ be the sequence generated by Algorithm* `direct_search`. *Then*

$$\liminf_{k\to\infty} \chi(x_k) = 0. \tag{2.5}$$

Moreover, stencil failure happens at infinitely many iterates, and for each limit point x of the stencil failure iterates, it holds that

$$\chi(x) \;=\; 0. \tag{2.6}$$

Proof. Let h_0 be the initial stencil size and

$$X_t \;=\; X \cap \left\{ x_0 + \sum_{i=1}^{n} \frac{j_i h_0}{2^t} e_i \;\mid\; j_i = 0, \pm 1, \pm 2, \cdots \right\}, \quad t = 0, 1, 2, \ldots. \tag{2.7}$$

Since X is bounded, X_t is a finite set, and it contains at least one iterate. Stencil failure occurs at $\tilde{x}_t$, the last iterate contained in X_t and the size of the stencil at that iteration is $\tilde{h}_t = h_0/2^t$. For each $t \geq 1$, define

$$I_t^+ \;=\; \{i \mid \tilde{x}_t + \tilde{h}_t e_i \in X, 1 \leq i \leq n\}, \tag{2.8}$$

$$I_t^- \;=\; \{i \mid \tilde{x}_t - \tilde{h}_t e_i \in X, 1 \leq i \leq n\}, \tag{2.9}$$

and denote $g_t = \nabla f(\tilde{x}_t)$. Let L be the Lipschitz constant of ∇f. Then by the definition of stencil failure and Taylor's theorem,

$$0 \;\leq\; f(\tilde{x}_t + \tilde{h}_t e_i) - f(\tilde{x}_t) \;\leq\; \tilde{h}_t e_i^T g_t + \frac{L}{2}\tilde{h}_t^2 \quad \text{for all} \quad i \in I_t^+, \tag{2.10}$$

$$0 \;\leq\; f(\tilde{x}_t - \tilde{h}_t e_i) - f(\tilde{x}_t) \;\leq\; -\tilde{h}_t e_i^T g_t + \frac{L}{2}\tilde{h}_t^2 \quad \text{for all} \quad i \in I_t^-, \tag{2.11}$$

and consequently,

$$e_i^T g_t \;\geq\; -\frac{L\tilde{h}_t}{2} \quad \text{for all} \quad i \in I_t^+, \tag{2.12}$$

$$e_i^T g_t \;\leq\; \frac{L\tilde{h}_t}{2} \quad \text{for all} \quad i \in I_t^-. \tag{2.13}$$

By the assumption, there exists a positive constant Υ such that

$$\|\nabla f(x)\| \;\leq\; \Upsilon \quad \text{for all} \quad x \in X. \tag{2.14}$$

Specifically, $\|g_t\| \leq \Upsilon$. Therefore, for each d such that $\tilde{x}_t + d \in X$ and $\|d\| \leq 1$, it holds that

$$\begin{aligned}
-d^T g_t \;&=\; -\sum_{i=1}^{n} d_i (g_t)_i \\
&=\; -\sum_{i \in I_t^+ \setminus I_t^-} d_i (g_t)_i - \sum_{i \in I_t^- \setminus I_t^+} d_i (g_t)_i - \sum_{i \in I_t^+ \cap I_t^-} d_i (g_t)_i - \sum_{i \notin I_t^+ \cup I_t^-} d_i (g_t)_i \\
&\leq\; n \max\left\{ \frac{L\tilde{h}_t}{2}, \Upsilon\tilde{h}_t \right\} + n\frac{L\tilde{h}_t}{2} + n\Upsilon\tilde{h}_t \\
&\leq\; 3n \max\left\{ \frac{L\tilde{h}_t}{2}, \Upsilon\tilde{h}_t \right\},
\end{aligned} \tag{2.15}$$

where the first inequality uses the fact that for each d with $\tilde{x}_t + d \in X$, $i \notin I_t^-$ implies $d_i > \tilde{h}_t$ and $i \notin I_t^+$ implies $d_i < \tilde{h}_t$, since X is a bounded box.

Hence, we obtain

$$\chi(\tilde{x}_t) \;\le\; 3n \max\left\{\frac{L\tilde{h}_t}{2}, \Upsilon\tilde{h}_t\right\} \;\to\; 0 \quad \text{when} \quad t \to \infty. \tag{2.16}$$

Since $\{\tilde{x}_t\}$ is a subsequence of $\{x_k\}$ and X is bounded, we conclude that

$$\liminf_{k\to\infty} \chi(x_k) \;=\; 0. \tag{2.17}$$

If x is an accumulation point of the stencil failure iterates $\{\tilde{x}_t\}$, then continuity of χ implies that $\chi(x) = 0$.□

2.1. Nonsmooth f. In this subsection, we consider problem (1.1) where F can be computed exactly. We will use smoothing methods which approximate f by a parameterized family of smoothing functions $\hat{f}(\cdot, \mu)$ given by (1.7), where $\mu > 0$ is the smoothing parameter.

We formally give a definition of smoothing functions used in this paper.

DEFINITION 2.3. *[10] Let $f : R^n \to R$ be a locally Lipschitz continuous function. We call $\hat{f} : R^n \times (0, \infty) \to R$ a smoothing function of f, if $\hat{f}(\cdot, \mu)$ is continuously differentiable and $\nabla\hat{f}(\cdot, \mu)$ is Lipschitz continuous in R^n for any fixed $\mu \in (0, \infty)$, and*

$$\lim_{x\to\hat{x}, \mu\downarrow 0} \hat{f}(x, \mu) = f(\hat{x}). \tag{2.18}$$

The limit in (2.18) is simultaneous in x and μ for all sequences $x_k \to \hat{x}$ and $\mu_k \to 0$ ($\mu_k \ge 0$).

Throughout this subsection we let $\nabla\hat{f}$ denote the gradient $\hat{f}$ with respect to x.

ASSUMPTION 2.1. *(i) There are constants $c_1, c_2 \ge 0$ such that for any $x \in R^n$, $\mu \in (0, 1]$,*

$$|f(x) - \hat{f}(x, \mu)| \le \mu(c_1 + c_2|f(x)|). \tag{2.19}$$

(ii) $\hat{f}$ satisfies the gradient consistency condition,

$$\partial f(x) = \mathrm{con}\{v \,|\, \nabla\hat{f}(x_k, \mu_k) \to v, \text{ for } x_k \to x, \ \mu_k \downarrow 0\}, \tag{2.20}$$

where "con" denotes the convex hull and $\partial f(x)$ is the Clarke subgradient at x.

(iii) There are $\Upsilon > 0$, $\Gamma > 0$ and $\mu_- > 0$ such that $\|\nabla\hat{f}(x, \mu)\| \le \Upsilon$ and

$$\|\nabla\hat{f}(x, \mu) - \nabla\hat{f}(y, \mu)\| \le \frac{\Gamma}{\mu}\|x - y\| \tag{2.21}$$

uniformly in x, $y \in X$, and $\mu \in (0, \mu_-)$.

In Assumption 2.1, c_1, c_2, Υ, Γ and μ_- are fixed constants which are independent of x. Since X is bounded and f is continuous, Assumption 2.1 (i) implies that there is a constant C such that

$$|f(x) - \hat{f}(x, \mu)| \le \mu C, \quad \text{for} \quad x \in X, \ \mu \in (0, 1]$$

which means that $\hat{f}$ converges to f uniformly as $\mu \to 0$.

In section 4, we use examples to illustrate the definition of smoothing functions and Assumption 2.1.

Note that X is bounded and hence there are only finitely many points in the stencil for each h. Therefore the stencil will fail infinitely often.

In the case where f is known exactly and there is no embedded Monte Carlo simulation, we propose a smoothing direct search algorithm, Algorithm `smoothing_search` that decreases μ and h simultaneously, but in a way that ensures $h/\mu \to 0$ as $\mu \to 0$, which will be important in the convergence analysis.

Algorithm `smoothing_search` generates three sequences $\{x_k\}$, $\{h_k\}$ and $\{\mu_k\}$. The convergence analysis is based on their subsequences $\{\tilde{x}_t\}$, $\{\tilde{h}_t\}$ and and $\{\tilde{\mu}_t\}$, whose generation is described in the following box with initial points x_0, h_0, and μ_0, and $t = 0$.

Algorithm `smoothing_search` $(x, \hat{f}, h, \mu, \tau)$

for forever **do**
  $\hat{f}_{base} = \hat{f}(x, \mu)$
  $\hat{f}_{min} = \min\{\hat{f}(y, \mu) \,|\, y = x \pm hv,\ v \in V \text{ and } y \in X\}$
  $\hat{y} \in \{y \,|\, \hat{f}(y, \mu) = \hat{f}_{min},\ y = x \pm hv,\ v \in V \text{ and } y \in X\}$
  if $\hat{f}_{min} \geq \hat{f}_{base}$ **then**
    $h \leftarrow h/2; \quad \mu \leftarrow \mu/2^{\tau}$
  else
    $x \leftarrow \hat{y}$
  end if
end for

Sequences generated by Algorithm `smoothing_search`
for $k \geq 0$
  $\hat{y} \in \text{argmin}\{\hat{f}(y, \mu) \,|\, y = x_k \pm h_k v,\ v \in V \text{ and } y \in X\}$
  if $\hat{f}(\hat{y}, \mu_k) \geq \hat{f}(x_k, \mu_k)$ **(Stencil failure)**
    set $x_{k+1} = x_k$, $h_{k+1} = h_k/2$, $\mu_{k+1} = \mu_k/2^{\tau}$, $t = t + 1$, $\tilde{x}_t = x_{k+1}$, $\tilde{h}_t = h_{k+1}$, $\tilde{\mu}_t = \mu_{k+1}$
  else
    $x_{k+1} = \hat{y}$, $h_{k+1} = h_k$, $\mu_{k+1} = \mu_k$, $k = k + 1$
  end if
end for

In Algorithm `smoothing_search`, $\tau \in (0, 1)$ is an input parameter. We must choose an initial point $x \in X$, the initial stencil size $h > 0$ and the initial smoothing parameter $\mu > 0$.

As an extension of (2.4), we use

$$\tilde{\chi}(x) \;=\; \min_{v \in \partial f(x)} \big(\max_{x+d \in X, \|d\| \leq 1} -d^T v \big) \tag{2.22}$$

to measure the first-order sationarity of x with respect to problem (1.2) when f is locally Lipschitz continuous but not necessarily differentiable, where $\partial f(x)$ is the Clarke subdifferential of f at x [14,41]. If f is smooth, then $\tilde{\chi}(\cdot)$ is the same as $\chi(\cdot)$. Moreover, if x is a local minimizer of problem (1.2), then there exists a $v \in \partial f(x)$ such that

$$\max_{x+d \in X, \|d\| \leq 1} [-d^T v] = 0,$$

that is, $\tilde{\chi}(x) = 0$.

The convergence result follows the same argument as in the proof of Proposition 2.2 and Assumption 2.1 on the smoothing function of f.

THEOREM 2.4. *Assume that Assumption 2.1 holds. Let $\{x_k, \mu_k\}$ with $x_0 \in X$ and $\mu_0 > 0$ be the iterates generated by Algorithm* `smoothing_search`*, and*

$$\chi_k(x) \;=\; \max_{x+d \in X, \|d\| \leq 1} [-d^T \nabla \hat{f}(x, \mu_k)]. \tag{2.23}$$

Then

$$\liminf_{k \to \infty} \chi_k(x_k) \;=\; 0. \tag{2.24}$$

Moreover, stencil failure happens at infinitely many iterates, and for each limit point x of the stencil failure iterates, it holds that

$$\tilde{\chi}(x) \;=\; 0. \tag{2.25}$$

Proof. Define X_t by (2.7). As in the proof of Proposition 2.2, we denote the last iterate in X_t, where stencil failur occurs by $\tilde{x}_t$, the corresponding stencil size by $\tilde{h}_t = h_0/2^t$, and the corresponding smoothing parameter by $\tilde{\mu}_t = \tilde{h}_t^\tau$. For each $t \geq 1$, define I_t^+ and I_t^- in the same way as in the proof of Proposition 2.2, and denote $\hat{g}_t = \nabla \hat{f}(\tilde{x}_t, \tilde{\mu}_t)$. According to the definition of stencil failure and Taylor expansion, noticing part (iii) of Assumption 2.1, we have

$$0 \;\leq\; \hat{f}(\tilde{x}_t + \tilde{h}_t e_i, \tilde{\mu}_t) - \hat{f}(\tilde{x}_t, \tilde{\mu}_t) \;\leq\; \tilde{h}_t e_i^T \hat{g}_t + \frac{\Gamma}{2\tilde{\mu}_t}\tilde{h}_t^2 \quad \text{for all} \quad i \in I_t^+, \tag{2.26}$$

$$0 \;\leq\; \hat{f}(\tilde{x}_t - \tilde{h}_t e_i, \tilde{\mu}_t) - \hat{f}(\tilde{x}_t, \tilde{\mu}_t) \;\leq\; -\tilde{h}_t e_i^T \hat{g}_t + \frac{\Gamma}{2\tilde{\mu}_t}\tilde{h}_t^2 \quad \text{for all} \quad i \in I_t^-, \tag{2.27}$$

and consequently,

$$e_i^T \hat{g}_t \;\geq\; -\frac{\Gamma \tilde{h}_t}{2\tilde{\mu}_t} \quad \text{for all} \quad i \in I_t^+, \tag{2.28}$$

$$e_i^T \hat{g}_t \;\leq\; \frac{\Gamma \tilde{h}_t}{2\tilde{\mu}_t} \quad \text{for all} \quad i \in I_t^-. \tag{2.29}$$

By Assumption 2.1, there exists a positive constant Υ such that $\|\hat{g}_t\| \leq \Upsilon$. Using similar argument to those for (2.15) and (2.16), we have

$$\max_{\tilde{x}_t + d \in X, \|d\| \leq 1} -d^T \hat{g}_t \;\leq\; 3n \max\left\{\frac{\Gamma \tilde{h}_t}{2\tilde{\mu}_t},\; \Upsilon \tilde{h}_t\right\}. \tag{2.30}$$

Noticing the fact that $\tilde{h}_t/\tilde{\mu}_t \to 0$, we have

$$\max_{\tilde{x}_t + d \in X, \|d\| \leq 1} -d^T \hat{g}_t \;\to\; 0 \quad \text{when} \quad t \to \infty, \tag{2.31}$$

which implies (2.24).

Let x be a limit point of $\{\tilde{x}_t\}$, and $\{\tilde{x}_{t_i}\}$ be a subsequence that converges to x. Since $\{\hat{g}_t\}$ is bounded, we may suppose that $\{\hat{g}_{t_i}\}$ converges to a point v (if not, replace $\{t_i\}$ by an appropriately chosen subsequence). Let

$$d^*(v) \;\in\; \operatorname*{argmax}_{x + d \in X, \|d\| \leq 1} [-d^T v], \tag{2.32}$$

and

$$y_i \;=\; \frac{x - \tilde{x}_{t_i} + d^*(v)}{\max\{\|x - \tilde{x}_{t_i} + d^*(v)\|,\; 1\}}. \tag{2.33}$$

Then $\|y_i\| \leq 1$, and $\tilde{x}_{t_i} + y_i \in X$ due to the convexity of X ($\tilde{x}_{t_i} + y_i$ lies on the line segment between $\tilde{x}_{t_i}$ and $x + d^*$). Hence

$$\begin{aligned} 0 \;\leq\; \max_{x + d \in X, \|d\| \leq 1} -d^T v \;&=\; -\left(d^*(v)\right)^T v \\ &= \lim_{i \to \infty} -y_i^T \hat{g}_{t_i} \\ &\leq \lim_{i \to \infty} \max_{\tilde{x}_{t_i} + d \in X, \|d\| \leq 1} -d^T \hat{g}_{t_i} \\ &= 0. \end{aligned} \tag{2.34}$$

Notice that $v \in \partial f(x)$ according to the gradient consistency of $\hat{f}$. By the definition (2.22) of $\tilde{\chi}(\cdot)$, we have

$$0 \;\leq\; \tilde{\chi}(x) \;\leq\; \max_{x + d \in X, \|d\| \leq 1} -d^T v \;=\; 0, \tag{2.35}$$

which completes the proof. ∎

3. Smoothing direct search method with Monte Carlo simulations. In this section we extend the algorithms and analysis from § 2 to the case where f is nonsmooth and approximated by a Monte Carlo simulation. Deterministic direct search methods for nonsmooth optimization problems have been studied in [2–4, 11, 23, 28, 42].

In this section, we assume that for any $x \in X$, $\mu > 0$, we can estimate the value of $\hat{f}(x, \mu)$ by Monte Carlo simulation $\tilde{f}(x, \mu, N)$ with N realizations. The value of $\tilde{f}(x, \mu, N)$ is random and the sample of N realizations is independently identically distributed (iid). We can view $\tilde{f}(x, \mu, N)$ as defined on a common probability space (see [49, pp. 156] for details).

The following is an assumption on the effectiveness of $\tilde{f}(\cdot, \cdot, N)$ as an approximation of $\hat{f}(\cdot, \cdot)$.

ASSUMPTION 3.1. *For each $p \in (0, 1/2)$, there exist constants $\delta \in (0, 1)$, $c_F > 0$, $\bar{N} > 0$, and $\bar{\mu} > 0$ such that*

$$\text{Prob}\left(\sup_{x \in X} |\hat{f}(x, \mu) - \tilde{f}(x, \mu, N)| \geq \frac{c_f}{N^p}\right) \leq \delta \tag{3.1}$$

for each $N \geq \bar{N}$ and $\mu \in (0, \bar{\mu}]$.

Consider the composite nonsmooth function in the form (1.1) with

$$F(x) = E[\phi(\xi, x)], \quad x \in X, \tag{3.2}$$

where ξ is a random vector. Let

$$\tilde{F}^N(x) = \frac{1}{N} \sum_{i=1}^{N} \phi(\xi_i, x), \tag{3.3}$$

$\xi_1, \xi_2, \ldots, \xi_N$ being iid samples of ξ. Assume that $\phi(\xi, x)$ is *sub-exponential* for each $x \in X$ (see Appendix B for the definition of sub-exponential random variables/vectors), and that $\phi(\xi, \cdot)$ is L-Lipschitz continuous with respect to $x \in X$ for a constant L independent of ξ. Then, as we show in Appendix B, for any $p \in (0, 1/2)$, there exist constants $\delta \in (0, 1)$, $c_F > 0$ and $\bar{N} > 0$ such that

$$\text{Prob}\left(\sup_{x \in X} \|F(x) - \tilde{F}^N(x)\| \geq \frac{c_F}{N^p}\right) \leq \delta \tag{3.4}$$

for each $N \geq \bar{N}$. If G is Lipschitz continuous with respect to $F(x)$ and $\hat{G}$ satisfies Assumption 2.1, then there exists a constant L_F independent of N and μ such that

$$|\hat{G}(x, F(x), \mu) - \hat{G}(x, \tilde{F}^N(x), \mu)| \leq L_F \|F(x) - \tilde{F}^N(x)\|.$$

This, together with (3.4), implies that function

$$\tilde{f}(x, \mu, N) \equiv \hat{G}(x, \tilde{F}^N(x), \mu)$$

fulfills Assumption 3.1 with $c_f = L_F c_F$.

Algorithm `mc_smoothing_search` for the embedded Monte Carlo case is a simple extension of Algorithm `smoothing_search`.

In Algorithm `mc_smoothing_search`, $\tau \in (0, 1)$ and $\gamma > 1$ are input parameters. The objective function f is evaluated through $\tilde{f}(x, \mu, N)$, the Monte Carlo simulation of $\hat{f}(x, \mu)$ with sample size N, where $\hat{f}(x, \mu)$ is a smoothing function of f that is defined in Definition 2.3 and satisfies Assumption 2.1.

Algorithm `mc_smoothing_search` generates four sequences $\{x_k\}$, $\{h_k\}$, $\{\mu_k\}$ and $\{N_k\}$. The convergence analysis is based on their subsequences $\{\tilde{x}_t\}$, $\{\tilde{h}_t\}$, $\{\tilde{\mu}_t\}$ and $\{\tilde{N}_t\}$, whose generation is described in the following box with initial points x_0, h_0, μ_0 and N_0, and $t = 0$.

Algorithm `mc_smoothing_search` $(x, \tilde{f}, h, \mu, N, \tau, \gamma)$

for forever **do**
  $\tilde{f}_{base} = \tilde{f}(x, \mu, N)$
  $\tilde{f}_{min} = \min\{\tilde{f}(y, \mu, N) \mid y = x \pm hv, v \in V \text{ and } y \in X\}$
  $\hat{y} \in \{y \mid \tilde{f}(y, \mu, N) = \tilde{f}_{min},\ y = x \pm hv,\ v \in V \text{ and } y \in X\}$
  if $\tilde{f}_{min} \geq \tilde{f}_{base}$ **then**
    $h \leftarrow h/2; \quad \mu \leftarrow \mu/2^{\tau}; \quad N \leftarrow 4^{\gamma} N$
  else
    $x \leftarrow \hat{y}$
  end if
end for

Sequences generated by Algorithm `mc_smoothing_search`
for $k \geq 0$
  $\hat{y} \in \operatorname{argmin}\{\tilde{f}(y, \mu, N) \mid y = x_k \pm h_k v,\ v \in V \text{ and } y \in X\}$
  if $\tilde{f}(\hat{y}, \mu_k, N_k) \geq \tilde{f}(x_k, \mu_k, N_k)$ (**Stencil failure**)
    set $x_{k+1} = x_k$, $h_{k+1} = h_k/2$, $\mu_{k+1} = \mu_k/2^{\tau}$, $N_{k+1} = 4^{\gamma} N_k$,
    $t = t + 1$, $\tilde{x}_t = x_{k+1}$, $\tilde{h}_t = h_{k+1}$, $\tilde{\mu}_t = \mu_{k+1}$, $\tilde{N}_t = N_{k+1}$
  else
    $x_{k+1} = \hat{y}$, $h_{k+1} = h_k$, $\mu_{k+1} = \mu_k$, $N_{k+1} = N_k$, $k = k + 1$
  end if
end for

As in Proposition 2.2 and Theorem 2.4, the boundedness of X ensures the convergence of the algorithm. Let $x_0 \in X$ denote the initial point of Algorithm `mc_smoothing_search`, $h_0 > 0$ the initial stencil size, $\mu_0 > 0$ the initial smoothing parameter, and $N_0 > 0$ the initial sample size.

The main result of the paper is Theorem 3.1, which states that w.p.1. the iteration has an accumulation point which is a Clarke stationary point.

THEOREM 3.1. *Suppose that the Monte Carlo simulations in Algorithm* `mc_smoothing_search` *are mutually independent for different N. Assume that Assumptions 2.1 and 3.1 hold. Let $\{x_k, \mu_k, N_k\}$ be the sequence generated by Algorithm* `mc_smoothing_search`*. Then*

$$\text{Prob}\left(\liminf_{k \to \infty} \chi_k(x_k) = 0\right) \ = \ 1, \tag{3.5}$$

where χ_k is defined in (2.23), and

$$\text{Prob}(\{x_k\} \textit{ has an accumulation point } x \textit{ such that } \tilde{\chi}(x) = 0) \ = \ 1. \tag{3.6}$$

Proof. Define X_t by (2.7). As before we denote the last iterate in X_t, where stencil failure occurs by $\tilde{x}_t$, the corresponding stencil size by $\tilde{h}_t = h_0/2^t$, and the corresponding smoothing parameter by $\tilde{\mu}_t = \mu_0/2^{t\tau}$. The sample size at this point in the algorithm is $\tilde{N}_t = 4^{t\gamma} N_0$. Define index sets I_t^+ and I_t^- in the same way as in the proof of Proposition 2.2. Since stencil failure happens at $\tilde{x}_t$, we have

$$0 \ \leq \ \tilde{f}(\tilde{x}_t + \tilde{h}_t e_i, \tilde{\mu}_t, \tilde{N}_t) - \tilde{f}(\tilde{x}_t, \tilde{\mu}_t, \tilde{N}_t) \quad \text{for all} \quad i \in I_t^+, \tag{3.7}$$

$$0 \ \leq \ \tilde{f}(\tilde{x}_t - \tilde{h}_t e_i, \tilde{\mu}_t, \tilde{N}_t) - \tilde{f}(\tilde{x}_t, \tilde{\mu}_t, \tilde{N}_t) \quad \text{for all} \quad i \in I_t^-. \tag{3.8}$$

Let p be a constant such that

$$\frac{1}{2\gamma} < p < \frac{1}{2}. \tag{3.9}$$

Then $\tilde{h}_t \tilde{N}_t^p \to \infty$ as $t \to \infty$. Set $\delta \in (0,1)$ and $c_f > 0$ to be the constants that fulfill Assumption 3.1, and consider the event

$$E_t \;=\; \left\{ \sup_{x \in X} |\hat{f}(x, \tilde{\mu}_t) - \tilde{f}(x, \tilde{\mu}_t, \tilde{N}_t)| \le \frac{c_f}{\tilde{N}_t^p} \right\}. \tag{3.10}$$

By assumption, $\{E_t\}_{t=1}^{\infty}$ are mutually independent, and

$$\text{Prob}(E_t) \;\ge\; 1 - \delta > 0$$

for each t sufficiently large so that $\tilde{N}_t \ge \bar{N}$ and $\tilde{\mu}_t \le \bar{\mu}$. Therefore,

$$\text{Prob}(E_t \text{ happens for infinitely many } t) \;=\; 1. \tag{3.11}$$

When E_t happens, according to (3.7) and (3.8), we have

$$-\frac{2c_f}{\tilde{N}_t^p} \;\le\; \hat{f}(\tilde{x}_t + \tilde{h}_t e_i, \tilde{\mu}_t) - \hat{f}(\tilde{x}_t, \tilde{\mu}_t) \;\le\; \tilde{h}_t e_i^T \hat{g}_t + \frac{\Gamma}{2\tilde{\mu}_t} \tilde{h}_t^2 \quad \text{for all} \quad i \in I_t^+, \tag{3.12}$$

$$-\frac{2c_f}{\tilde{N}_t^p} \;\le\; \hat{f}(\tilde{x}_t - \tilde{h}_t e_i, \tilde{\mu}_t) - \hat{f}(\tilde{x}_t, \tilde{\mu}_t) \;\le\; -\tilde{h}_t e_i^T \hat{g}_t + \frac{\Gamma}{2\tilde{\mu}_t} \tilde{h}_t^2 \quad \text{for all} \quad i \in I_t^-, \tag{3.13}$$

where $\hat{g}_t = \nabla \hat{f}(\tilde{x}_t, \tilde{\mu}_t)$ and Γ is the constant in item (iii) of Assumption 2.1. Subsequently, it holds that

$$e_i^T \hat{g}_t \;\ge\; -\frac{\Gamma \tilde{h}_t}{2\tilde{\mu}_t} - \frac{2c_f}{\tilde{h}_t \tilde{N}_t^p} \quad \text{for all} \quad i \in I_t^+, \tag{3.14}$$

$$e_i^T \hat{g}_t \;\le\; \frac{\Gamma \tilde{h}_t}{2\tilde{\mu}_t} + \frac{2c_f}{\tilde{h}_t \tilde{N}_t^p} \quad \text{for all} \quad i \in I_t^-. \tag{3.15}$$

By Assumption 2.1, there exists a positive constant Υ such that $\|\hat{g}_k\| \le \Upsilon$. Using similar argument as (2.15) and (2.16), we obtain from (3.14) and (3.15) that

$$\max_{\tilde{x}_t + d \in X, \|d\| \le 1} -d^T \hat{g}_t \;\le\; 3n \max\left\{ \frac{\Gamma \tilde{h}_t}{2\tilde{\mu}_t} + \frac{2c_f}{\tilde{h}_t \tilde{N}_t^p},\; \Upsilon \tilde{h}_t \right\}. \tag{3.16}$$

Hence, by (3.11) and the fact that $\tilde{h}_t \to 0$, $\tilde{h}_t / \tilde{\mu}_t \to 0$, and $\tilde{h}_t \tilde{N}_t^p \to \infty$, we have

$$\text{Prob}(\{\chi_k(x_k)\} \text{ has a subsequence that converges to zero}) \;=\; 1, \tag{3.17}$$

which implies (3.5).

When $\liminf_{k \to \infty} \chi_k(x_k) = 0$, let $\{k_i\}$ be the index sequence such that $\chi_{k_i}(x_{k_i}) \to 0$. Since $\{x_{k_i}\}$ is bounded (guaranteed the boundedness of X), it has an accumulation point x. By the same argument that leads to the second part of Theorem 2.4, we have that $\tilde{\chi}(x) = 0$. Thus (3.6) holds. ∎

4. Numerical experiments. In this section, we test Algorithm `mc_smoothing_search` on two problems: a stochastic optimization problem arising from censored regression and a two-stage optimization problem arising from portfolio management. The problems in § 4.1 and 4.2.1 are derived from applications, but use synthetic data to enable us to control the sample size.

4.1. Censored regression. We consider the following regularized censored regression problem [1, 6, 34, 35, 51, 53]

$$\begin{aligned} &\min_{x \in R^n} \; f(x) \\ &\text{s.t.} \; -e \le x \le e, \end{aligned} \tag{4.1}$$

where $e \in R^n$ is the vector with all its entries being one and

$$f(x) = E_{c,y}[(\max(c^T x, 0) - y)^2] + \lambda \sum_{i=1}^{n} \log(1 + |(x)_i|). \tag{4.2}$$

Here the random variable pair (c, y) represents a data set of interest ($c \in R^n$, $y \in R$), and $\lambda > 0$ is a regularization parameter.

The regularization term

$$\lambda \sum_{i=1}^{n} \log(1 + |(x)_i|)$$

in the objective function is used to enforce sparsity.

We assume that $c \sim N(0, I)$, and $y = \max(c^T x^* + \epsilon, 0)$ for some underlying ground truth feature x^* and unobservable noise $\epsilon \sim N(0, \sigma^2)$. Moreover, we assume x^* is sparse, that is, x^* has few nonzero entries. Using the concave regularized model (4.1), we want to recover a sparse feature x to approximate x^* as accurate as possible given that $x^* \in \{x \mid \|x\|_\infty \leq 1\} = [-e, e]$.

The functions $(\max(c^T x, 0) - y)^2$ and $\log(1 + |(x)_i|)$ are not differentiable but do admit smoothing functions (see Appendix A). Using the smoothing functions, we can define a smoothing function $\hat{f}_{c,y}(x, \mu)$ for $(\max(c^T x, 0) - y)^2$ which satisfies Assumption 2.1. From the convexity of $(\max(c^T x, 0) - y)^2$, the Clarke subdifferential and the expectation can be exchanged, that is,

$$\partial E_{c,y}[(\max(c^T x, 0) - y)^2] = E_{c,y}[\partial(\max(c^T x, 0) - y)^2]$$

(see [14]). Moreover, $E_{c,y}[\hat{f}_{c,y}(x, \mu)]$ is a smoothing function for $E_{c,y}[(\max(c^T x, 0) - y)^2]$, and satisfies Assumption 2.1 (see [8]). Problem (4.1) is a constrained nonsmooth nonconvex optimization problem where the objective function values cannot be computed directly.

In practice the data in these problems are limited. To mimic the finite size of the data we will pose an approximation to problem (4.1) that replaces the expectation with the sample average of a finite, but large, data set. We will manage the sampling in the algorithm by randomly sampling from that data set. To this end, we consider $X = \{x \mid \|x\|_\infty \leq 1\} = [-e, e]$, and we randomly generate a true feature $x^* \in R^{20}$ whose 5 nonzero entries are from uniform distribution on $[-1, 1]$. Independently, we generate samples c_i from $c \sim N(0, I)$ and ϵ_i from $\epsilon \sim N(0, 0.01)$ with a sample size 10^7. Let $y_i = \max(c_i^T x^* + \epsilon_i, 0)$. The new problem is an approximation to (4.1) with a finite data set. We have

$$\bar{f}(x, 10^7) = \frac{1}{10^7} \sum_{i=1}^{10^7} [(\max(c_i^T x, 0) - y_i)^2] + \lambda \sum_{i=1}^{n} \log(1 + |(x)_i|). \tag{4.3}$$

We used the regularization parameter $\lambda = 10^{-2}$. This is large enough to capture the sparsity exactly and small enough to allow us to observe several iterations before the iteration stagnates.

We configure the optimization as follows:

- The algorithmic parameters are $c = 2$, $\gamma = 1.5$, and $\tau = 0.5$.
- We begin with $h = 0.5$ and terminate when $h \leq 10^{-3}$.
- $N = 100$ and $\mu = 0.1$ at the beginning of the iteration.

Given N, for each evaluation of f, we independently and randomly sample vectors $(c_i, y_i), i = 1, \ldots, N$ from the data set $(c_i, y_i), i = 1, \ldots, 10^7$ generated above. Note that we sample with replacement, following the bootstrapping technique in statistics [19]. This allows the sample size to be larger than 10^7. Then we compute smoothing approximation $\tilde{f}(x, \mu, N)$ of the following function

$$\bar{f}(x, N) = \frac{1}{N} \sum_{i=1}^{N} [(\max(c_i^T x, 0) - y_i)^2] + \lambda \sum_{i=1}^{n} \log(1 + |(x)_i|) \tag{4.4}$$

by using smoothing functions for $\max(., 0)$ and $|\cdot|$.

For the initial point $x_0 = (0, \ldots, 0)^T$ we performed 20 runs of Algorithm `mc_smoothing_search`. In Figure 4.1, we show histories of the distance $\|x_k - x^*\|$ and the value $f(x_k)$.

FIG. 4.1. *Histories of the distance* $\|x_k - x^*\|$ *and the value* $f(x_k)$

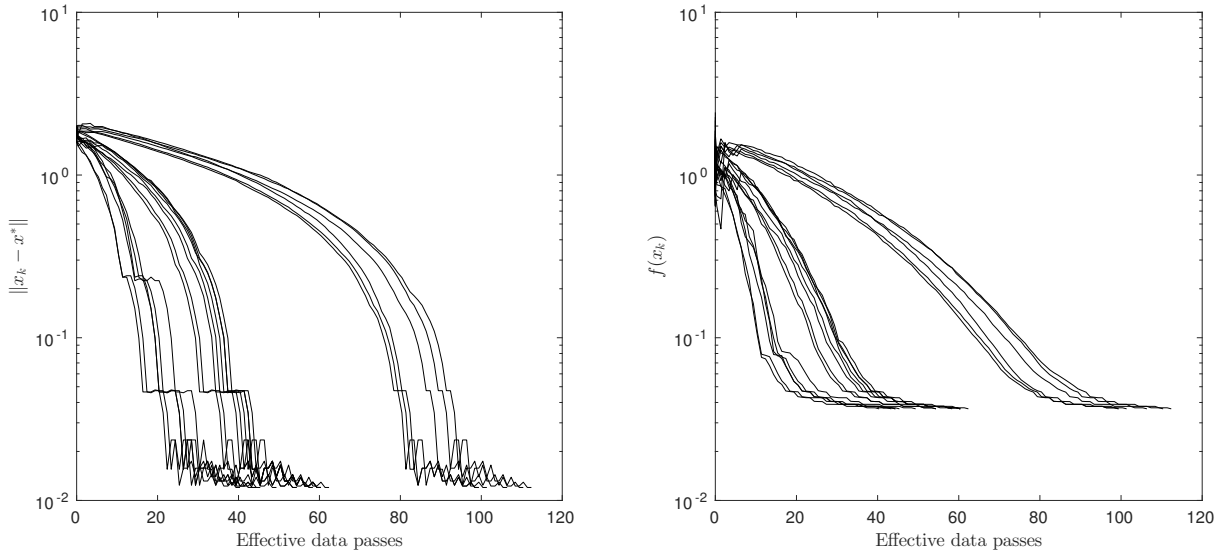


Figure 4.1 illustrates several properties of the algorithm and the problem. At the end of the iteration, all of the iteration histories are very similar. The theory would lead one to expect similar histories if N is large. On the other hand, the initial value of N is large enough to cause considerable variation in f early in the iteration. This variation accounts for the differences in the histories. Finally, the iteration stagnates in the terminal phase when the differences from the iterates and x^* are roughly at the level of the regularization parameter. The reason for this is that the regularization term would dominate the error term when x is near x^*. While a smaller regularization would defer the stagnation, it would make it harder to capture the sparsity. Our choice of $\lambda = 10^{-2}$ captures the sparsity exactly. At each final iterate x_k, we have $(x_k)_i = (x^*)_i = 0$ for all $i \in S^c$, where $S = \{i \,|\, (x^*)_i \neq 0, i = 1, \ldots, n\}$, the support set of x^*.

Figure 4.2 plots $\|x_k - x^*\|$ and $f(x_k)$ against the number of *Effective Data Passes* performed until the k-th iteration (EDP_k), which is defined as the number of samples made during the first k iterations divided by the data size 10^7. Note that EDP_k increases with respect to k, but not strictly. The number of Effective Data Passes may stay unchanged during several successive iterations because the former changes only after a stencil failure, which happens only once in a few iterations. Consequently, a single number of Effective Data Passes can correspond to several values of $\|x_k - x^*\|$ or $f(x_k)$. That is why we observe vertical lines in Figure 4.2.

Figure 4.2 shows that the algorithm is capable of achieving considerable progress using very few Effective Data Passes. Both $\|x_k - x^*\|$ and $f(x_k)$ are reduced significantly even before one single Effective Data Pass is made. This shows the effectiveness of our sampling strategy, which increases the sample size steadily in course of the iterations. The stagnations in the final stage of the plots are seemingly more visible than in Figure 4.1. This is because the sample size increases rapidly, and hence the variations in $\|x_k - x^*\|$ or $f(x^*)$ are less visible when plotted against EDP_k than plotted against k.

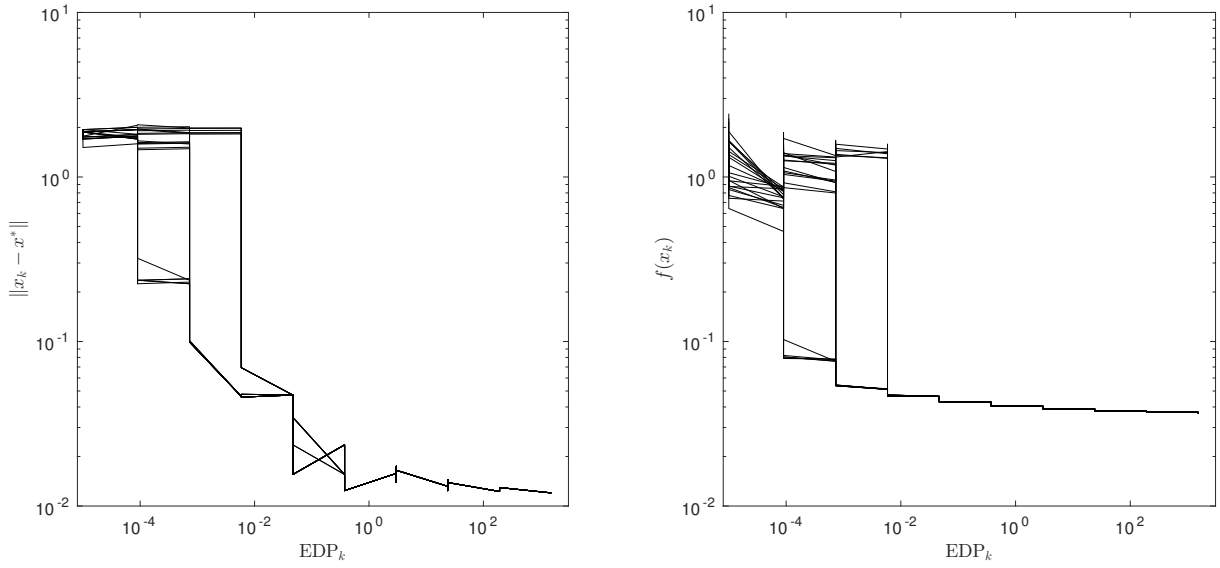
Note that we begin with $h = 0.5$ and terminate when $h \leq 10^{-3}$. By the structure of Algorithm `mc_smoothing_search`, the algorithmic parameters $\tau = 0.5$, $\gamma = 1.5$ and the initial values $N_0 = 100$, $\mu_0 = 0.1$, we know that after $t = 9$ iterations, $\tilde{h}_9 \leq 10^{-3}$, and hence $\tilde{\mu}_9 = \mu_0/2^{t\tau} = 0.1/2^{9\times 0.5} = 0.044$ and $\tilde{N}_9 = 4^{t\gamma} N_0 = 4^{9\times 1.5} 100 \approx 1.34\mathrm{E}10$.

4.2. Portfolio management. Consider ν assets. Let $u \in R^\nu$ denote the random returns of them, and

$$r = E[u], \qquad C = E[(u - r)(u - r)^T]. \tag{4.5}$$

Here r is the vector of expected returns of the different assets, and C is the covariance matrix of the return on the assets in the portfolio. When r and C are known, as discussed in [44], the Markowitz mean-variance

FIG. 4.2. *$\|x_k - x^*\|$ and $f(x_k)$ plotted against the number of Effective Data Passes EDP_k*



model [36, 37] for portfolio selection can be formulated as

$$\begin{aligned} \min_{w} \ & \frac{1}{2} w^T C w - \eta r^T w \\ \text{s.t.} \ & e^T w = 1 \\ & a \leq w \leq b, \end{aligned} \tag{4.6}$$

where w denotes the weights of the assets in the portfolio, $a \in R^\nu$ and $b \in R^\nu$ $(a \leq b)$ are lower and upper bounds enforced on w, and η is a nonnegative parameter (called the risk aversion factor) to balance the conflicting aspects of minimizing the risk measured by $w^T C w$ and maximizing the expected return measured by $r^T w$. The Markowitz mean-variance model [36, 37] was first proposed and solved when the total return is known. The model captures the essence of two conflicting aspects in portfolio management; namely, the risk and the return.

The use of mean variance analysis in portfolio selection normally requires the knowledge of means, variances, and covariances of returns of all securities under consideration. However, in general, these data are not known exactly. Treating their estimates as if they were the exact parameters can lead to suboptimal portfolio choices.

The experiments reported in [22, 25, 30] show that, influenced by the sampling error, portfolios selected with the mean-variance model by Markowitz are not as efficient as an equally weighted portfolio. Other results [13, 39] show that the mean-variance model tends to magnify the errors associated with the estimates. In this section, we consider an optimal parameter selection model based on the Markowitz mean-variance model to find optimal parameters for portfolio selection.

For simplicity, here we only consider the case where C is positive definite and the feasible set $\{w \,|\, e^T w = 1, a \leq w \leq b\}$ is nonempty. Given a, b, and η, problem (4.6) has a unique solution w. In other words, w is uniquely defined by a, b, and η, the values of which will determine the quality of the portfolio selected by solving problem (4.6). A common measure for the quality is the Sharpe ratio [50]

$$\mathrm{SR} = \frac{r^T w}{\sqrt{w^T C w}}.$$

The Sharpe ratio characterizes how well the return of an asset compensates the investor for the risk taken. In general, a strategy is better than others if its Sharpe ratio is higher.

In practice, a, b, and η are usually set by investors empirically according to their preferences. We consider selecting them by solving the two-stage optimization problem

$$\begin{aligned} \max_{(a,b,\eta)\in\Omega} \ & \frac{r^T w(a,b,\eta)}{\sqrt{w(a,b,\eta)^T C w(a,b,\eta)}} \\ \text{where } w(a,b,\eta) = \operatorname*{argmin}_{w} \ & \frac{1}{2} w^T C w - \eta r^T w \\ \text{s.t.} \ & e^T w \ = \ 1 \\ & a \ \le \ w \ \le b, \end{aligned} \tag{4.7}$$

where the feasible set

$$\Omega = [\underline{a}, \overline{a}] \times [\underline{b}, \overline{b}] \times [\underline{\eta}, \overline{\eta}] \quad \text{with} \quad \underline{\eta} < \overline{\eta}$$

is given. The number of variables of the first level problem is

$$\#\{i|\underline{a}_i < \overline{a}_i, i = 1, \ldots, \nu\} + \#\{i|\underline{b}_i < \overline{b}_i, i = 1, \ldots, \nu\} + 1.$$

For example, if we choose

$$\underline{a}_i = \overline{a}_i = 0, \text{ for } i \neq 1, \quad \underline{a}_1 = 0, \ \overline{a}_1 = 1 \quad \text{and} \quad \underline{b}_i = \overline{b}_i = 1, \text{ for } i \neq 2, \quad \underline{b}_2 = 0, \ \overline{b}_2 = 1, \tag{4.8}$$

then the number of variables of the first level problem is 3.

Finding optimal parameters a, b, η is a challenging problem. Since C is positive definite, the second level optimization problem has a unique solution. Hence $w(a, b, \eta)$ is well defined, and it is Lipschitz continuous with respect to (a, b, η). However, $w(a, b, \eta)$ is not differentiable, and the covariance matrix C and the vector of expected return r cannot be computed directly in general. We will use the barrier method [43, Chapter 19] to solve the second stage problem of (4.7) and define a smoothing function $w_\mu(a, b, \eta)$ [43]. In particular, we use Algorithm `mc_smoothing_search` to solve

$$\begin{aligned} \max_{(a,b,\eta)\in\Omega} \ & \frac{r^T w_\mu(a,b,\eta)}{\sqrt{w_\mu(a,b,\eta)^T C w_\mu(a,b,\eta)}} \\ \text{where } w_\mu(a,b,\eta) = \operatorname{argmin} \ & \frac{1}{2} w^T C w - \eta r^T w - \mu \sum_{i=1}^{\nu} \log(s_i) - \mu \sum_{i=1}^{\nu} \log(t_i) \\ \text{s.t.} \ & e^T w \ = \ 1 \\ & w - a - s = 0 \\ & b - w - t = 0. \end{aligned} \tag{4.9}$$

In this section, we report numerical results that we obtained with Algorithm `mc_smoothing_search` for five standard data sets from the OR-Library [5], the SSE50 index, the CSI 100 index, and the CSI 300 index from Shanghai-Shenzhen stock market. The data are the weekly or daily prices of the component stocks for the eight stock market indices drawn from different countries. See Table 4.1 for the description of the data sets. The ν and T columns are the number of the component assets included in the index and the number of the observations for the assets, respectively.

We report on two experiments: randomly generated problems in §4.2.1, which use the mean and the covariance matrix generated from the real data in Table 4.1 and rolling window procedures for out-of-sample comparison in §4.2.2, which use the stock prices to generate the returns of assets and the covariance matrix by Monte Carlo simulation.

4.2.1. Randomly generated problems. We choose the following parameters as input data of Algorithm `mc_smoothing_search`:

$$h = 0.5, \quad \mu = 0.1, \quad N = 100, \quad \tau = 0.5, \quad \gamma = 1.5.$$

We choose the feasible set X as in (4.8).

TABLE 4.1
Description of the eight real data sets

Data set	ν	Location	Index	T	Description
data 1	31	Hong Kong	Hang Seng	291	weekly prices from 1992 to 1997
data 2	85	Germany	DAX 100	291	weekly prices from 1992 to 1997
data 3	89	UK	FTSE 100	291	weekly prices from 1992 to 1997
data 4	98	USA	S&P 100	291	weekly prices from 1992 to 1997
data 5	225	Japan	Nikkei 225	291	weekly prices from 1992 to 1997
SSE50	50	China	SSE50	501	daily prices from 2013 to 2015
CSI100	100	China	CSI 100	501	daily prices from 2013 to 2015
CSI300	300	China	CSI 300	401	daily prices from 2011 to 2013

For all tests, we terminated the algorithm if the stencil size is less than 10^{-2}.

For each data set in Table 4.1, we first calculate the average $\hat{r} \in R^{\nu}$ and the covariance matrix $\hat{C} \in R^{\nu \times \nu}$ for the returns of the assets. Given a sample size N, we generate i.i.d. random vectors $u_i \in R^{\nu}$, $i = 1, \ldots, N$ normally distributed with mean $\hat{r}$ and covariance matrix $\hat{C}$, that is

$$u_i = \hat{r} + \hat{C}^{\frac{1}{2}} \text{randn}(\nu, 1), \qquad i = 1, 2, ..., N,$$

and then take r^N to be the sample average of $u_i, i = 1, 2, ..., N$, and C^N to be the sample covariance matrix

$$r^N = \frac{1}{N} \sum_{i=1}^{N} u_i \quad \text{and} \quad C^N = \frac{1}{N} \sum_{i=1}^{N} (u_i - r^N)(u_i - r^N)^T.$$

Then we compute the smoothing approximation for the problem of minimizing the negative Sharpe ratio which is given by (4.9) with $r = r^N$, $C = C^N$, and $x = (a_1, b_2, \eta)$.

For each data set in Table 4.1, we use Algorithm `mc_smoothing_search` to solve problem (4.7) with the starting point $(a_1, b_2, \eta) = (0, 1, 0.5)$. Table 4.2 presents the results. From Table 4.2, we can see the optimal value of objective function (Opt. sharpe ratio) at the final iteration is bigger than the value of objective function at the point $w = e/\nu$, which is a feasible point of problem (4.6) with $a = 0$ and $b = e$.

TABLE 4.2
Numerical results for the portfolio management problem with randomly generated data

data set	data1	data2	data3	data4	data5	SSE50	CSI100	CSI300
lower bound a_1	4.69E-1	4.22E-1	6.56E-1	1.25E-1	1.00E00	2.00E-2	1.00E-2	7.50E-1
upper bound b_2	8.13E-1	8.75E-1	9.69E-1	9.69E-1	7.12E-1	1.14E-1	3.85E-1	7.50E-1
risk aversion η	9.69E-1	5.31E-1	3.43E-1	4.38E-1	8.59E-1	1.25E-1	6.25E-1	6.41E-1
opt. sharpe ratio	1.57E-1	2.85E-1	2.51E-1	2.47E-1	9.76E-2	3.92E-1	2.74E-1	7.98E-2
sharpe ratio e/ν	1.04E-1	9.15E-2	1.53E-1	1.99E-1	-4.90E-2	2.36E-1	2.65E-1	-9.05E-2

4.2.2. Problems with rolling window procedures. For a given data set, assuming that the observations of the stock prices are $\{P_{i,t} : 1 \le i \le \nu, 1 \le t \le T\}$, we can compute the (logarithmic) returns of the stocks:

$$r_{i,t} = \log \frac{P_{i,t+1}}{P_{i,t}}, \ \ i = 1, \ldots, \nu, \ t = 1, \ldots, T-1.$$

For the purpose of numerical comparisons, we partition the data set into two subsets: a training set and a testing set. The training set, called in-sample set, consists of the first half of the data set and is used to compute an optimal parameter x^* and the corresponding optimal portfolio selection $w(x^*)$. The testing set,

called out-of-sample, consists of the second half of the data set and is used to test how well the optimal parameter x^* and the corresponding optimal portfolio selection $w(x^*)$.

More exactly, for stock i with $i = 1, \ldots, \nu$, we can use the training set to compute the in-sample expectation and the standard deviation by

$$\bar{\mu}_i = \frac{1}{M}\sum_{t=1}^{M} r_{i,t} \quad \text{and} \quad \bar{\sigma}_i = \sqrt{\frac{1}{M}\sum_{t=1}^{M}(r_{i,t} - \bar{\mu}_i)^2},$$

respectively, where $M = (T-1)/2$. As is standard in finance [24], we simulate the out-of-sample prices as follows. Let N be the sample size. Then at the j-th simulation ($1 \le j \le N$), for $M+1 \le t \le T-1$, if the price $S_{i-1,t}^{(j)}$ of stock i at an out-of-sample time $t-1$ is known, the price $S_{i,t}^{(j)}$ of this stock at time t is generated by

$$S_{i,t}^{(j)} = S_{i,t-1}^{(j)} \exp(\bar{\mu}_i + \bar{\sigma}_i Z),$$

where $S_{i,M}^{(j)} = P_{i,M}$ for all $1 \le j \le N$ and Z is randomly produced by the standard normal distribution $N(0,1)$. In a similar way, we can calculate the (logarithmic) returns by this simulation

$$r_{i,t}^{(j)} = \log \frac{S_{i,t+1}^{(j)}}{S_{i,t}^{(j)}}, \quad t = M+1, \ldots, T-1.$$

For $t = M+1, \ldots, T-1$, denote the column vector $r_t^{(j)}$ with its i-th component being $r_{i,t}^{(j)}$ and its average vector $\bar{r}_t = \frac{1}{N}\sum_{j=1}^{N} r_t^{(j)}$, the sample mean r^N and the sample variance C^N of the out-of-sample can be computed by

$$r^N = \frac{1}{M}\sum_{t=M+1}^{T-1} \bar{r}_t \quad \text{and} \quad C^N = \frac{1}{M}\sum_{t=M+1}^{T-1} (\bar{r}_t - r^N)(\bar{r}_t - r^N)^T.$$

Then we solve problem (4.9) with the sample mean r^N and the sample variance C^N to obtain the optimal parameter x^* and the corresponding optimal portfolio selection $w(x^*)$ by Algorithm `mc_smoothing_search`.

We choose the following parameters as input data of Algorithm `mc_smoothing_search`:

$$h = 0.5, \quad \mu = 0.1, \quad N = 10, \quad \tau = 0.5, \quad \gamma = 1.5.$$

We choose the feasible set X as in (4.8). For all tests, we choose the starting point $(a_1, b_2, \eta) = (0, 1, 0.5)$ and terminated the algorithm when the sample size N gets larger than 10^5.

To evaluate the quality of the optimal portfolio selection $w(x^*)$, we shall make use of the real out-of-sample data. We denote by r^{out} and C^{out} the mean and variance of the real returns of the out-of-sample set; namely,

$$r^{out} = \frac{1}{M}\sum_{t=M+1}^{T-1} r_t \quad \text{and} \quad C^{out} = \frac{1}{M}\sum_{t=M+1}^{T-1} (r_t - r^{out})(r_t - r^{out})^T,$$

where r_t is the vector formed by the stock prices $r_{i,t} (i = 1, \ldots, n)$ for $t = M+1, \ldots, T-1$. Then we can calculate the Sharpe ratio of the optimal solution $w(x^*)$ by using r^{out} and C^{out} as

$$\mathrm{SR}^* = \frac{(r^{out})^T w(x^*)}{\sqrt{w(x^*)^T C^{out} w(x^*)}}.$$

In Table 4.3, for all the eight data sets, we list the optimal values of a_1, b_2, η achieved by Algorithm `mc_smoothing_search`, and the corresponding SR^*. For comparison, we also list the Sharpe ratio of the average strategy (namely, taking $1/\nu$ portion of each portfolio) using r^{out} and C^{out}. From Table 4.3, we can see that using Algorithm `mc_smoothing_search` to solve problem (4.7) can provide a portfolio strategy with higher Sharpe ratio than the average strategy for all data sets.

TABLE 4.3
Numerical results for the portfolio management problem with rolling window procedures

data set	data1	data2	data3	data4	data5	SSE50	CSI100	CSI300
lower bound a_1	1.00E-3	1.18E-2	1.12E-2	1.02E-1	4.40E-2	2.00E-2	1.00E-2	3.33E-3
upper bound b_2	3.14E-1	2.31E-1	6.36E-1	4.15E-2	1.29E-2	5.83E-1	2.60E-1	2.53E-1
risk aversion η	2.81E-1	9.38E-1	6.25E-1	6.25E-2	1.00E00	7.19E-1	2.50E-1	7.50E-1
sharpe ratio SR*	3.35E-1	2.36E-1	3.72E-1	5.12E-1	2.19E-1	2.71E-1	4.33E-1	2.19E-1
sharpe ratio e/ν	1.57E-1	2.10E-1	2.79E-1	3.44E-1	-3.85E-2	2.36E-1	2.65E-1	3.18E-3

5. Conclusions. In this paper we propose a smoothing direct search algorithm with Monte Carlo simulation **Algorithm** `mc_smoothing_search` for the constrained nonsmooth nonconvex optimization problem (1.2), where the objective function value $f(x)$ cannot be computed directly, but are approximated by Monte Carlo simulation. This algorithm updates the stencil size h, smoothing parameter μ and the sample size N simultaneously with the rate $h/\mu \to 0$, and $(h\sqrt{N})^{-1} \to 0$. We prove that any accumulation point of the sequence generated by the algorithm satisfies the first order optimality condition $\tilde{\chi}(x) = 0$ with probability one, where $\tilde{\chi}(x)$ is defined by (2.22). We report on a set of numerical experiments which illustrate the analysis and show that **Algorithm** `mc_smoothing_search` is an effective method for minimizing nonsmooth functions whose function values cannot be computed directly but are approximated by Monte Carlo simulation.

Appendix A. Smoothing functions.

We give an example of smoothing functions to explain Assumption 2.1. Let $f(x) = 2\max(0, p(x))$, where $p : R^n \to R$ is twice continuously differentiable with

$$\|\nabla p(x) \nabla p(x)^T\| \le \Gamma.$$

We use the smoothing function

$$\hat{f}(x, \mu) = p(x) + \sqrt{p(x)^2 + 4\mu^2}, \tag{A.1}$$

and $V = \{e_1, \ldots, e_n\}$, the unit coordinate directions in R^n.

Clearly part (i) of Assumption 2.1 holds with $c_1 = 2$ and $c_2 = 0$, since

$$|f(x) - \hat{f}(x, \mu)| \le 2\mu.$$

Now we consider part (ii) of Assumption 2.1. The Clarke subgradient has the form

$$\partial f(x) = 2 \begin{cases} \nabla p(x) & \text{if } p(x) > 0 \\ \mathbf{0} & \text{if } p(x) < 0 \\ [0, 1] \nabla p(x) & \text{if } p(x) = 0 \end{cases} \tag{A.2}$$

and the gradient of the smoothing function is

$$\nabla \hat{f}(x, \mu) = \left(1 + \frac{p(x)}{\sqrt{p(x)^2 + 4\mu^2}} \right) \nabla p(x).$$

Hence, we have

$$\|\nabla \hat{f}(x, \mu)\| \le 2 \|\nabla p(x)\|.$$

If p is Lipschitz continuously differentiable on a convex and compact set Ω, then there is an Υ such that $\|\nabla \hat{f}(x, \mu)\| \le \Upsilon$ on Ω.

It is easy to see that for $p(x) \ne 0$, f is differentiable at x and

$$\partial f(x) = \nabla f(x) = \text{con}\{ v \,|\, \nabla \hat{f}(x_k, \mu_k) \to v, \text{ for } x_k \to x, \ \mu_k \downarrow 0 \}.$$

For $p(x)=0$, since $0 \le 1+\frac{p(x)}{\sqrt{p(x)^2+4\mu^2}} \le 2$, we have

$$\mathrm{con}\{v \,|\, \nabla \hat{f}(x_k,\mu_k) \to v, \text{ for } x_k \to x,\ \mu_k \downarrow 0\,\} \subseteq \partial f(x).$$

Now, let $\mu_k^2 = (1-h^2)p(x_k)^2/(4h^2)$ for some $h \in (0,1]$. Then for $x_k \to x$ with $p(x_k) \downarrow 0$, we have $\mu_k \downarrow 0$ and

$$\nabla \hat{f}(x_k,\mu_k) = (1+h)\nabla p(x_k) \to (1+h)\nabla p(x),$$

and for $x_k \to x$ with $p(x_k) \uparrow 0$, we have $\mu_k \downarrow 0$ and

$$\nabla \hat{f}(x_k,\mu_k) = (1-h)\nabla p(x_k) \to (1-h)\nabla p(x).$$

Moreover, if we take $\mu_k = \sqrt{|p(x_k)|}$, then for $x_k \to x$, we have $p(x_k) \to 0$, $\mu_k \downarrow 0$ and

$$\nabla \hat{f}(x_k,\mu_k) = \left(1+\frac{p(x_k)}{\sqrt{p(x_k)^2+4|p(x_k)|}}\right)\nabla p(x_k) \to \nabla p(x).$$

Hence we find that for $p(x)=0$,

$$\partial f(x) = [0,2]\nabla p(x) = \mathrm{con}\{v \,|\, \nabla \hat{f}(x_k,\mu_k) \to v, \text{ for } x_k \to x,\ \mu_k \downarrow 0\,\}.$$

Finally, we consider part (iii) of Assumption 2.1. Since

$$\nabla^2 \hat{f}(x,\mu) = \left(1+\frac{p(x)}{\sqrt{p(x)^2+4\mu^2}}\right)\nabla^2 p(x) + \frac{4\mu^2}{(p(x)^2+4\mu^2)^{\frac{3}{2}}}\nabla p(x)\nabla p(x)^T,$$

we have

$$\|\nabla^2 \hat{f}(x,\mu)\| \le \frac{1}{2\mu}\|\nabla p(x)\nabla p(x)^T\| + 2\|\nabla^2 p(x)\| \le \frac{\Gamma}{2\mu} + 2\|\nabla^2 p(x)\|,$$

which implies that part (iii) of Assumption 2.1 holds.

A smoothing function of $|p(x)|$ can be defined by using the relation $|p(x)| = \max(0,p(x)) + \max(0,-p(x))$ and a smoothing function of $\max(0,p(x))$. For example, using (A.1), we can have a smoothing function $\sqrt{(p(x)^2+4\mu^2)}$ for $|p(x)|$.

There is a detailed discussion of smoothing functions in [10].

Appendix B. The proof of (3.4).

In this appendix, we will show the existence of $\delta \in (0,1)$, $c_F > 0$, and $\bar{N} > 0$ that fulfill (3.4). To this end, we have to study the uniform convergence rate of the empirical mean (3.3), which has been investigated in [26, 59] under certain conditions. In particular, if $\phi(\xi,x) = \mathbf{1}(\xi \le x)$, where ξ, $x \in \mathbb{R}$, and $\mathbf{1}(E)$ is the indicator function of an event E, then inequality (3.4) is indeed satisfiable with $p = 1/2$ (see [18, 38]). However, in general, one can only achieve (3.4) for $p < 1/2$. Our argument here is essentially an extension of the discussions in [59, Section 3].

Before the proof, we recall that $\phi(\xi,x)$ is sub-exponential for each $x \in X$. As defined in [56, Definition 5.13], a real-value random variable ζ is called a sub-exponential [1] random variable if

$$\sup_{p \ge 1} p^{-1} E(|\zeta|^p)^{1/p} < \infty. \tag{B.1}$$

The quantity on the left-hand side of (B.1), often denoted by $\|\zeta\|_{\psi_1}$, is called the sub-exponential norm of ζ. According to [17, Theorem 3.14], ζ is sub-exponential as per (B.1) if and only if its *moment generating*

[1] Note that there is another widely used but completely different concept of sub-exponentiality in probability theory, which refers to a certain heavy-tail behavior of distributions as detailed in [21,54]. The sub-exponentiality defined by (B.1) is commonly found in areas like machine learning and data analysis (see [17, Section 3.1.2] and [57, Chapter 2]), and is a slight generalization of the pre-Gaussianity defined in [7, Chapter 1].

function $G(t) \equiv E(e^{\zeta t})$ is finite in a neighbourhood of zero (see also [56, inequality (5.16)] and [57, Theorem 2.2]). As elaborated in [56, inequality (5.14)] and [57, Theorem 2.2], assuming sub-exponentiality is also equivalent to requiring that the tail probability $P(|\zeta| > t)$ decays exponentially or faster, which is certainly not a trivial condition. However, it encompasses a large class of distributions that are interesting in practice, including all the distributions with bounded support sets, the normal distribution, Gamma distribution, Weibull distribution, Poisson distribution, geometric distribution, and any Lipschitz continuous functions of random variables following such distributions (see standard statistics textbooks like [40] for the definitions and moment generating functions of the named distributions).

An important property of sub-exponential random variables is a Bernstein-type inequality presented in [56, Proposition 5.16]: if $\zeta_1, \zeta_2, \ldots, \zeta_N$ are independent sub-exponential random variables with zero mean, and $\sigma = \max_{1\le i\le N} \|\zeta_i\|_{\psi_1}$, then

$$\text{Prob}\left(\frac{1}{N}\left|\sum_{i=1}^{N}\zeta_i\right| \ge t\right) \le 2\exp\left(-cN\min\left\{\frac{t^2}{\sigma^2}, \frac{t}{\sigma}\right\}\right) \quad \text{for each} \quad t \ge 0, \tag{B.2}$$

where c is an *absolute constant*. Therefore,

$$\text{Prob}\left(\frac{1}{N}\left|\sum_{i=1}^{N}\zeta_i\right| \ge t\right) \le 2\exp\left(-\frac{cNt^2}{\sigma^2}\right) \quad \text{when} \quad 0 \le t \le \sigma. \tag{B.3}$$

More general forms of inequalities (B.2) and (B.3) can be found in [9, Theorem 1.2.7] and [17, Corollary 3.17]. Note that (B.2) still holds if we replace σ with any number larger than $\max_{1\le i\le N} \|\zeta_i\|_{\psi_1}$, for the right-hand side of (B.2) is decreasing with respect to σ. Consequently, inequality (B.3) is valid as long as

$$\sigma \ge \max_{1\le i\le N} \|\zeta_i\|_{\psi_1}.$$

In Section 3, by saying that the m-dimensional random vector $\phi(\xi, x)$ is sub-exponential, we mean that each entry of $\phi(\xi, x)$ is a sub-exponential random variable (not necessarily independent of each other). Then, according to [56, Remark 5.18], each entry of $F(x) - \phi(\xi, x)$ is a sub-exponential random variable with zero mean, since $F(x) = E[\phi(\xi, x)]$ as stated in (3.2).

Now we give the proof of (3.4).

Proof. Consider an arbitrary point $y \in X$. Let $\Big(F(y) - \phi(\xi, y)\Big)_i$ be the i-th entry of the random vector $F(y) - \phi(\xi, y)$, and

$$\sigma(y) \;=\; 1 + \max_{1\le i\le m} \Big\|\Big(F(y) - \phi(\xi, y)\Big)_i\Big\|_{\psi_1}.$$

For each $i \in \{1, 2, \ldots, m\}$, since $\sigma(y) \ge \Big\|\Big(F(y) - \phi(\xi, y)\Big)_i\Big\|_{\psi_1}$, we can invoke the Bernstein-type bound (B.3) to obtain

$$\text{Prob}\Big(\Big|\Big(F(y) - \tilde{F}^N(y)\Big)_i\Big| \ge t\Big) \;=\; \text{Prob}\left(\frac{1}{N}\left|\sum_{\ell=1}^{N}\Big(F(y) - \phi(\xi_\ell, y)\Big)_i\right| \ge t\right) \;\le\; 2\exp\left[-\frac{cNt^2}{\sigma^2(y)}\right] \tag{B.4}$$

whenever $t \in [0, 1] \subset [0, \sigma(y)]$. Hence

$$\text{Prob}\Big(\|F(y) - \tilde{F}^N(y)\| \ge \epsilon\Big) \;\le\; \text{Prob}\Big(\max_{1\le i\le m}\Big|\Big(F(y) - \tilde{F}^N(y)\Big)_i\Big| \ge \frac{\epsilon}{\sqrt{m}}\Big) \;\le\; 2m\exp\left[-\frac{cN\epsilon^2}{m\sigma^2(y)}\right] \tag{B.5}$$

for each $\epsilon \in [0, 1]$. This gives us the point-wise convergence rate of $\tilde{F}^N$. In the following, we will extend this to obtain an estimation for the uniform convergence rate. The key is to exploit the Lipschitz continuity of ϕ and the boundedness of X.

Since $\phi(\xi, x)$ is L-Lipschitz with respect to x for a constant L independent of ξ, both F and $\tilde{F}^N$ are L-Lipschitz continuous. Let D be the diameter of X, which is finite since X is bounded. Then there exits a set $\{y_j\}_{j=1}^K \subseteq X$ such that

$$K \;\le\; \left\lceil \frac{\sqrt{n}D}{\epsilon/(4L)} \right\rceil^n \quad \text{and} \quad X \subseteq \bigcup_{j=1}^{K} B\left(y_j, \frac{\epsilon}{4L}\right). \tag{B.6}$$

For each $x \in X$, let j_x be an integer in $\{1, 2, \ldots, K\}$ such that $x \in B(y_{j_x}, \epsilon/(4L))$. Then

$$\begin{aligned} \|F(x) - \tilde{F}^N(x)\| &\le \|F(x) - F(y_{j_x})\| + \|\tilde{F}^N(x) - \tilde{F}^N(y_{j_x})\| + \|F(y_{j_x}) - \tilde{F}^N(y_{j_x})\| \\ &\le L\|x - y_{j_x}\| + L\|x - y_{j_x}\| + \max_{1\le j\le K} \|F(y_j) - \tilde{F}^N(y_j)\| \\ &\le \frac{\epsilon}{2} + \max_{1\le j\le K} \|F(y_j) - \tilde{F}^N(y_j)\|. \end{aligned}$$

This, together with (B.5) and (B.6), tells us that

$$\begin{aligned} \text{Prob}\left(\sup_{x\in X} \|F(x) - \tilde{F}^N(x)\| \ge \epsilon\right) &\le \text{Prob}\left(\max_{1\le j\le K} \|F(y_j) - \tilde{F}^N(y_j)\| \ge \frac{\epsilon}{2}\right) \\ &\le \sum_{j=1}^{K} \left\{ 2m \exp\left[-\frac{cN\epsilon^2}{4m\sigma^2(y_j)} \right] \right\} \\ &\le 2m \left\lceil \frac{\sqrt{n}D}{\epsilon/(2L)} \right\rceil^n \exp\left(-c'\, N\epsilon^2\right) \end{aligned}$$

for each $\epsilon \in [0, 1]$, where

$$c' \;\equiv\; \min_{1\le j\le K} \frac{c}{4m\sigma^2(y_j)} \;> 0.$$

Setting $\epsilon = 1/N^p$, we obtain

$$\text{Prob}\left(\sup_{x\in X} \|F(x) - \tilde{F}^N(x)\| \ge \frac{1}{N^p}\right) \;\le\; 2m \left\lceil 2\sqrt{n}LDN^p \right\rceil^n \exp\left(-c' N^{1-2p}\right). \tag{B.7}$$

Given $p \in (0, 1/2)$, we can choose $\bar{N}$ large enough (depending on m, n, L, D, c', and p) such that the right-hand side of (B.7) is at most $1/2$ for each $N \ge \bar{N}$. Consequently,

$$\text{Prob}\left(\sup_{x\in X} \|F(x) - \tilde{F}^N(x)\| \ge \frac{1}{N^p}\right) \;\le\; \frac{1}{2} \tag{B.8}$$

for each $N \ge \bar{N}$. In other words, this $\bar{N}$ fulfills (3.4) together with $\delta = 1/2$ and $c_F = 1$. ∎

Acknowledgement. The authors are grateful to three referees and the associate editor for their very helpful comments and suggestions.

REFERENCES

[1] J. G. Altonji, H. Ichimura, and T. Otsu, *Estimating derivatives in nonseparable models with limited dependent variables*, Econometrica, 80 (2012), pp. 1701–1719.
[2] C. Audet and J. E. Dennis, *Analysis of generalized pattern searches*, SIAM J. Optim., 13 (2003), pp. 889–903.
[3] C. Audet and J. E. Dennis, *Mesh adaptive direct search algorithms for constrained optimization*, SIAM J. Optim., 17 (2006), pp. 188–217.
[4] C. Audet and J. E. Dennis, *A progressive barrier for derivative-free nonlinear programming*, SIAM J. Optim., 20 (2009), pp. 445–472.
[5] J. E. Beasley, *OR-Library: distributing test problems by electronic mail*, J. Oper. Res. Soc., (1990), pp. 1069–1072.
[6] R. W. Blundell and J. L. Powell, *Censored regression quantiles with endogenous regressors*, J. Econometrics, 141 (2007), pp. 65–83.

[7] V. V. Buldygin and I. U. V. Kozachenko, *Metric characterization of random variables and random processes*, American Mathematical Society, Providence, 2000.
[8] J. V. Burke, X. Chen, and H. Sun, *Compute Clarke subgradients of expectation of measurable composite affine max function via smoothing*, 2017. manuscript.
[9] D. Chafaï, O. Guédon, G. Lecué, and A. Pajor, *Interactions between compressed sensing random matrices and high dimensional geometry*, Société Mathématique de France, 2012, http://lecueguillaume.github.io/assets/CSbook.pdf.
[10] X. Chen, *Smoothing methods for nonsmooth, nonconvex minimization*, Math. Program., 134 (2012), pp. 71–99.
[11] X. Chen and C. T. Kelley, *Optimization with hidden constraints and embedded Monte Carlo computations*, Optim. Eng., 17 (2016), pp. 157–175.
[12] X. Chen, T. K. Pong, and R. J.-B. Wets, *Two-stage stochastic variational inequalities: an ERM-solution procedure*, Math. Program., 165 (2017), pp. 71–112.
[13] V. K. Chopra, *Mean-variance revisited: near-optimal portfolios and sensitivity to input variations*, J. Investing, 2 (1993), pp. 51–59.
[14] F. H. Clarke, *Optimization and Nonsmooth Analysis*, no. 5 in Classics in Applied Mathematics, SIAM, Philadelphia, 1990.
[15] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2009.
[16] J. E. Dennis and V. Torczon, *Direct search methods on parallel machines*, SIAM J. Optim., 1 (1991), pp. 448 – 474.
[17] J. Duchi, *Lecture notes for Statistics 311/Electrical Engineering 377: Information theory and statistics*, https://stanford.edu/class/stats311/Lectures/full_notes.pdf.
[18] A. Dvoretzky, J. Kiefer, and J. Wolfowitz, *Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator*, Ann. Math. Statist., 27 (1956), pp. 642–669.
[19] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, CRC Press, New York, 1994.
[20] F. Facchinei and J.-S. Pang, *Finite-dimensional Variational Inequalities and Complementarity Problems*, Springer-Verlag, New York, 2003.
[21] S. Foss, D. Korshunov, and S. Zachary, *An Introduction to Heavy-Tailed and Subexponential Distributions*, Springer, New York, 2011.
[22] G. M. Frankfurter, H. E. Phillips, and J. P. Seagle, *Portfolio selection: the effects of uncertain means, variances, and covariances*, J. Financ. Quantit. Anal., 6 (1971), pp. 1251–1262.
[23] R. Garmanjani and L. N. Vicente, *Smoothing and worst-case complexity for direct-search methods in nonsmooth optimization*, IMA J. Numer. Anal., 33 (2012), pp. 1008–1028.
[24] J. C. Hull, *Options, Futures, and Other Derivatives*, Pearson, New York, 1988.
[25] P. Jorion, *Bayes-Stein estimation for portfolio analysis*, J. Financ. Quantit. Anal., 21 (1986), pp. 279–292.
[26] R. L. Karandikar and M. Vidyasagar, *Rates of uniform convergence of empirical means with mixing processes*, Statist. Probab. Lett., 58 (2002), pp. 297–307.
[27] C. T. Kelley, *Iterative Methods for Optimization*, no. 18 in Frontiers in Applied Mathematics, SIAM, Philadelphia, 1999.
[28] C. T. Kelley, *Implicit Filtering*, no. 23 in Software Environments and Tools, SIAM, Philadelphia, 2011.
[29] S. Kim, R. Pasupathy, and S. G. Henderson, *A guide to sample average approximation*, in Handbook of Simulation Optimization, M. C. Fu, ed., Springer New York, New York, 2015, pp. 207–243.
[30] R. W. Klein and V. S. Bawa, *The effect of estimation risk on optimal portfolio choice*, J. Financ. Econom., 3 (1976), pp. 215–231.
[31] T. G. Kolda, R. M. Lewis, and V. Torczon, *Stationarity results for generating set search for linearly constrained optimization*, SIAM J. Optim., 17 (2006), pp. 943–968.
[32] T. G. Kolda, R. M. Lewis, and V. J. Torczon, *Optimization by direct search: New perspectives on some classical and modern methods*, SIAM Review, 45 (2003), pp. 385–482.
[33] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, Springer-Verlag, New York, 2003.
[34] G. Lan, *An optimal method for stochastic composite optimization*, Math. Program., 133 (2012), pp. 365–397.
[35] Y.-F. Liu, S. Ma, Y.-H. Dai, and S. Zhang, *A smoothing SQP framework for a class of composite l_q minimization over polyhedron*, Math. Program., 158 (2016), pp. 467–500.
[36] H. Markowitz, *Portfolio selection*, J. Finance, 7 (1952), pp. 77–91.
[37] H. Markowitz, *The optimization of a quadratic function subject to linear constraints*, Naval Research Logistics Quarterly, 3 (1956), pp. 111–133.
[38] P. Massart, *The tight constant in the Dvoretzky-Kiefer-Wolfowitz inequality*, Ann. Probab., 18 (1990), pp. 1269–1283.
[39] R. O. Michaud, *The Markowitz optimization enigma: Is optimized optimal?*, Financial Analysts J., 45 (1989), pp. 43–54.
[40] A. M. Mood, *Introduction to the Theory of Statistics.*, McGraw-Hill, New York, 1950.
[41] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, vol. 87 of Applied Optimization, Kluwer Academic Publishers, London, 2004.
[42] Y. Nesterov and V. Spokoiny, *Random gradient-free minimization of convex functions*, Found. Comput. Math., 17 (2017), pp. 527–566.
[43] J. Nocedal and S. Wright, *Numerical Optimization*, Springer Series in Operations Research, Springer, New York, second ed., 2006.
[44] J.-S. Pang, *A new and efficient algorithm for a class of portfolio selection problems*, Oper. Res., 28 (1980), pp. 754–767.
[45] R. Pasupathy, *On choosing parameters in retrospective-approximation algorithms for stochastic root finding and simulation optimization*, Oper. Res., 58 (2010), pp. 889–901.
[46] R. Pasupathy, P. W. Glynn, G. S. G., and F. S. Hahemi, *On sampling rates in simulation-based recursions*, Optimiza-

tion Online, (2017).
[47] M. Porcelli and P. L. Toint, *BFO, a trainable derivative-free Brute Force Optimizer for nonlinear bound-constrained optimization and equilibrium computations with continuous and discrete variables*, ACM Trans. Math. Software, (to appear).
[48] S. Sankaran, C. Audet, and A. L. Marsden, *A method for stochastic constrained optimization using derivative-free surrogate pattern search and collocation*, Journal of Computational Physics, 229 (2010), pp. 4664–4682, doi:10.1016/j.jcp.2010.03.005, http://linkinghub.elsevier.com/retrieve/pii/S0021999110001154.
[49] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on Stochastic Programming*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2009.
[50] W. F. Sharpe, *The Sharpe ratio*, J. Portfolio Management, 21 (1994), pp. 49–58.
[51] R. J. Smith and R. W. Blundell, *An exogeneity test for a simultaneous equation Tobit model with an application to labor supply*, Econometrica, 54 (1986), pp. 679–685.
[52] T. A. Sriver, J. W. Chrissis, and M. A. Abramson, *Pattern search ranking and selection algorithms for mixed variable simulation-based optimization*, European J. Oper. Res., 198 (2009), pp. 878–890.
[53] L. Taylor and T. Otsu, *Estimation of nonseparable models with censored dependent variables and endogenous regressors*, Econom. Rev., (to appear).
[54] J. L. Teugels, *The class of subexponential distributions*, Ann. Probab., (1975), pp. 1000–1011.
[55] A. Toth, J. A. Ellis, T. Evans, S. Hamilton, C. T. Kelley, R. Pawlowski, and S. Slattery, *Local improvement results for Anderson acceleration with inaccurate function evaluations*, SIAM J. Sci. Comp., 39 (2017), pp. S47–S65, doi:10.1137/16M1080677.
[56] R. Vershynin, *Introduction to the non-asymptotic analysis of random matrices*, in Compressed Sensing: Theory and Applications, Y. C. Eldar and G. Kutyniok, eds., Cambridge University Press, Cambridge, 2012, pp. 210–268.
[57] M. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*, 2017, https://www.stat.berkeley.edu/~mjwain/stat210b/Chap2_TailBounds_Jan22_2015.pdf.
[58] J. Willert, X. Chen, and C. T. Kelley, *Newton's method for Monte Carlo-based residuals*, SIAM J. Numer. Anal., 53 (2015), pp. 1738–1757, doi:10.1137/130905691.
[59] B. Yu, *Rates of convergence for empirical processes of stationary mixing sequences*, Ann. Probab., 22 (1994), pp. 94–116.