Gradient Compressed Sensing: A Query-Efficient Gradient Estimator for High-Dimensional Zeroth-Order Optimization

Ruizhong Qiu [1] **Hanghang Tong** [1]

Abstract

We study nonconvex zeroth-order optimization (ZOO) in a high-dimensional space $\mathbb{R}^d$ for functions with approximately s-sparse gradients. To reduce the dependence on the dimensionality d in the query complexity, high-dimensional ZOO methods seek to leverage gradient sparsity to design gradient estimators. The previous best method needs $O\big(s \log \frac{d}{s}\big)$ queries per step to achieve $O\big(\frac{1}{T}\big)$ rate of convergence w.r.t. the number T of steps. In this paper, we propose Gradient Compressed Sensing (GraCe), a query-efficient and accurate estimator for sparse gradients that uses only $O\big(s \log\log \frac{d}{s}\big)$ queries per step and still achieves $O\big(\frac{1}{T}\big)$ rate of convergence. To our best knowledge, we are the first to achieve a *double-logarithmic* dependence on d in the query complexity, and our proof uses weaker assumptions than previous work. Our proposed GraCe generalizes the Indyk–Price–Woodruff (IPW) algorithm in compressed sensing from linear measurements to nonlinear functions. Furthermore, since the IPW algorithm is purely theoretical due to its impractically large constant, we improve the IPW algorithm via our *dependent random partition* technique together with our corresponding novel analysis and successfully reduce the constant by a factor of nearly 4300. Our GraCe is not only theoretically query-efficient but also achieves strong empirical performance. We benchmark our GraCe against 12 existing ZOO methods with 10000-dimensional functions and demonstrate that GraCe significantly outperforms existing methods. Our code is publicly available at https://github.com/q-rz/ICML24-GraCe.

[1]Department of Computer Science, University of Illinois Urbana–Champaign, USA. Correspondence to: Ruizhong Qiu <rq5@illinois.edu>, Hanghang Tong <htong@illinois.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

1. Introduction

We study the problem of unconstrained optimization:

$$\min_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x}), \tag{1}$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is a (possibly nonconvex) function over a high-dimensional space $\mathbb{R}^d$.

Gradient-free optimization (GFO), also known as black-box optimization, was among the first schemes explored in the history of optimization theory (Matyáš, 1965). In GFO, the function f is unknown to the optimizer, and the optimizer can obtain information about the function f only via *queries* (i.e., function evaluations). The goal of GFO is to optimize the function f using a minimal number of queries.

Zeroth-order optimization (ZOO), a paradigm of GFO, aims to apply first-order optimization methods to GFO with gradients estimated from queries. Through a long history of study, various *full-gradient* ZOO methods have been proposed. Early full-gradient ZOO methods such as the Kiefer–Wolfowitz method (Kiefer & Wolfowitz, 1952) use dimension-wise finite difference to approximate gradients, which suffers from an $O(d)$ dependence in the query complexity. Later methods achieve $O(1)$ queries per step via stochastic gradient estimators such as Gaussian smoothing (Nesterov & Spokoiny, 2015), but they suffer from a $\mathrm{poly}(d)$ factor in their rates of convergence. Hence, their overall query complexity still depends polynomially on d.

Meanwhile, the dimensionality d can be very large in modern real-world applications. For instance, a high-resolution image can have millions of pixels. These real-world scenarios call for *high*-dimensional ZOO. High-dimensional ZOO aims to develop gradient estimators with minimal dependence on the dimensionality d in the query complexity under gradient sparsity assumptions (Wang et al., 2018; Cai et al., 2022). In contrast to full-gradient ZOO, the research on *sparse-gradient* ZOO is still in its infancy. Existing methods suffer from slow convergence and/or a suboptimal query complexity. For example, Wang et al. (2018) proposed a LASSO-based method that uses $O(s^{2/3}\sqrt{T})$ queries per step and has $O\big(\frac{s^2\sqrt{\log d}}{T^{1/3}}\big) + \widetilde{O}\big(\frac{1}{T^{5/12}}\big)$ rate of convergence for stochastic convex ZOO. For nonconvex ZOO, the previous best method ZORO (Cai et al., 2022), which is based

Table 1. Comparison in nonconvex ZOO. (q, b: hyperparameters; $\tau := \arg\min_{t=1,\dots,T} \|\nabla f(\boldsymbol{x}_t)\|_2$; w.h.p.: with high probability.) To our best knowledge, we are the first to achieve a *double-logarithmic* dependence on d in the query complexity under weaker assumptions.

Type	Method	Queries per step	Rate of convergence
Full Gradient	RS (Ghadimi & Lan, 2012)	$O(1)$	$\mathbb{E}[\|\nabla f(\boldsymbol{x}_\tau)\|_2^2] \le O\big(\frac{\sqrt{d}}{\sqrt{T}} + \frac{d}{T}\big)$
	TPGE (Duchi et al., 2015)	$O(1)$	$\mathbb{E}[\|\nabla f(\boldsymbol{x}_\tau)\|_2^2] \le O\big(\frac{\sqrt{d}}{\sqrt{T}}\big)$
	RSPG (Ghadimi et al., 2016)	$O(q)$	$\mathbb{E}[\|\nabla f(\boldsymbol{x}_\tau)\|_2^2] \le O\big(\frac{d}{q} + \frac{d^2}{qT}\big)$
	ZO-signSGD (Liu et al., 2019)	$O(bq)$	$\mathbb{E}[\|\nabla f(\boldsymbol{x}_\tau)\|_2] \le O\big(\frac{\sqrt{d}\sqrt{q+d}}{\sqrt{bq}} + \frac{\sqrt{d}}{\sqrt{T}}\big)$
	ZO-AdaMM (Chen et al., 2019)	$O(1)$	$\mathbb{E}[\|\nabla f(\boldsymbol{x}_\tau)\|_2^2] \le O\big(\frac{d}{\sqrt{T}} + \frac{d^2}{T}\big)$
Sparse Gradient	ZORO (Cai et al., 2022)	$O\big(s \log \frac{d}{s}\big)$	$\|\nabla f(\boldsymbol{x}_\tau)\|_2^2 \le O\big(\frac{1}{T}\big)$ w.h.p.
	GraCe (ours)	$O\big(s \log\log \frac{d}{s}\big)$	$\|\nabla f(\boldsymbol{x}_\tau)\|_2^2 \le O\big(\frac{1}{T}\big)$ w.h.p.

on CoSaMP (Needell & Tropp, 2009), needs $O\big(s \log \frac{d}{s}\big)$ queries per step to achieve $O\big(\frac{1}{T}\big)$ rate[1] of convergence. The root cause of the technical difficulty here lies in the inaccurate gradient estimation in existing methods.

In this paper, we propose *Gradient Compressed Sensing* (GraCe), a new sparse gradient estimator that uses only $O\big(s \log\log \frac{d}{s}\big)$ queries per step and still achieves $O\big(\frac{1}{T}\big)$ rate of convergence for nonconvex ZOO. It generalizes the Indyk–Price–Woodruff (IPW) algorithm (Indyk et al., 2011) in compressed sensing from linear measurements to nonlinear functions. Our main contributions are as follows:

- **Query-efficient gradient estimator.** We propose *Gradient Compressed Sensing* (GraCe), a new gradient estimator that uses only $O\big(s \log\log \frac{d}{s}\big)$ queries per step and still achieves $O\big(\frac{1}{T}\big)$ rate of convergence for nonconvex ZOO. To our best knowledge, we are the first to achieve a *double-logarithmic* dependence on d in the query complexity (see Table 1).
- **Relaxed sparsity assumption.** Our analysis is based on a new assumption of approximate gradient sparsity, which is weaker than previous assumptions — *exact sparsity* (Wang et al., 2018) and *compressibility* (Cai et al., 2022).
- **Improvement of the IPW algorithm.** The IPW algorithm is purely theoretical due to its impractically large constant. To make the IPW algorithm practical, we improve the IPW algorithm via our *dependent random partition* technique together with our corresponding novel analysis and successfully reduce the constant by a factor of nearly 4300.
- **Strong empirical performance.** Our GraCe is not only theoretically query-efficient but also achieves strong empirical performance. We benchmark our GraCe against 12 existing ZOO methods with 10000-dimensional functions and demonstrate that GraCe significantly outperforms existing methods.

[1]This rate is for non-stochastic nonconvex ZOO. For stochastic nonconvex ZOO, ZORO has $O\big(1 + \frac{1}{\sqrt{T}}\big)$ rate of convergence.

2. Preliminaries

2.1. Notation

Throughout the paper, we use the bold font for vectors (e.g., $\boldsymbol{x}$) and the italic font for scalars (e.g., x_i). We use the same alphabet for a vector and its entries.

For $i \in [d]$, let $\boldsymbol{e}_i := [1_{[i'=i]}]_{i' \in [d]} \in \mathbb{R}^d$ denote the i-th standard basis of $\mathbb{R}^d$. For vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$, let $\langle \boldsymbol{u}, \boldsymbol{v} \rangle := \boldsymbol{u}^\mathsf{T} \boldsymbol{v}$ denote the standard inner product. For a vector $\boldsymbol{u} \in \mathbb{R}^d$, let $\|\boldsymbol{u}\|_2 := \sqrt{\langle \boldsymbol{u}, \boldsymbol{u} \rangle}$ denote the Euclidean norm.

For a dimension $i \in [d]$, let ∇_i denote the partial derivative operator w.r.t. the dimension i. For a subset $S \subseteq [d]$ of dimensions, let $\nabla_S := [\nabla_i]_{i \in S}$ denote the partial derivative operator (as a column vector) w.r.t. dimensions S. Let $\nabla := \nabla_{[d]}$ denote the gradient operator.

For two finite sets A, B with $|A| = |B|$, let $\mathcal{P}_{A \to B}$ denote the set of bijections from A to B. For instance, $\mathcal{P}_{[d] \to [d]}$ is the set of permutations over $[d]$. For a finite set A, let $\mathsf{Unif}(A)$ denote the uniform distribution over A.

2.2. Assumptions

We first introduce our new assumption on approximate gradient sparsity.

Assumption A (Approximate gradient sparsity)**.** *The function f has ρ-approximately s-sparse gradients ($0 < \rho \le 1$, $1 \le s \le d$):*

$$\max_{I \subseteq [d] \,:\, |I| = s} \|\nabla_I f(\boldsymbol{x})\|_2^2 \ge \rho \|\nabla f(\boldsymbol{x})\|_2^2, \quad \forall \boldsymbol{x}.$$

Our Assumption A is weaker than previous assumptions on gradient sparsity. The *exact sparsity* assumption (i.e., $\|\nabla f(\boldsymbol{x})\|_0 \le s$) in Wang et al. (2018) corresponds to $\rho = 1$

in our Assumption A. The *compressibility* assumption (i.e., $\exists \kappa > 1$ s.t. the i-th largest magnitude in $\nabla f(\boldsymbol{x})$ is at most $i^{-\kappa}\|\nabla f(\boldsymbol{x})\|_2, \forall i \in [d]$) in Cai et al. (2022) assumes the distribution of the entries of $\nabla f(\boldsymbol{x})$ while our Assumption A does not assume the distribution; also, it implies our Assumption A with $\rho = 1 - \frac{1}{(2\kappa-1)s^{2\kappa-1}}$. Hence, our Assumption A is a relaxation of existing assumptions.

In addition to approximate gradient sparsity, we make the following standard assumptions on the function f.

Assumption B (Lower boundedness)**.** *The function f is lower-bounded:*

$$f_* := \inf_{\boldsymbol{x}} f(\boldsymbol{x}) > -\infty.$$

Assumption C (Lipschitz continuity)**.** *The function f is L_0-Lipschitz continuous:*

$$|f(\boldsymbol{x}+\boldsymbol{u}) - f(\boldsymbol{x})| \le L_0\|\boldsymbol{u}\|_2, \quad \forall (\boldsymbol{x}, \boldsymbol{u}).$$

Assumption D (Lipschitz smoothness)**.** *The function f is differentiable and L_1-Lipschitz smooth:*

$$\|\nabla f(\boldsymbol{x}+\boldsymbol{u}) - \nabla f(\boldsymbol{x})\|_2 \le L_1\|\boldsymbol{u}\|_2, \quad \forall (\boldsymbol{x}, \boldsymbol{u}).$$

3. GraCe: Gradient Compressed Sensing

In this section, we first propose a query-efficient method, Gradient Compressed Sensing (GraCe), for estimating ρ-approximately s-sparse gradients using only $O\big(s \log\log \frac{d}{s}\big)$ adaptive queries.

Our GraCe generalizes the Indyk–Price–Woodruff (IPW) algorithm (Indyk et al., 2011) in compressed sensing from linear measurements to nonlinear functions. First, GraCe randomly partitions the d dimensions into $O(s)$ groups of size $O\big(\frac{d}{s}\big)$ so that (with high probability) each group has at most one large-gradient dimension. Then for each group, to locate the large-gradient dimension, GraCe constructs adaptive queries to iteratively shrink the candidate set of dimensions and finds the large-gradient dimension after $O\big(\log\log \frac{d}{s}\big)$ iterations (with high probability). The procedure of GraCe is presented in Algorithm 1.

In the rest of this section, Section 3.1 introduces how to design adaptive queries to locate the large-gradient dimension in a group, and Section 3.2 describes how to divide the groups to achieve accurate gradient estimation with high probability. Proofs are deferred to Appendix A.

3.1. Base case: Approximately 1-sparse gradient

Suppose that we have a candidate group $S \subseteq [d]$ in which there is only one dimension $j \in S$ with a large gradient $|\nabla_j f(\boldsymbol{x})|$. We will introduce how to find j with a small number of adaptive queries.

Algorithm 1 Gradient Compressed Sensing (GraCe)

Input: point $\boldsymbol{x}$; sparsity s; finite difference ϵ; number m of repeats; group size n; division schedule $\{D_r\}_{r\ge 1}$
Output: the gradient estimate $\boldsymbol{g} \in \mathbb{R}^d$

candidate set $J \leftarrow \varnothing$
for $l = 1$ **to** m **do**
  random permutation $\omega \sim \mathsf{Unif}(\mathcal{P}_{[d]\to[d]})$
  for $k = 1$ **to** $\lceil d/n \rceil$ **do**
    candidate group $S \leftarrow \big\{i \in [d] : \big\lceil \frac{\omega(i)}{n} \big\rceil = k\big\}$
    iteration number $r \leftarrow 0$
    repeat
      iteration number $r \leftarrow r + 1$
      random permutation $\varpi \sim \mathsf{Unif}(\mathcal{P}_{S\to[|S|]})$
      block size $B \leftarrow \big\lceil \frac{|S|}{D_r} \big\rceil$
      perturbations $\boldsymbol{u} \leftarrow \mathbf{0}_d$, $\boldsymbol{v} \leftarrow \mathbf{0}_d$
      for $i \in S$ **do**
        random sign $\sigma_i \sim \mathsf{Unif}(\{\pm 1\})$
        block label $h_i \leftarrow \big\lceil \frac{\varpi(i)}{B} \big\rceil$
        perturbations $u_i \leftarrow \epsilon \cdot \sigma_i$, $v_i \leftarrow \epsilon \cdot \sigma_i \cdot h_i$
      end for
      target $q \leftarrow \mathrm{round}\big(\frac{f(\boldsymbol{x}+\boldsymbol{v})-f(\boldsymbol{x})}{f(\boldsymbol{x}+\boldsymbol{u})-f(\boldsymbol{x})}\big)$ via 2 queries
      candidate group $S \leftarrow \{i \in S : h_i = q\}$
    until $|S| \le 2$
    candidate set $J \leftarrow J \cup S$
  end for
end for
gradient estimate $\boldsymbol{g} \leftarrow \mathbf{0}_d$
for $j \in J$ **do**
  finite difference $g_j \leftarrow \frac{f(\boldsymbol{x}+\epsilon \boldsymbol{e}_j)-f(\boldsymbol{x})}{\epsilon}$ via 1 query
end for
return gradient estimate $\boldsymbol{g}$

First, consider a motivating case: the *signal-to-noise ratio* (SNR) $\frac{|\nabla_j f(\boldsymbol{x})|}{\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2}$ is sufficiently large. Then, one can use an idea in Ba et al. (2010) to encode dimension information into queries. Given a small $\epsilon > 0$, define perturbations $\boldsymbol{u}', \boldsymbol{v}' \in \mathbb{R}^d$ by

$$u'_i := \epsilon \cdot 1_{[i\in S]}, \quad v'_i := \epsilon \cdot i \cdot 1_{[i\in S]}, \quad i \in [d]. \tag{2}$$

With a sufficiently large SNR $\frac{|\nabla_j f(\boldsymbol{x})|}{\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2}$,

$$\begin{aligned} \frac{f(\boldsymbol{x}+\boldsymbol{v}') - f(\boldsymbol{x})}{f(\boldsymbol{x}+\boldsymbol{u}') - f(\boldsymbol{x})} &\approx \frac{\sum_{i\in[d]} v'_i \cdot \nabla_i f(\boldsymbol{x})}{\sum_{i\in[d]} u'_i \cdot \nabla_i f(\boldsymbol{x})} \\ = \frac{\sum_{i\in S} \epsilon \cdot i \cdot \nabla_i f(\boldsymbol{x})}{\sum_{i\in S} \epsilon \cdot \nabla_i f(\boldsymbol{x})} &\approx \frac{\epsilon \cdot j \cdot \nabla_j f(\boldsymbol{x})}{\epsilon \cdot \nabla_j f(\boldsymbol{x})} = j. \end{aligned} \tag{3}$$

In this case, we can find j using only $O(1)$ queries by rounding $\frac{f(\boldsymbol{x}+\boldsymbol{v}')-f(\boldsymbol{x})}{f(\boldsymbol{x}+\boldsymbol{u}')-f(\boldsymbol{x})}$ to the nearest integer.

In general, however, it can happen that the SNR is not sufficiently large. To address this issue, the idea is iteratively

increasing the SNR by identifying small-gradient dimensions and removing them from the candidate group S. As an improvement over the IPW algorithm, we introduce a technique that we call *dependent random partition* for increasing the SNR: given a parameter D, we randomly divide S into blocks of a fixed size $B := \lceil \frac{|S|}{D} \rceil$ and label the blocks $1, \dots, \lceil \frac{|S|}{B} \rceil$. For each dimension $i \in S$, let h_i denote the label of the block that i belongs to. Each $i \in S$ is also assigned a random sign $\sigma_i \sim \mathsf{Unif}\{\pm 1\}$. Then, define perturbations $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$ by

$$u_i := \epsilon \cdot \sigma_i \cdot 1_{[i \in S]},\ v_i := \epsilon \cdot \sigma_i \cdot h_i \cdot 1_{[i \in S]},\ i \in [d]. \quad (4)$$

Intuitively, since all dimensions in the same block have the same label, then the "signal" of the label h_j should be strengthened. Furthermore, although the labels $\{h_i\}$ are not mutually independent, their dependence is weakened by the random signs $\{\sigma_i\}$. Thus, the "noises" $h_i \neq h_j$ would not be strengthened. Hence, under suitable conditions,

$$\frac{f(\boldsymbol{x}+\boldsymbol{v}) - f(\boldsymbol{x})}{f(\boldsymbol{x}+\boldsymbol{u}) - f(\boldsymbol{x})} \approx \frac{\sum_{i\in[d]} v_i \cdot \nabla_i f(\boldsymbol{x})}{\sum_{i\in[d]} u_i \cdot \nabla_i f(\boldsymbol{x})} \quad (5)$$

$$= \frac{\sum_{i\in S} \epsilon\sigma_i h_i \cdot \nabla_i f(\boldsymbol{x})}{\sum_{i\in S} \epsilon\sigma_i \cdot \nabla_i f(\boldsymbol{x})} \approx \frac{\sum_{\substack{i\in S \\ h_i = h_j}} \epsilon\sigma_i h_i \cdot \nabla_i f(\boldsymbol{x})}{\sum_{\substack{i\in S \\ h_i = h_j}} \epsilon\sigma_i \cdot \nabla_i f(\boldsymbol{x})} = h_j.$$

Once we obtain h_j, we can shrink the candidate group to $S' := \{i \in S : h_i = h_j\}$, which has an increased SNR. This is formally stated in Lemma 3.1.

Lemma 3.1. *There is an absolute constant $C_1 > 0$ such that given $\boldsymbol{x} \in \mathbb{R}^d$, $\epsilon > 0$, $S \subseteq [d]$, $0 < \delta_1, \delta_2 < 1$, and integer $2 \le D \le d$, if there exists $j \in S$ with $|\nabla_j f(\boldsymbol{x})| >$*

$$\big(C_1 D + \tfrac{1}{D}\big)\sqrt{2\ln\tfrac{3}{\delta_1}}\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon, \quad (6)$$

then using $O(1)$ queries, with probability $\ge 1 - (\delta_1 + \delta_2)$, we can find a subset $S' \subseteq S$ with $j \in S'$ and

$$|S' \setminus \{j\}| \le \frac{|S \setminus \{j\}|}{D}, \quad (7)$$

$$\|\nabla_{S'\setminus\{j\}} f(\boldsymbol{x})\|_2 \le \frac{\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2}{\sqrt{D\delta_2}}. \quad (8)$$

Here, $\lambda_{1,n} := L_1\big(d^2 + d + \frac{1}{2}\big)n$.

Eq. (6) quantifies the condition between the SNR and the division parameter D; Eq. (7) shows that the size of the candidate group shrinks by D times; and Eq. (8) shows that the SNR $\frac{|\nabla_j f(\boldsymbol{x})|}{\|\nabla_{S'\setminus\{j\}} f(\boldsymbol{x})\|_2}$ increases by a factor of $\sqrt{D\delta_2}$. Lemma 3.1 will be used next as the key subroutine in Lemma 3.2.

Since the SNR has increased, we can repeat Lemma 3.1 with a larger division parameter D for the next iteration. Let D_r denote the division parameter for the r-th iteration. With the help of our *dependent random partition* technique, the size of the candidate group shrinks rapidly. With the candidate set S shrinking and the division parameter increasing, we shall have $|S\setminus\{j\}| < D_r$ at some iteration. Then by Eq. (7),

$$|S' \setminus \{j\}| \le \frac{|S \setminus \{j\}|}{D_r} < 1, \quad (9)$$

which implies that S' contains j only, i.e., j is found.

It remains to bound the number of iterations. This depends on the growth rate of D_r. We show in Lemma 3.2 that D_r can grow rapidly so that $O(\log\log|S|)$ iterations suffice.

Lemma 3.2. *There exist absolute constants $C_2, C_3 > 0$, $A > 1$, and a division schedule $\{D_r\}_{r\ge1}$ such that (i) $D_r \ge C_3 A^{(3/2)^{r-1}}$, and (ii) given $\boldsymbol{x} \in \mathbb{R}^d$, $\epsilon > 0$, and $S \subseteq [d]$, if there exists $j \in S$ such that*

$$|\nabla_j f(\boldsymbol{x})| > C_2\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon, \quad (10)$$

then $O(\log_{3/2}\log_A|S|)$ iterations of Lemma 3.1 with parameters $\{D_r\}_{r\ge1}$ can find j with probability at least $1/2$.

We provide the general version of Lemma 3.2 in Lemma A.4, which gives the exact relation between the failure probability and the absolute constants. Besides that, we recommend choosing division parameters via the recurrence $D_{r+1} := \lfloor D_r^{3/2} \rfloor$ with an appropriate D_1 in practice. Note that the subset S in Lemma 3.2 is different from the subset I in Assumption A. We will show how to find such subsets S in Section 3.2.

We remark that our constant $C_2 \approx 135$ is nearly 4300 times smaller than the corresponding constant $C_2' \approx 579263$ of the IPW algorithm. This is owing to our *dependent random partition* technique and our corresponding novel analysis. In contrast to the purely theoretical IPW algorithm, our GraCe achieves strong empirical performance, which is demonstrated by our experiments in Section 5.

3.2. General case: Approximately s-sparse gradient

Building upon the base case, next we describe how to partition the d dimensions into groups so that most groups satisfy the condition Eq. (10) in Lemma 3.2.

Here we employ again our aforementioned technique *dependent random partition*: given a parameter n, we randomly partition the d dimensions into groups of a fixed size n. It remains to determine the group size n. On the one hand, the condition Eq. (10) requires a group to have an SNR greater than an absolute constant C_2. This means that $S \setminus \{j\}$ should not contain too many dimensions, so the group size n should not be too large. On the other hand, the group size n should not be too small. Otherwise, the number of groups would be too large, resulting in a large number of queries.

For example, if $S \setminus \{j\} = \varnothing$, the SNR would be ∞, but the overall query complexity would be $\Omega(d)$. We will show in Lemma 3.3 that $n = \Theta\big(\frac{d}{s}\big)$ suffices, and we use repetition to ensure success with high probability.

Combining the dimensions j found in each group gives a candidate set $J \subseteq [d]$. We show in Lemma 3.3 that J is likely to contain most large-gradient dimensions.

Lemma 3.3. *Given $\boldsymbol{x} \in \mathbb{R}^d$, $\epsilon > 0$, $0 < \alpha < \rho$, and $0 < \delta < 1$, there exist hyperparameters for Algorithm 1 such that with probability at least $1-\delta$, it can use $O\big(s \log\log \frac{d}{s}\big)$ adaptive queries to find a set $J \subseteq [d]$ of size $O(s)$ such that*

$$\|\nabla_J f(\boldsymbol{x})\|_2^2 \geq \alpha \|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2 \epsilon^2, \tag{11}$$

where $\lambda_{2,d} := 2L_0\lambda_{1,d}$. The O notation hides constants that depend only on ρ, α, and δ.

Finally, for each candidate dimension $j \in J$, we estimate the gradient $\nabla_j f(\boldsymbol{x})$ via finite difference:

$$g_j := \frac{f(\boldsymbol{x} + \epsilon \boldsymbol{e}_j) - f(\boldsymbol{x})}{\epsilon}. \tag{12}$$

With the good candidate set J, we show in Theorem 3.4 that the direction of the gradient estimate $\boldsymbol{g}$ aligns well with that of the true gradient $\nabla f(\boldsymbol{x})$.

Theorem 3.4. *Given $\boldsymbol{x} \in \mathbb{R}^d$, $\epsilon > 0$, and $0 < \alpha < \rho$, there exist hyperparameters for Algorithm 1 such that it can use $O\big(s \log\log \frac{d}{s}\big)$ adaptive queries to find a gradient estimate $\boldsymbol{g} \in \mathbb{R}^d$ such that with probability 1,*

$$\begin{aligned} \|\boldsymbol{g}\|_2 &\leq \|\nabla f(\boldsymbol{x})\|_2 + O(L_1\sqrt{s}\epsilon), \\ \mathbb{E}[\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle \mid \boldsymbol{x}] &\geq \alpha \|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{3,d,s}\epsilon - \lambda_{4,d}\epsilon^2, \end{aligned} \tag{13}$$

where $\lambda_{3,d,s} = O(\lambda_{2,d} + L_0 L_1 \sqrt{s})$, $\lambda_{4,d} = O(\lambda_{1,d}^2)$. The O notation hides constants that depend only on ρ and α.

Theorem 3.4 shows that the inner product $\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle$ is relatively large, and Eq. (13) shows that it is not due to an unbounded norm $\|\boldsymbol{g}\|_2$. Together, we can conclude that the gradient estimate $\boldsymbol{g}$ has high cosine similarity with the true gradient $\nabla f(\boldsymbol{x})$. This property will be useful in improving the rate of convergence in nonconvex ZOO.

4. Zeroth-Order Optimization with GraCe

As a zeroth-order gradient estimator, GraCe can be applied to ZOO by integrating the estimated gradient into existing first-order methods. In this work, we consider zeroth-order gradient descent with GraCe.

Let $\boldsymbol{x}_1 \in \mathbb{R}^d$ denote the initial point, let $\eta > 0$ denote the step size, and let $\{\epsilon_t\}_{t\geq 1}$ denote the finite difference schedule. At each iteration $t \geq 1$, the algorithm finds a gradient estimate $\boldsymbol{g}_t \in \mathbb{R}^d$ using GraCe with $(\boldsymbol{x}_t, \epsilon_t)$ and performs a gradient descent step:

$$\boldsymbol{x}_{t+1} \leftarrow \boldsymbol{x}_t - \eta_t \boldsymbol{g}_t. \tag{14}$$

The overall procedure is presented in Algorithm 2.

Algorithm 2 Zeroth-order gradient descent with GraCe

Input: initial point $\boldsymbol{x}_1$; step size η; finite difference schedule $\{\epsilon_t\}_{t\geq 1}$; hyperparameters for GraCe
Output: optimized point

```
step number t ← 1
repeat
    gradient estimate g_t via GraCe with (x_t, ε_t)
    next point x_{t+1} ← x_t − η g_t
    step number t ← t + 1
until stopping criterion is met
return arg min_{x ∈ {x_1, ..., x_t}} f(x)
```

Next, we analyze the rate of convergence of Algorithm 2. With the help of the accurate gradient estimation by GraCe, Algorithm 2 achieves an $O\big(\frac{1}{T}\big)$ rate of convergence for finding a first-order stationary point in nonconvex ZOO. A comparison of nonconvex bounds is summarized in Table 1.

Theorem 4.1. *Given any initial point $\boldsymbol{x}_1 \in \mathbb{R}^d$ and any $\Delta > 0$, there exist a step size η, a finite difference schedule $\{\epsilon_t\}_{t\geq 1}$ for Algorithm 2, and hyperparameters for GraCe such that for every $T \geq 1$,*

$$\mathbb{E}\Big[\min_{t=1,\dots,T} \|\nabla f(\boldsymbol{x}_t)\|_2^2\Big] \leq \frac{\frac{2L_1}{\rho^2}(f(\boldsymbol{x}_1) - f_*) + \Delta}{T}. \tag{15}$$

The proof of Theorem 4.1 is owing to Theorem 3.4, which enables us to show a constant upper bound of the cumulative regret $\mathbb{E}[\sum_{t=1}^T \|\nabla f(\boldsymbol{x}_t)\|_2^2]$. Furthermore, we also provide a high-probability bound of convergence.

Theorem 4.2. *Given any initial point $\boldsymbol{x}_1 \in \mathbb{R}^d$, any step size $0 < \eta < \frac{\rho}{L_1}$, any $0 < \beta < 1$, and any $\Delta > 0$, there exist a finite difference schedule $\{\epsilon_t\}_{t\geq 1}$ for Algorithm 2 and hyperparameters for GraCe such that with probability at least $1-\beta$, for all $T \geq 1$ simultaneously,*

$$\min_{t=1,\dots,T} \|\nabla f(\boldsymbol{x}_t)\|_2^2 \leq \frac{\frac{1 + \frac{2(1-L_1\eta)}{L_1\eta\beta}}{\eta - \frac{L_1\eta^2}{2}}(f(\boldsymbol{x}_1) - f_*) + \Delta}{T}. \tag{16}$$

In practice, we recommend using a constant ϵ for all t in order to avoid underflow in floating point arithmetics. Our experiments demonstrate that a constant ϵ still works well.

5. Experiments

To demonstrate the empirical competence of our GraCe, we compare it with 12 strong baselines on three challenging functions. In the rest of the section, we introduce

Table 2. Comparison among ZOO methods (mean ± s.e.).

Type	Method	DISTANCE	MAGNITUDE	ATTACK
Full Gradient	RS	0.66326 ± 0.00780	0.91847 ± 0.00140	0.41310 ± 0.00048
	TPGE	0.75618 ± 0.00680	0.96111 ± 0.00110	0.42757 ± 0.00310
	RSPG	0.47299 ± 0.00994	0.69877 ± 0.00228	0.46512 ± 0.00115
	ZO-signSGD	0.93413 ± 0.00823	0.98787 ± 0.00024	0.81652 ± 0.00046
	ZO-AdaMM	0.80454 ± 0.01442	0.97235 ± 0.00076	0.59624 ± 0.00747
	GLD	0.85677 ± 0.00436	0.98267 ± 0.00074	0.85497 ± 0.00172
Sparse Gradient	LASSO	0.47432 ± 0.00873	0.70524 ± 0.00343	0.33776 ± 0.00027
	SparseSZO	0.27062 ± 0.00994	0.09523 ± 0.00277	0.45858 ± 0.00151
	TruncZSGD	0.18022 ± 0.01223	0.14323 ± 0.01869	0.99149 ± 0.00214
	ZORO	0.51254 ± 0.06313	0.02534 ± 0.00188	0.99998 ± 0.00001
	ZO-BCD	0.00708 ± 0.00256	0.02759 ± 0.01988	0.99994 ± 0.00003
	SZOHT	0.49686 ± 0.03160	0.12000 ± 0.09466	0.33883 ± 0.00554
	GraCe (ours)	$\mathbf{0.00508} \pm 0.00242$	$\mathbf{0.00449} \pm 0.00005$	$\mathbf{0.32381} \pm 0.00097$

our benchmark functions in Section 5.1, describe baselines and implementation details in Section 5.2, and discuss the results in Section 5.3. The results are presented in Table 2 and Figures 1 & 2. Our code is publicly available at https://github.com/q-rz/ICML24-GraCe.

5.1. Benchmark Functions

To demonstrate the empirical competence of our GraCe in high-dimensional ZOO, we consider two challenging synthetic functions in $d = 10,000$ dimensions and a real-world task in $d = 13,225$ dimensions. For each synthetic benchmark function, we randomly generate 10 instantiations.

- DISTANCE: $f(\boldsymbol{x}) := (\boldsymbol{x} - \boldsymbol{x}_*)^\top \boldsymbol{W}(\boldsymbol{x} - \boldsymbol{x}_*)$, where $\boldsymbol{v} \in \mathbb{R}^d$ is an s-sparse vector, and $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ is a diagonal matrix. We randomly sample a subset $S \subseteq [d]$ of size s as the nonzero dimensions of $\boldsymbol{x}_*$, and we generate the nonzero entries in $\boldsymbol{x}_*$ and $\boldsymbol{W}$ from $\mathsf{Unif}(0, 1)$. We use $s = 10$ and initial point $\boldsymbol{x}_1 = \boldsymbol{0}_d$.
- MAGNITUDE: $f(\boldsymbol{x}) := \lambda \cdot \sum_{i=s+1}^{d} \tanh(x_{(i)}^2) - \sum_{i=1}^{s} \tanh(x_{(i)}^2) + s$, where λ is a constant, and $x_{(i)}$ denotes the i-th largest magnitude among the coordinates of $\boldsymbol{x}$. For the initial point $\boldsymbol{x}_1$, we randomly sample a subset $S \subseteq [d]$ of size s as the nonzero dimensions of $\boldsymbol{x}_i$ and let $(\boldsymbol{x}_1)_S$ be a constant w times random signs. We use $s = 5$, $\lambda = 0.1$, and $w = 0.2$.
- ATTACK: There are various attacks on graphs (Dai et al., 2018; Fu et al., 2023), and here we consider attacking the connectivity between two vertices on a real-world undirected graph (Girvan & Me, 2002) with $n = 115$ vertices. We assume that the attacker wants to minimally change the adjacency matrix $\boldsymbol{A} \in [0, 1]^{n\times n}$ by a perturbation $\boldsymbol{X} \in \mathbb{R}^{n\times n}$ into $\widetilde{\boldsymbol{A}} := \max\{\boldsymbol{A} \odot (\boldsymbol{1}_{n\times n} - |\boldsymbol{X}|) + (\boldsymbol{1}_{n\times n} - \boldsymbol{A}) \odot |\boldsymbol{X}|, 0\}$ to minimize the connectivity between two vertices u, v, where $|\cdot|$ denotes entry-wise absolute value operation, and $\odot$ denotes entry-wise multiplication. Let $\widetilde{\boldsymbol{D}} := \mathrm{diag}(\widetilde{\boldsymbol{A}}\boldsymbol{1}_{n\times 1})$ denote the degree matrix w.r.t. $\widetilde{\boldsymbol{A}}$, and let $\widetilde{\boldsymbol{A}}_{\text{sym}} := \widetilde{\boldsymbol{D}}^{-1/2}\widetilde{\boldsymbol{A}}\widetilde{\boldsymbol{D}}^{-1/2}$ denote the symmetric normalization of $\widetilde{\boldsymbol{A}}$. Then, we define the objective function by $f(\boldsymbol{X}) := \sum_{w=1}^{W} (\widetilde{\boldsymbol{A}}_{\text{sym}}^{w})_{u,v} + \lambda \|\boldsymbol{X}\|_{\text{F}}^2$, where W denotes the number of hops in connectivity estimation, and $\|\cdot\|_{\text{F}}$ denotes the Frobenius norm. We use $u = 0$, $v = 1$, $W = 4$, and $\lambda = \frac{100}{n^2}$ here. The true sparsity s is unknown, and we use $s = 30$ here. The initial point is $\boldsymbol{X}_1 = \boldsymbol{0}_{n\times n}$.

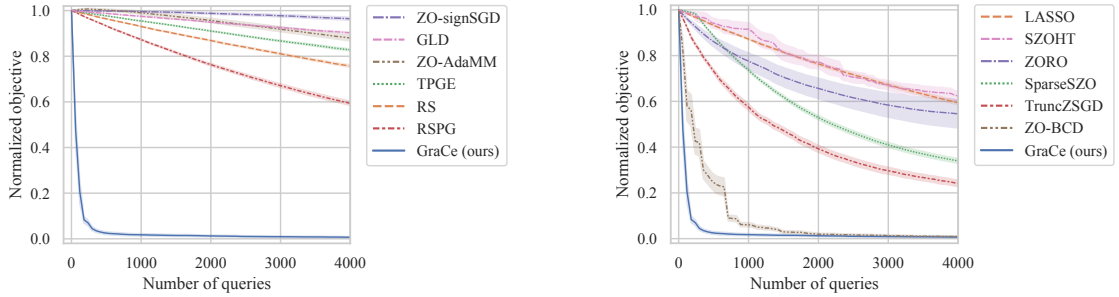
We remark that these functions have approximately sparse gradients along the gradient flow starting from the initial point (although they might have non-sparse gradients elsewhere). Thus, as long as the the gradient estimates are sufficiently accurate, the gradients should be approximately sparse along the optimization trajectory.

5.2. Baselines & Implementation Details

We extensively compare our GraCe with existing ZOO methods including full-gradient and sparse-gradient methods. We do not compare proposed GraCe with global optimization methods such as evolutionary algorithms because they rely on strong prior knowledge on the function structure.

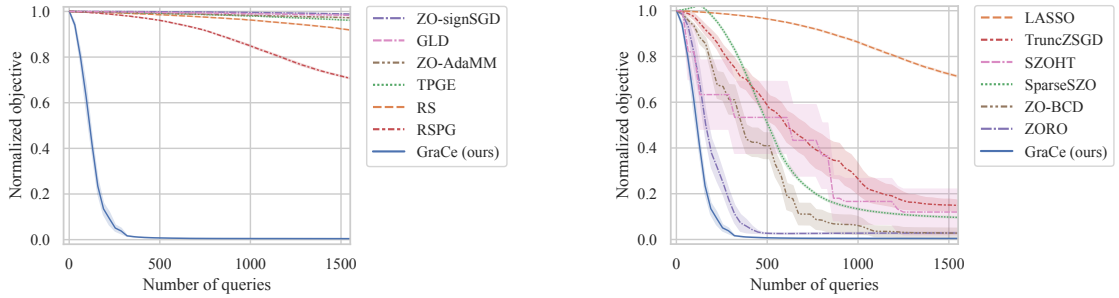
For full-gradient methods, we use the following baselines:

- **RS** (random search): a classic gradient estimator using a random perturbation, which is one of the oldest estimators in ZOO (Spall, 1998) and is later referred to as Gaussian smoothing (Nesterov & Spokoiny, 2015).
- **TPGE** (Duchi et al., 2015): a two-point gradient estimator that uses two additive random perturbations to smooth the function. We use their general-case version.
- **RSPG** (Ghadimi et al., 2016): can be viewed as RS



(a) Comparison with full-gradient methods. (b) Comparison with sparse-gradient methods.

Figure 1. Convergence plots for DISTANCE (mean $\pm$ s.e.).



(a) Comparison with full-gradient methods. (b) Comparison with sparse-gradient methods.

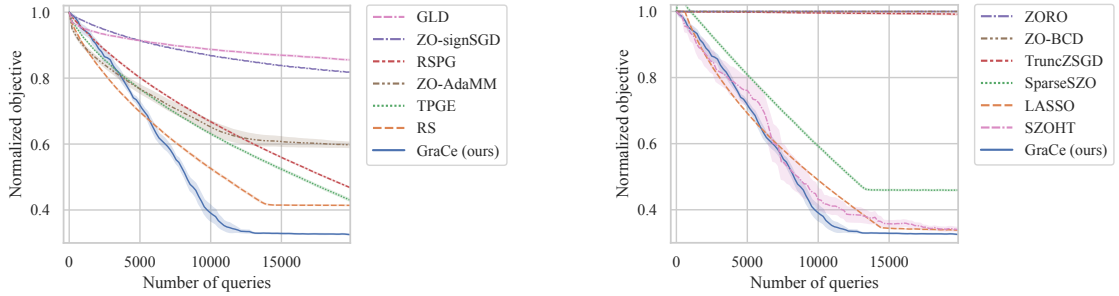
Figure 2. Convergence plots for MAGNITUDE (mean $\pm$ s.e.).

with multiple random perturbations in unconstrained optimization. The multiple perturbations help to reduce the variance of the estimator.

- **ZO-signSGD** (Liu et al., 2019): using only the signs of the zeroth-order gradient estimate instead of specific gradient values.
- **ZO-AdaMM** (Chen et al., 2019): applying the adaptive momentum method to the zeroth-order gradient estimate. As suggested by the authors, we use momentum parameters $\beta_1 = 0.9$ and $\beta_2 = 0.5$.
- **GLD** (Golovin et al., 2020): moving to the best point among K random perturbations of different scales. We use their binary search version with $K = 4$, so the perturbation scales for GLD are $\{\eta, \eta/2, \eta/4, \eta/8\}$, where η is the step size.

For sparse-gradient methods, we use the following state-of-the-art methods as baselines:

- **LASSO** (Wang et al., 2018): generating random signs as perturbations and using LASSO (Tibshirani, 1996) to estimate the sparse gradient. As suggested by the authors, we use their mirror descent version.
- **SparseSZO** (Ohta et al., 2020): applying a mask to the gradient estimate to ensure sparsity and updating the mask periodically according to the magnitudes of coordinates. Following the authors, we update the mask every 5 steps.
- **TruncZSGD** (Balasubramanian & Ghadimi, 2022): truncating the gradient estimate according to the magnitude of its coordinates.
- **ZORO** (Cai et al., 2022): generating random signs as perturbations and using CoSaMP (Needell & Tropp, 2009) to estimate the sparse gradient. Following the authors, we run CoSaMP for at most 10 iterations with tolerence 0.5.
- **ZO-BCD** (Cai et al., 2021): dividing the dimensions into blocks and applying CoSaMP to each block. We use 5 blocks for ZO-BCD because this gives the best performance. Following the authors, we run CoSaMP for at most 10 iterations with tolerence 0.5.
- **SZOHT** (de Vazelhes et al., 2022): perturbing only a random subset of dimensions and applying hard thresholding to the point according to the magnitudes of the coordinates.



(a) Comparison with full-gradient methods.

(b) Comparison with sparse-gradient methods.

Figure 3. Convergence plots for ATTACK (mean $\pm$ s.e.).

The hyperparameters of all methods are summarized in Table 3 in Appendix B.1. To ensure a fair comparison, we let all methods have the same budget number of queries as that of GraCe. Since RS, TPGE, ZO-AdaMM, and GLD use $O(1)$ queries per step, we adjust their number T of steps so that their total number of queries matches that of our GraCe; for other methods, we use the same number of queries per step as that of our GraCe. For each method, we choose the best step size η among $\{0.5, 0.2, 0.1, 0.05, 0.02, 0.01, \dots\}$. For GraCe, we use $m = 1$, $n = \lfloor \frac{0.7d}{s} \rfloor$, and $D_1 = 20$ for DISTANCE and MAGNITUDE and $D_1 = 10$ for ATTACK.

5.3. Results & Discussion

We use the *normalized objective* $\frac{f(\boldsymbol{x}_t)}{f(\boldsymbol{x}_1)}$ as the evaluation metric (the lower, the better). The best normalized objectives found by each method are presented in Table 2, and their convergence plots are shown in Figures 1, 2, & 3. We report means and standard errors (s.e.) over 10 runs.

From Table 2 and Figures 1, 2, & 3, we can observe that our GraCe significantly outperforms baseline methods. For DISTANCE, our GraCe finds a near-optimal solution within 1000 queries while none of the baselines converge even with over 5000 queries. For MAGNITUDE, our GraCe finds a near-optimal solution within 500 queries while most baselines need at least 1000 queries. Furthermore, we can observe that our GraCe achieves consistent strong performance for both DISTANCE and MAGNITUDE. In contrast, the performance of sparse-gradient baselines varies drastically between DISTANCE and MAGNITUDE. For example, TruncZSGD performs well for DISTANCE but not satisfactorily for MAGNITUDE; ZORO performs well for MAGNITUDE but badly for DISTANCE. Our GraCe also achieves the best performance on the real-world dataset ATTACK.

5.4. Additional Experiments

Due to the space limit, we provide additional experiments in Appendix B (i) to show that our GraCe still achieves strong performance even when s is inexact or when the gradient is non-sparse and (ii) to validate that the actual number of queries does scale as the query complexity $O\big(s \log\log \frac{d}{s}\big)$.

6. Related Work

6.1. High-dimensional zeroth-order optimization

Zeroth-order optimization aims to apply first-order optimization methods except with gradients estimated from queries. As queries are typically expensive in practice, the most important metric for comparing ZOO methods is their query complexity (i.e., the total number of queries till convergence). In high-dimensional ZOO, the dimensionality d can be very large. Thus, the main goal of high-dimensional ZOO is to reduce the dependence on d in the query complexity under structural assumptions such as gradient sparsity and solution sparsity. Existing works in high-dimensional ZOO can be categorized into two lines. One line (Ohta et al., 2020; Balasubramanian & Ghadimi, 2022; de Vazelhes et al., 2022) applies a a mask on the gradient or the solution to enforce sparsity. Their query complexity depends on the quality of the masks. The other line (Wang et al., 2018; Cai et al., 2022) employs sparse learning algorithms such as LASSO (Tibshirani, 1996) and CoSaMP (Needell & Tropp, 2009). Their query complexity depends on the accuracy of the sparse learning algorithms.

6.2. High-dimensional first-order optimization

A parallel line of research is high-dimensional first-order optimization, where the gradients of the objective function f can be exactly computed or unbiasedly estimated. In the stark contrast to ZOO, general first-order optimization methods typically have dimension-independent rates of convergence. Thus, unlike ZOO, high-dimensional first-order optimization mainly focuses on handling high-dimensional constraints and achieving further acceleration for special problem structures. Mirror descent (Nemirovski & Yudin, 1983) is an efficient method to handle non-standard geom-

etry. It has been successfully applied to high-dimensional optimization with simplicial (Beck & Teboulle, 2003) and sparsity (Shalev-Shwartz & Tewari, 2009) constraints, and also to problems with convex–concave (Nemirovski et al., 2009) and compositional (Lan, 2012) structures. Other methods such as coordinate descent (Shalev-Shwartz et al., 2010) for sparse optimization and the homotopy method (Xiao & Zhang, 2013) for ℓ_1-regularized least squares have also been developed to further accelerate convergence for special problem structures over general methods.

6.3. Compressed sensing

Our GraCe is a generalization and an improvement of the Indyk–Price–Woodruff (IPW) algorithm (Indyk et al., 2011) in compressed sensing, bridging an interesting connection between zeroth-order optimization and this parallel field. Compressed sensing is a classic field that has been widely studied in various domains, including signal processing, medical imaging, and data compression (Price, 2013). The aim of compressed sensing is to recover a sparse signal from a minimal number of linear measurements. Early methods focuses on non-adaptive measurements. For instance, magnetic resonance imaging (MRI) machines uses 2-dimensional Fourier transforms of the image (Lustig et al., 2008); single-pixel cameras employs wavelet transforms (Duarte et al., 2008). More recently, adaptive methods have been proposed (Haupt et al., 2009), and the IPW algorithm is the state of the art among adaptive methods. Nonetheless, the IPW algorithm is purely theoretical due to its impractically large constant. We have improved the IPW algorithm via our *dependent random partition* technique and our corresponding novel analysis to make it practical.

7. Concluding Remarks

In this paper, we have studied the problem of zeroth-order optimizing (ZOO) in high dimensions for functions with approximately sparse gradients. We have introduced a relaxed assumption on approximate gradient sparsity, which is weaker than previous assumptions. We have proposed a query-efficient gradient estimator called GraCe, whose query complexity has only double-logarithmic dependence on the dimensionality. With the help of GraCe, we have achieved an $O\big(\frac{1}{T}\big)$ rate of convergence for nonconvex ZOO. Experiments have demonstrated the strong empirical performance of our proposed method. We view our work as an early yet inspiring step towards high-dimensional ZOO.

The following are limitations of this work that we wish to be addressed in future work.

- **Stochastic ZOO.** GraCe encodes information into queries, so a limitation of GraCe is that it assumes noise-free function evaluations. Unfortunately, this does not always hold for real-world applications. Thus, an interesting open question is: can we encode information into queries under noisy function evaluations? A possible idea is by employing error correcting codes (Berrou et al., 1993) to encode the dimension information under noisy queries.

- **Lower bound for ZOO.** It is still unclear whether our $O\big(s \log\log \frac{d}{s}\big)$ query complexity is optimal for sparse-gradient ZOO. To date, there is only limited work on the lower bound for ZOO. For instance, Alabdulkareem & Honorio (2021) show that $\Omega(d/\varepsilon^2)$ queries are required for noisy ZOO to achieve ε error in the worst case, but it is still unclear how many queries are required for sparse-gradient ZOO. Thus, an interesting open problem is: can we find a matching lower bound for the query complexity? A possible idea is by considering the example in Price & Woodruff (2013), which has been used to show a $O(\log\log d)$ lower bound for $s = O(1)$ in compressed sensing.

- **ZOO with memory.** Another limitation of GraCe is that it uses only the information collected in each step to find the candidate set J but ignores previous steps. Meanwhile, we observe that the optimal candidate sets of different steps are typically the same or at least highly correlated. Thus, an interesting open problem is: can we leverage the information collected in previous steps to help find the candidate set J using fewer queries than $O\big(s \log\log \frac{d}{s}\big)$? A possible idea is by keeping the candidate set J from the previous step and updating it using a small number of queries.

Acknowledgements

This work was supported by NSF (2134079 and 2324770), NIFA (2020-67021-32799), DHS (17STQAC00001-07-00), AFOSR (FA9550-24-1-0002), and the C3.ai Digital Transformation Institute. The content of the information in this document does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

Impact Statement

This paper presents work whose goal is to advance the field of zeroth-order optimization. There are many potential societal consequences of zeroth-order optimization, including both positive and negative consequences. On the one hand, zeroth-order optimization can be applied to sequential experimental design. Since the query complexity of the zeroth-order optimizer corresponds to the number of experiments, improvement in the query complexity reduces the

cost for the experimenter. On the other hand, zeroth-order optimization can also be applied to black-box adversarial attacks against machine learning models. Since the query complexity of the zeroth-order optimizer corresponds to the number of trials needed by the attacker, improvement in the query complexity reduces the cost for the attacker. We remark that these societal impacts apply to all zeroth-order optimization methods and are not specific to our work.

References

Alabdulkareem, A. and Honorio, J. Information-theoretic lower bounds for zero-order stochastic gradient estimation. In *Proceedings of the 2021 IEEE International Symposium on Information Theory*, pp. 2316–2321. IEEE, 2021.

Ba, K. D., Indyk, P., Price, E., and Woodruff, D. P. Lower bounds for sparse recovery. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1190–1197. SIAM, 2010.

Balasubramanian, K. and Ghadimi, S. Zeroth-order nonconvex stochastic optimization: Handling constraints, high dimensionality, and saddle points. *Foundations of Computational Mathematics*, 22(1):35–76, 2022.

Beck, A. and Teboulle, M. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

Berrou, C., Glavieux, A., and Thitimajshima, P. Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1. In *Proceedings of ICC'93–IEEE International Conference on Communications*, volume 2, pp. 1064–1070. IEEE, 1993.

Cai, H., Lou, Y., McKenzie, D., and Yin, W. A zeroth-order block coordinate descent algorithm for huge-scale blackbox optimization. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 1193–1203. PMLR, 2021.

Cai, H., McKenzie, D., Yin, W., and Zhang, Z. Zeroth-order regularized optimization (ZORO): Approximately sparse gradients and adaptive sampling. *SIAM Journal on Optimization*, 32(2):687–714, 2022.

Chen, X., Liu, S., Xu, K., Li, X., Lin, X., Hong, M., and Cox, D. D. ZO-AdaMM: Zeroth-order adaptive momentum method for black-box optimization. In *Advances in Neural Information Processing Systems*, volume 32, pp. 7202–7213, 2019.

Dai, H., Li, H., Tian, T., Huang, X., Wang, L., Zhu, J., and Song, L. Adversarial attack on graph structured data. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 1115–1124, 2018.

de Vazelhes, W., Zhang, H., Wu, H., Yuan, X., and Gu, B. Zeroth-order hard-thresholding: Gradient error vs expansivity. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Duarte, M. F., Davenport, M. A., Takhar, D., Laska, J. N., Sun, T., Kelly, K. F., and Baraniuk, R. G. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):83–91, 2008.

Duchi, J. C., Jordan, M. I., Wainwright, M. J., and Wibisono, A. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Fu, D., Bao, W., Maciejewski, R., Tong, H., and He, J. Privacy-preserving graph machine learning from data to computation: A survey. *ACM SIGKDD Explorations Newsletter*, 25(1):54–72, 2023.

Ghadimi, S. and Lan, G. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2012.

Ghadimi, S., Lan, G., and Zhang, H. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155 (1-2):267–305, 2016.

Girvan, M. and Me, N. Network of American football games between division ia colleges during regular season fall 2000. *Proceedings of the National Academy of Sciences of the United States of America*, 99:7821–7826, 2002.

Golovin, D., Karro, J., Kochanski, G., Lee, C., Song, X., and Zhang, Q. Gradientless descent: High-dimensional zeroth-order optimization. In *International Conference on Learning Representations*, 2020.

Haupt, J. D., Baraniuk, R. G., Castro, R. M., and Nowak, R. D. Compressive distilled sensing: Sparse recovery using adaptivity in compressive measurements. In *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, pp. 1551–1555. IEEE, 2009.

Indyk, P., Price, E., and Woodruff, D. P. On the power of adaptivity in sparse recovery. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pp. 285–294. IEEE Computer Society, 2011.

Kiefer, J. and Wolfowitz, J. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.

Lan, G. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1-2):365–397, 2012.

Liu, S., Chen, P.-Y., Chen, X., and Hong, M. signSGD via zeroth-order oracle. In *International Conference on Learning Representations*, 2019.

Lustig, M., Donoho, D. L., Santos, J. M., and Pauly, J. M. Compressed sensing MRI. *IEEE Signal Processing Magazine*, 25(2):72–82, 2008.

Matyáš, J. Random optimization. *Avtomatika i Telemekhanika*, pp. 246–253, 1965.

Needell, D. and Tropp, J. A. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and computational harmonic analysis*, 26(3):301–321, 2009.

Nemirovski, A., Juditsky, A., Lan, G., and Shapiro, A. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Nemirovski, A. S. and Yudin, D. B. *Problem Complexity and Method Efficiency in Optimization*. Wiley-Interscience, 1983.

Nesterov, Y. *Lectures on Convex Optimization*, volume 137. Springer, 2018.

Nesterov, Y. and Spokoiny, V. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17, 2015.

Ohta, M., Berger, N., Sokolov, A., and Riezler, S. Sparse perturbations for improved convergence in stochastic zeroth-order optimization. In *International Conference on Machine Learning, Optimization, and Data Science (LOD 2020)*, volume 12566, pp. 39–64. Springer, 2020.

Price, E. and Woodruff, D. P. Lower bounds for adaptive sparse recovery. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 652–663. SIAM, 2013.

Price, E. C. *Sparse recovery and Fourier sampling*. PhD thesis, Massachusetts Institute of Technology, 2013.

Shalev-Shwartz, S. and Tewari, A. Stochastic methods for l 1 regularized loss minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 929–936, 2009.

Shalev-Shwartz, S., Srebro, N., and Zhang, T. Trading accuracy for sparsity in optimization problems with sparsity constraints. *SIAM Journal on Optimization*, 20(6): 2807–2832, 2010.

Spall, J. C. An overview of the simultaneous perturbation method for efficient optimization. *Johns Hopkins APL Technical Digest*, 19(4):482–492, 1998.

Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

Wang, Y., Du, S. S., Balakrishnan, S., and Singh, A. Stochastic zeroth-order optimization in high dimensions. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics (AISTATS) 2018*, volume 84, pp. 1356–1365. PMLR, 2018.

Xiao, L. and Zhang, T. A proximal-gradient homotopy method for the sparse least-squares problem. *SIAM Journal on Optimization*, 23(2):1062–1091, 2013.

Contents

A. Proofs

A.1. Preliminaries

We introduce two classic results on Lipschitz properties, whose proofs can be found, for example, in Nesterov (2018). We restate the theorems and their proofs with our notation as follows.

Lemma A.1 (Lemma 1.2.3, Nesterov, 2018)**.** *For an L_1-Lipschitz smooth function f,*

$$|f(\boldsymbol{x}+\boldsymbol{u}) - f(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \boldsymbol{u}\rangle| \le \frac{L_1}{2}\|\boldsymbol{u}\|_2^2, \qquad \forall \boldsymbol{x}, \boldsymbol{u}. \tag{17}$$

Proof. By the fundamental theorem of calculus, the chain rule, and the the Cauchy–Schwarz inequality,

$$\begin{aligned}
|f(\boldsymbol{x}+\boldsymbol{u}) - f(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \boldsymbol{u}\rangle| &= \left| \int_0^1 \frac{\partial}{\partial \xi} f(\boldsymbol{x}+\xi\boldsymbol{u})\,\mathrm{d}\xi - \langle \nabla f(\boldsymbol{x}), \boldsymbol{u}\rangle \right| && (18)\\
&= \left| \int_0^1 \langle \nabla f(\boldsymbol{x}+\xi\boldsymbol{u}), \boldsymbol{u}\rangle\,\mathrm{d}\xi - \int_0^1 \langle \nabla f(\boldsymbol{x}), \boldsymbol{u}\rangle\,\mathrm{d}\xi \right| && (19)\\
&= \left| \int_0^1 \langle \nabla f(\boldsymbol{x}+\xi\boldsymbol{u}) - \nabla f(\boldsymbol{x}), \boldsymbol{u}\rangle\,\mathrm{d}\xi \right| && (20)\\
&\le \int_0^1 |\langle \nabla f(\boldsymbol{x}+\xi\boldsymbol{u}) - \nabla f(\boldsymbol{x}), \boldsymbol{u}\rangle|\,\mathrm{d}\xi && (21)\\
&\le \int_0^1 \|\nabla f(\boldsymbol{x}+\xi\boldsymbol{u}) - \nabla f(\boldsymbol{x})\|_2 \|\boldsymbol{u}\|_2\,\mathrm{d}\xi && (22)\\
&\le \int_0^1 L_1 \|\xi\boldsymbol{u}\|_2 \|\boldsymbol{u}\|_2\,\mathrm{d}\xi && (23)\\
&= L_1\|\boldsymbol{u}\|_2^2 \int_0^1 \xi\,\mathrm{d}\xi = L_1\|\boldsymbol{u}\|_2^2 \cdot \frac{1}{2}. && \square
\end{aligned}$$

Lemma A.2 (Lemma 1.2.2, Nesterov, 2018)**.** *For an L_0-Lipschitz continuous function f with continuous gradients,*

$$\|\nabla f(\boldsymbol{x})\|_2 \le L_0, \qquad \forall \boldsymbol{x}.$$

Proof. By the chain rule and the L_0-Lipschitz continuity of f,

$$\|\nabla f(\boldsymbol{x})\|_2^2 = \langle \nabla f(\boldsymbol{x}), \nabla f(\boldsymbol{x}) \rangle \tag{24}$$

$$= \frac{\partial}{\partial \xi}\Big|_{\xi=0} f(\boldsymbol{x} + \xi \nabla f(\boldsymbol{x})) \tag{25}$$

$$= \lim_{\xi \searrow 0} \frac{f(\boldsymbol{x} + \xi \nabla f(\boldsymbol{x})) - f(\boldsymbol{x})}{\xi} \tag{26}$$

$$\leq \lim_{\xi \searrow 0} \frac{L_0 \|\xi \nabla f(\boldsymbol{x})\|_2}{\xi} \tag{27}$$

$$= \lim_{\xi \searrow 0} \frac{L_0 \xi \|\nabla f(\boldsymbol{x})\|_2}{\xi} \tag{28}$$

$$= L_0 \|\nabla f(\boldsymbol{x})\|_2. \tag{29}$$

It follows that $\|\nabla f(\boldsymbol{x})\|_2 \leq L_0$, which still holds even when $\|\nabla f(\boldsymbol{x})\|_2 = 0$. □

A.2. Proof of Lemma 3.1

Our Lemma 3.1 is an improvement over Lemma 3.2 in Indyk et al. (2011). Our main improvement here is by using dependent random partition to bound the worst-case block size while Indyk et al. (2011) used independent subsampling. A main difference here is that we use Azuma's inequality to handle weak dependence between blocks.

Before proving our Lemma 3.1, we show a technical lemma via basic calculus.

Lemma A.3. *There is an absolute constant $0 < C_1 < 2.29$ such that for any $0 < \delta < 1$ and any $D \geq 2$,*

$$2D\sqrt{2\ln\frac{4D+1}{2D\delta}} + \sqrt{2\ln\frac{8D+2}{\delta}} \leq \Big(C_1 D + \frac{1}{D}\Big)\sqrt{2\ln\frac{3}{\delta}}.$$

Proof. It suffices to find

$$C_1 := \sup_{\substack{0<\delta<1 \\ D\geq 2}} \left(2\sqrt{\log_{3/\delta}\frac{4D+1}{2D\delta}} + \frac{1}{D}\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}} - \frac{1}{D^2}\right). \tag{30}$$

Note that

$$\frac{\partial}{\partial D}\left(D - \frac{D^2}{(4D+1)\sqrt{\ln\frac{3}{\delta}\ln\frac{4D+1}{2D\delta}}} + \frac{2D^2}{(4D+1)\sqrt{\ln\frac{3}{\delta}\ln\frac{8D+2}{\delta}}} - D\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}}\right) \tag{31}$$

$$= -\frac{1}{2(4D+1)^2}\left(\frac{4D(2D+1)}{\sqrt{\ln\frac{3}{\delta}\ln\frac{4D+1}{2D\delta}}} + \frac{D\sqrt{\log_{3/\delta}\frac{4D+1}{2D\delta}}}{\big(\ln\frac{4D+1}{2D\delta}\big)^2}\right. \tag{32}$$

$$\left. + \frac{2\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}}\big(\big(2D - \frac{1}{2}\ln\frac{8D+2}{\delta}\big)^2 + \big(16D^2+8D+\frac{3}{4}\big)\big(\ln\frac{8D+2}{\delta}\big)^2\big)}{\big(\ln\frac{8D+2}{\delta}\big)^2}\right) \tag{33}$$

$$\leq 0, \tag{34}$$

and that

$$\frac{\partial}{\partial \delta}\left(2 - \frac{4}{9\sqrt{\ln\frac{3}{\delta}\ln\frac{9}{4\delta}}} + \frac{8}{9\sqrt{\ln\frac{3}{\delta}\ln\frac{18}{\delta}}} - 2\sqrt{\log_{3/\delta}\frac{18}{\delta}}\right) \tag{35}$$

$$= -\frac{1}{9\delta\big(\ln\frac{3}{\delta}\big)^{3/2}}\left(\frac{2\big(\ln\frac{27}{4} + 2\ln\frac{1}{\delta}\big)}{\big(\ln\frac{9}{4\delta}\big)^{3/2}} + \frac{(9\ln 6\ln 18 - 4\ln 54) + (9\ln 6 - 8)\ln\frac{1}{\delta}}{\big(\ln\frac{18}{\delta}\big)^{3/2}}\right) \tag{36}$$

$$\leq 0. \tag{37}$$

Thus,

$$\frac{\partial}{\partial D}\left(2\sqrt{\log_{3/\delta}\frac{4D+1}{2D\delta}}+\frac{1}{D}\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}}-\frac{1}{D^2}\right) \tag{38}$$

$$=\frac{1}{D^3}\left(D-\frac{D^2}{(4D+1)\sqrt{\ln\frac{3}{\delta}\ln\frac{4D+1}{2D\delta}}}+\frac{2D^2}{(4D+1)\sqrt{\ln\frac{3}{\delta}\ln\frac{8D+2}{\delta}}}-D\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}}\right) \tag{39}$$

$$\le\frac{1}{D^3}\left(\left(D-\frac{D^2}{(4D+1)\sqrt{\ln\frac{3}{\delta}\ln\frac{4D+1}{2D\delta}}}+\frac{2D^2}{(4D+1)\sqrt{\ln\frac{3}{\delta}\ln\frac{8D+2}{\delta}}}-D\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}}\right)\Bigg|_{D=2}\right) \tag{40}$$

$$=\frac{1}{D^3}\left(2-\frac{4}{9\sqrt{\ln\frac{3}{\delta}\ln\frac{9}{4\delta}}}+\frac{8}{9\sqrt{\ln\frac{3}{\delta}\ln\frac{18}{\delta}}}-2\sqrt{\log_{3/\delta}\frac{18}{\delta}}\right) \tag{41}$$

$$\le\frac{1}{D^3}\lim_{\delta\searrow 0}\left(2-\frac{4}{9\sqrt{\ln\frac{3}{\delta}\ln\frac{9}{4\delta}}}+\frac{8}{9\sqrt{\ln\frac{3}{\delta}\ln\frac{18}{\delta}}}-2\sqrt{\log_{3/\delta}\frac{18}{\delta}}\right) \tag{42}$$

$$=\frac{1}{D^3}(2-0+0-2)=0. \tag{43}$$

It follows that

$$2\sqrt{\log_{3/\delta}\frac{4D+1}{2D\delta}}+\frac{1}{D}\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}}-\frac{1}{D^2} \tag{44}$$

$$\le\left(2\sqrt{\log_{3/\delta}\frac{4D+1}{2D\delta}}+\frac{1}{D}\sqrt{\log_{3/\delta}\frac{8D+2}{\delta}}-\frac{1}{D^2}\right)\Bigg|_{D=2} \tag{45}$$

$$=2\sqrt{\log_{3/\delta}\frac{9}{4\delta}}+\frac{1}{2}\sqrt{\log_{3/\delta}\frac{18}{\delta}}-\frac{1}{4}. \tag{46}$$

Futhermore, note that

$$\frac{\partial}{\partial\ln\frac{1}{\delta}}\left(2\sqrt{\log_{3/\delta}\frac{9}{4\delta}}+\frac{1}{2}\sqrt{\log_{3/\delta}\frac{18}{\delta}}-\frac{1}{4}\right) \tag{47}$$

$$=\frac{\partial}{\partial\ln\frac{1}{\delta}}\left(2\sqrt{\frac{\ln\frac{9}{4}+\ln\frac{1}{\delta}}{\ln 3+\ln\frac{1}{\delta}}}+\frac{1}{2}\sqrt{\frac{\ln 18+\ln\frac{1}{\delta}}{\ln 3+\ln\frac{1}{\delta}}}-\frac{1}{4}\right) \tag{48}$$

$$=\frac{1}{\left(\ln 3+\ln\frac{1}{\delta}\right)^{3/2}}\left(\frac{\ln\frac{4}{3}}{\sqrt{\ln\frac{9}{4}+\ln\frac{1}{\delta}}}-\frac{\ln 6}{4\sqrt{\ln 18+\ln\frac{1}{\delta}}}\right). \tag{49}$$

The derivative equals 0 when $\ln\frac{1}{\delta}=\frac{16\ln 2\left(\ln\frac{4}{3}\right)^2+\ln 4(\ln 6)^2}{(\ln 6)^2-16\left(\ln\frac{4}{3}\right)^2}-\ln 9\approx 0.648887$. Therefore,

$$C_1=\left(2\sqrt{\log_{3/\delta}\frac{9}{4\delta}}+\frac{1}{2}\sqrt{\log_{3/\delta}\frac{18}{\delta}}-\frac{1}{4}\right)\Bigg|_{\ln\frac{1}{\delta}=\frac{16\ln 2\left(\ln\frac{4}{3}\right)^2+\ln 4(\ln 6)^2}{(\ln 6)^2-16\left(\ln\frac{4}{3}\right)^2}-\ln 9}\approx 2.28955. \qquad \square$$

Now we are ready to prove Lemma 3.1.

Proof of Lemma 3.1. To simplify notation, for fixed $\boldsymbol{x}\in\mathbb{R}^d$ and $\epsilon>0$, define functions $g,e:\mathbb{R}^d\to\mathbb{R}$ by

$$g(\boldsymbol{w}):=\frac{f(\boldsymbol{x}+\epsilon\boldsymbol{w})-f(\boldsymbol{x})}{\epsilon},\quad e(\boldsymbol{w}):=g(\boldsymbol{w})-\langle\nabla f(\boldsymbol{x}),\boldsymbol{w}\rangle,\qquad \boldsymbol{w}\in\mathbb{R}^d. \tag{50}$$

Consider the following procedure, which is equivalent to the procedure in Algorithm 1. Sample $\sigma_i\sim\mathsf{Unif}(\{\pm 1\})$ for each $i\in S$ independently, and sample a random permutation $\omega:S\to[|S|]$ of S. Let $B:=\lceil|S|/D\rceil$, and let $h_i:=\lceil\omega(i)/B\rceil$ for

$i \in S$. Note that for any $i \in S$,

$$1 \le h_i \le \left\lceil \frac{|S|}{B} \right\rceil = \left\lceil \frac{|S|}{\lceil |S|/D \rceil} \right\rceil \le \left\lceil \frac{|S|}{|S|/D} \right\rceil = \lceil D \rceil = D. \tag{51}$$

Let $S_p := \{i \in S : h_i = p\}$ for each $p \in [D]$. Define $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$ by

$$u_i := \sigma_i \cdot 1_{\{i \in S\}}, \quad v_i := \sigma_i \cdot h_i \cdot 1_{\{i \in S\}}, \qquad i = 1, \dots, d. \tag{52}$$

Make queries $f(\boldsymbol{x} + \epsilon \boldsymbol{u})$ and $f(\boldsymbol{x} + \epsilon \boldsymbol{v})$. We claim that $S_{\mathrm{round}\left(\frac{f(\boldsymbol{x}+\epsilon\boldsymbol{v})-f(\boldsymbol{x})}{f(\boldsymbol{x}+\epsilon\boldsymbol{u})-f(\boldsymbol{x})}\right)}$ is the desired S'.

We prove the claim as follows. Let $q := h_j$. First, since

$$|S_q| = \#\left\{i \in S : \left\lceil \frac{\omega(i)}{B} \right\rceil = q\right\} = \#\left\{i \in S : q - 1 < \frac{\omega(i)}{B} \le q\right\} \tag{53}$$

$$= \#\{i \in S : (q-1)B < \pi(i) \le qB\} \le qB - (q-1)B = B = \left\lceil \frac{|S|}{D} \right\rceil. \tag{54}$$

then

$$|S_q \setminus \{j\}| = |S_q| - 1 \le \left\lceil \frac{|S|}{D} \right\rceil - 1 = \left\lfloor \frac{|S| + D - 1}{D} \right\rfloor - 1 \tag{55}$$

$$\le \frac{|S| + D - 1}{D} - 1 = \frac{|S| - 1}{D} = \frac{|S \setminus \{j\}|}{D}. \tag{56}$$

Second, since

$$\mathbb{E}[\|\nabla_{S_q \setminus \{j\}} f(\boldsymbol{x})\|_2^2] = \sum_{i \in S} (\nabla_i f(\boldsymbol{x}))^2 \mathbb{P}\{i \in S_q \setminus \{j\}\} = \sum_{i \in S} (\nabla_i f(\boldsymbol{x}))^2 \frac{|S_q| - 1}{|S|} \tag{57}$$

$$= \frac{\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2 (|S_q| - 1)}{|S|} \le \frac{\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2 (\lceil |S|/D \rceil - 1)}{|S|} \tag{58}$$

$$\le \frac{\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2 (|S|/D)}{|S|} = \frac{\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2}{D}, \tag{59}$$

then by Markov's inequality, w.p. $\ge 1 - \delta_2$,

$$\|\nabla_{S_q \setminus \{j\}} f(\boldsymbol{x})\|_2 \le \sqrt{\frac{\mathbb{E}[\|\nabla_{S_q \setminus \{j\}} f(\boldsymbol{x})\|_2^2]}{\delta_2}} \le \sqrt{\frac{\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2}{D\delta_2}} = \frac{\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2}{\sqrt{D\delta_2}}. \tag{60}$$

Next, since $|\nabla_i f(\boldsymbol{x}) \cdot u_i| \le |\nabla_i|$ for all i, and

$$\sum_{i \in S \setminus \{j\}} (|\nabla_i f(\boldsymbol{x})| - (-|\nabla_i f(\boldsymbol{x})|))^2 = 4\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2, \tag{61}$$

then by Hoeffding's inequality, w.p. $\ge 1 - \frac{\delta_1}{4D+1}$,

$$|\langle \nabla_{S \setminus \{j\}} f(\boldsymbol{x}), \boldsymbol{u}_{S \setminus \{j\}} \rangle| \le \sqrt{\frac{4\|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2}{2} \ln \frac{2}{\frac{\delta_1}{4D+1}}} = \|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2 \sqrt{2 \ln \frac{8D+2}{\delta_1}}. \tag{62}$$

Besides that, thanks to the independent signs $\{\sigma_i\}_{i \in S \setminus \{j\}}$, we can view $\{\nabla_i f(\boldsymbol{x}) \cdot (v_i - q u_i)\}_{i \in S \setminus \{j\}} = \{\nabla_i f(\boldsymbol{x}) \cdot \sigma_i \cdot (h_i - q)\}_{i \in S \setminus \{j\}}$ as a martingale difference sequence. Since $|\nabla_i f(\boldsymbol{x}) \cdot (v_i - q u_i)| \le (D-1)|\nabla_i| \le D|\nabla_i|$ for all i, and

$$\sum_{i \in S \setminus \{j\}} (D|\nabla_i f(\boldsymbol{x})| - (-D|\nabla_i f(\boldsymbol{x})|))^2 = 4D^2 \|\nabla_{S \setminus \{j\}} f(\boldsymbol{x})\|_2^2, \tag{63}$$

then by Azuma's inequality, w.p. $\geq 1 - \frac{4D\delta_1}{4D+1}$,

$$|\langle \nabla_{S\setminus\{j\}} f(\boldsymbol{x}), \boldsymbol{v}_{S\setminus\{j\}} - q\boldsymbol{u}_{S\setminus\{j\}}\rangle| \leq \sqrt{\frac{4D^2\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2^2}{2} \ln \frac{2}{\frac{4D\delta_1}{4D+1}}} = D\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2 \sqrt{2\ln\frac{4D+1}{2D\delta_1}}. \tag{64}$$

Third, by Lemma A.1,

$$|e(\boldsymbol{u})| \leq \frac{L_1\epsilon}{2}\|\boldsymbol{u}\|_2^2 = \frac{L_1\epsilon}{2}\sum_{i\in S}\sigma_i^2 = \frac{L_1\epsilon}{2}\sum_{i\in S}1^2 = \frac{L_1|S|}{2}\epsilon, \tag{65}$$

$$|e(\boldsymbol{v})| \leq \frac{L_1\epsilon}{2}\|\boldsymbol{v}\|_2^2 = \frac{L_1\epsilon}{2}\sum_{i\in S}(\sigma_i h_i)^2 \leq \frac{L_1\epsilon}{2}\sum_{i\in S}D^2 = \frac{L_1|S|D^2}{2}\epsilon \leq \frac{L_1|S|d^2}{2}\epsilon. \tag{66}$$

Fourth, with the absolute constant C_1 in Lemma A.3, the assumption on j implies

$$|\nabla_j f(\boldsymbol{x})| > \big(C_1 D + \tfrac{1}{D}\big)\sqrt{2\ln\tfrac{3}{\delta_1}}\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2 + L_1|S|\Big(d^2 + d + \frac{1}{2}\Big)\epsilon \tag{67}$$

$$\geq 2D\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2\sqrt{2\ln\frac{4D+1}{2D\delta_1}} + \|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2\sqrt{2\ln\frac{8D+2}{\delta_1}} + L_1|S|\Big(d^2 + d + \frac{1}{2}\Big)\epsilon. \tag{68}$$

Thus,

$$\frac{D\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2\sqrt{2\ln\frac{4D+1}{2D\delta_1}} + \frac{L_1|S|d^2}{2}\epsilon + d\frac{L_1|S|}{2}\epsilon}{|\nabla_j f(\boldsymbol{x})| - \|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2\sqrt{2\ln\frac{8D+2}{\delta_1}} - \frac{L_1|S|}{2}\epsilon} < \frac{1}{2}. \tag{69}$$

Finally, by Eqs. (62), (64), (65), (66), & (69), with probability $\geq 1 - \big(\delta_2 + \frac{\delta_1}{4D+1} + \frac{4D\delta_1}{4D+1}\big) = 1 - (\delta_1 + \delta_2)$,

$$\Big|\frac{f(\boldsymbol{x}+\epsilon\boldsymbol{v}) - f(\boldsymbol{x})}{f(\boldsymbol{x}+\epsilon\boldsymbol{u}) - f(\boldsymbol{x})} - q\Big| = \Big|\frac{g(\boldsymbol{v})}{g(\boldsymbol{u})} - q\Big| \tag{70}$$

$$= \Big|\frac{\sigma_j q\nabla_j f(\boldsymbol{x}) + \langle \nabla_{S\setminus\{j\}} f(\boldsymbol{x}), \boldsymbol{v}_{S\setminus\{j\}}\rangle + e(\boldsymbol{v})}{\sigma_j \nabla_j f(\boldsymbol{x}) + \langle \nabla_{S\setminus\{j\}} f(\boldsymbol{x}), \boldsymbol{u}_{S\setminus\{j\}}\rangle + e(\boldsymbol{u})} - q\Big| \tag{71}$$

$$= \Big|\frac{\langle \nabla_{S\setminus\{j\}} f(\boldsymbol{x}), \boldsymbol{v}_{S\setminus\{j\}} - q\boldsymbol{u}_{S\setminus\{j\}}\rangle + e(\boldsymbol{v}) - qe(\boldsymbol{u})}{\sigma_j \nabla_j f(\boldsymbol{x}) + \langle \nabla_{S\setminus\{j\}} f(\boldsymbol{x}), \boldsymbol{u}_{S\setminus\{j\}}\rangle + e(\boldsymbol{u})}\Big| \tag{72}$$

$$\leq \frac{|\langle \nabla_{S\setminus\{j\}} f(\boldsymbol{x}), \boldsymbol{v}_{S\setminus\{j\}} - q\boldsymbol{u}_{S\setminus\{j\}}\rangle| + |e(\boldsymbol{v})| + q|e(\boldsymbol{u})|}{|\nabla_j f(\boldsymbol{x})| - |\langle \nabla_{S\setminus\{j\}} f(\boldsymbol{x}), \boldsymbol{u}_{S\setminus\{j\}}\rangle| - |e(\boldsymbol{u})|} \tag{73}$$

$$\leq \frac{D\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2\sqrt{2\ln\frac{4D+1}{2D\delta_1}} + \frac{L_1|S|d^2}{2}\epsilon + d\frac{L_1|S|}{2}\epsilon}{|\nabla_j f(\boldsymbol{x})| - \|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2\sqrt{2\ln\frac{8D+2}{\delta_1}} - \frac{L_1|S|}{2}\epsilon} \tag{74}$$

$$< \frac{1}{2}. \tag{75}$$

This implies $\mathrm{round}\big(\frac{f(\boldsymbol{x}+\epsilon\boldsymbol{v})-f(\boldsymbol{x})}{f(\boldsymbol{x}+\epsilon\boldsymbol{u})-f(\boldsymbol{x})}\big) = q = h_j$, completing the proof. □

A.3. Proof of Lemma 3.2

The following Lemma A.4 is the general version of our Lemma 3.2, which is an improvement of Lemma 3.3 in Indyk et al. (2011). The main technical difficulties here are (i) how to construct a division schedule $\{D_r\}_{r\geq 1}$ with which we can iteratively apply Lemma 3.1 and (ii) how to show that the constructed division schedule grows rapidly.

Lemma A.4. *Suppose that hyperparameters $0 < \delta, \phi, \theta < 1$, and integer $D \geq 2$ satisfy*

$$\phi\bigg(D\frac{(1-\theta)(1-\phi)\delta\ln(3/(\theta(1-\phi)\delta))}{\ln(3/(\theta(1-\phi)\phi\delta))}\bigg)^{3/2} - D\frac{(1-\theta)(1-\phi)\delta\ln(3/(\theta(1-\phi)\delta))}{\ln(3/(\theta(1-\phi)\phi\delta))} \geq (1-\theta)(1-\phi)\phi\delta, \tag{76}$$

$$D\frac{(1-\theta)(1-\phi)\delta\ln(3/(\theta(1-\phi)\delta))}{\ln(3/(\theta(1-\phi)\phi\delta))} \geq \frac{3}{2}, \tag{77}$$

and

$$A := \left(D - \frac{1}{\sqrt{D\frac{(1-\theta)(1-\phi)\delta\ln(3/(\theta(1-\phi)\delta))}{\ln(3/(\theta(1-\phi)\phi\delta))}} - 1}\right)\frac{(1-\theta)(1-\phi)\phi^2\delta\ln\frac{3}{\theta(1-\phi)\delta}}{\ln\frac{3}{\theta(1-\phi)\phi\delta}} > 1. \tag{78}$$

There exists a division schedule $\{D_r\}_{r\geq 1}$ *with* $D_1 = D$ *and* $D_r \geq \frac{A^{(3/2)^{r-1}}}{(1-\theta)(1-\phi)\phi^2\delta}$ *such that given* $\boldsymbol{x} \in \mathbb{R}^d$, $\epsilon > 0$, $S \subseteq [d]$, *if there exists* $j \in S$ *with*

$$|\nabla_j f(\boldsymbol{x})| > \left(C_1 D + \frac{1}{D}\right)\sqrt{2\ln\frac{3}{\theta(1-\phi)\delta}}\|\nabla_{S\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon, \tag{79}$$

then $O(\log_{3/2}\log_A|S|)$ *iterations of Lemma 3.1 with parameters* $\{D_r\}_{r\geq 1}$ *can find* j *with probability at least* $1-\delta$.

Before proving Lemma A.4, we show a technical lemma.

Lemma A.5. *For any* $0 < \phi < 1$ *and* $0 < \delta < 1$, *if there is* $x_0 \geq 0$ *with* $\phi x_0^{3/2} - x_0 \geq \phi\delta$, *then* $\phi x^{3/2} - x \geq \phi\delta\ \forall x \geq x_0$.

Proof. Let $\psi(x) := \phi x^{3/2} - x$, $x \geq 0$. Since

$$\psi'(x) = \frac{3}{2}\phi x^{1/2} - 1, \tag{80}$$

then $\psi(x)$ is decreasing over $\left[0, \frac{4}{9\phi^2}\right]$ and increasing over $\left(\frac{4}{9\phi^2}, +\infty\right)$. Thus, for every $x \in \left[0, \frac{4}{9\phi^2}\right]$,

$$\psi(x) \leq \psi(0) = 0 < \phi\delta. \tag{81}$$

This implies $x_0 \in \left(\frac{4}{9\phi^2}, +\infty\right)$, over which ψ is increasing. It follows that for every $x \geq x_0$,

$$\psi(x) \geq \psi(x_0) \geq \phi\delta. \qquad \square$$

Now we are ready to prove Lemma A.4.

Proof of Lemma A.4. For $r \geq 1$, let

$$\delta_{r,1} := \theta(1-\phi)\phi^{r-1}\delta, \qquad \delta_{r,2} := (1-\theta)(1-\phi)\phi^{r-1}\delta. \tag{82}$$

Let $D_1 := D$, and let $D_{r+1} := \left\lfloor D_r^{3/2}\sqrt{\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}\right\rfloor$ for $r \geq 1$. Note that

$$\frac{\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}} = 1 - \frac{\ln\frac{1}{\phi}}{\ln\frac{3}{\delta_{r+1,1}}} \tag{83}$$

is non-decreasing as r increases, so we have

$$\frac{\ln\frac{3}{\delta_{r+1,1}}}{\ln\frac{3}{\delta_{r+2,1}}} \geq \frac{\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}. \tag{84}$$

We will show by strong induction that for every $r \geq 1$,

$$\phi\left(D_r\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}\right)^{3/2} - D_r\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}} \geq \frac{\phi\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}, \tag{85}$$

and

$$D_{r+1}\frac{\delta_{r+1,2}\ln(3/\delta_{r+1,1})}{\ln(3/\delta_{r+2,1})} \geq D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}. \tag{86}$$

First, consider the base case $r = 1$. Using the assumption Eq. (76) and the fact that $\delta_{2,1} \le \delta_{1,1}$,

$$\phi\left(D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}\right)^{3/2} - D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} \ge \phi\delta_{1,2} \ge \frac{\phi\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}. \tag{87}$$

Furthermore, by Eqs. (84) & (87),

$$\begin{aligned}
D_2 \frac{\delta_{2,2} \ln \frac{3}{\delta_{2,1}}}{\ln \frac{3}{\delta_{3,1}}} - D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} &= \left\lfloor D_1^{3/2} \sqrt{\frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}} \right\rfloor \frac{\phi\delta_{1,2} \ln \frac{3}{\delta_{2,1}}}{\ln \frac{3}{\delta_{3,1}}} - D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} && (88)\\
&\ge \left\lfloor D_1^{3/2} \sqrt{\frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}} \right\rfloor \frac{\phi\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} - D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} && (89)\\
&\ge \left(D_1^{3/2} \sqrt{\frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}} - 1 \right) \frac{\phi\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} - D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} && (90)\\
&= \phi\left(D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}\right)^{3/2} - D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} - \frac{\phi\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}} && (91)\\
&\ge 0. && (92)
\end{aligned}$$

Next, suppose that the inductive hypotheses Eqs. (85) & (86) hold for r, and consider the induction step from r to $r+1$. By strong induction w.r.t. Eq. (86),

$$D_{r+1} \frac{\delta_{r+1,2} \ln(3/\delta_{r+1,1})}{\ln(3/\delta_{r+2,1})} \ge D_1 \frac{\delta_{1,2} \ln(3/\delta_{1,1})}{\ln(3/\delta_{2,1})}. \tag{93}$$

By Lemma A.5 with $x_0 = D_1 \frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}$ and Eqs. (87) & (93) as well as the fact that $\delta_{r+2,1} \le \delta_{r+1,1}$,

$$\phi\left(D_{r+1} \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}}\right)^{3/2} - D_{r+1} \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} \ge \phi\delta_{1,2} \ge \phi\delta_{r+1,2} \ge \frac{\phi\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}}. \tag{94}$$

Furthermore, by Eqs. (84) & (94),

$$\begin{aligned}
&D_{r+2} \frac{\delta_{r+2,2} \ln \frac{3}{\delta_{r+2,1}}}{\ln \frac{3}{\delta_{r+3,1}}} - D_{r+1} \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} && (95)\\
&= \left\lfloor D_{r+1}^{3/2} \sqrt{\frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}}} \right\rfloor \frac{\phi\delta_{r+1,2} \ln \frac{3}{\delta_{r+2,1}}}{\ln \frac{3}{\delta_{r+3,1}}} - D_1 \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} && (96)\\
&\ge \left\lfloor D_{r+1}^{3/2} \sqrt{\frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}}} \right\rfloor \frac{\phi\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} - D_{r+1} \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} && (97)\\
&\ge \left(D_{r+1}^{3/2} \sqrt{\frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}}} - 1 \right) \frac{\phi\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} - D_{r+1} \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} && (98)\\
&= \phi\left(D_{r+1} \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}}\right)^{3/2} - D_{r+1} \frac{\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} - \frac{\phi\delta_{r+1,2} \ln \frac{3}{\delta_{r+1,1}}}{\ln \frac{3}{\delta_{r+2,1}}} && (99)\\
&\ge 0. && (100)
\end{aligned}$$

Hence, the inductive hypotheses hold for all $r \ge 1$. In particular, Eq. (86) shows that $D_r \frac{\delta_{r,2} \ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}$ is non-decreasing as r increases. Since $D_r \frac{\delta_{r,2} \ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}$ is non-decreasing with r, but both $\delta_{r,2}$ and $\frac{\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}$ are non-increasing with r, then D_r

is non-decreasing with r. Together with assumption Eq. (77), for every $r \geq 1$,

$$\frac{1}{D_r\left(\sqrt{D_r \frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1\right)} \leq \frac{1}{D_r\left(\sqrt{D_1 \frac{\delta_{1,2}\ln(3/\delta_{1,1})}{\ln(3/\delta_{2,1})}}-1\right)} \leq \frac{1}{D_r\left(\frac{3}{2}-1\right)} = \frac{2}{D_r} \leq \frac{2}{D_1} \leq \frac{2}{2} = 1. \tag{101}$$

Furthermore, using the fact that $\delta_{r+1,2} = \phi\delta_{r,2}$ and Eq. (84),

$$\left(D_{r+1} - \frac{1}{\sqrt{D_{r+1}\frac{\delta_{r+1,2}\ln(3/\delta_{r+1,1})}{\ln(3/\delta_{r+2,1})}}-1}\right)\frac{\phi^2\delta_{r+1,2}\ln\frac{3}{\delta_{r+1,1}}}{\ln\frac{3}{\delta_{r+2,1}}} \tag{102}$$

$$= \left(D_{r+1} - \frac{1}{\sqrt{D_{r+1}\frac{\delta_{r+1,2}\ln(3/\delta_{r+1,1})}{\ln(3/\delta_{r+2,1})}}-1}\right)\frac{\phi^3\delta_{r,2}\ln\frac{3}{\delta_{r+1,1}}}{\ln\frac{3}{\delta_{r+2,1}}} \tag{103}$$

$$\geq \left(D_{r+1} - \frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^3\delta_{r,2}\ln\frac{3}{\delta_{r+1,1}}}{\ln\frac{3}{\delta_{r+2,1}}} \tag{104}$$

$$\geq \left(D_{r+1} - \frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^3\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}. \tag{105}$$

Plugging into the definition of D_{r+1} gives

$$\left(D_{r+1} - \frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^3\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}} \tag{106}$$

$$= \left(\left\lfloor D_r^{3/2}\sqrt{\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}}\right\rfloor - \frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^3\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}} \tag{107}$$

$$\geq \left(D_r^{3/2}\sqrt{\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}} - 1 - \frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^3\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}} \tag{108}$$

$$= \left(\left(D_r\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}\right)^{3/2} - \frac{\sqrt{D_r}\left(\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}\right)^{3/2}}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\phi^3 \tag{109}$$

$$= \left(D_r\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}\right)^{3/2}\left(1 - \frac{1}{D_r\left(\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1\right)}\right)\phi^3. \tag{110}$$

Since $0 \leq 1 - \frac{1}{D_r\left(\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1\right)} \leq 1$ by Eq. (101), then

$$\left(D_r\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}\right)^{3/2}\left(1 - \frac{1}{D_r\left(\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1\right)}\right)\phi^3 \tag{111}$$

$$\geq \left(D_r\frac{\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}\right)^{3/2}\left(1 - \frac{1}{D_r\left(\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1\right)}\right)^{3/2}\phi^3 \tag{112}$$

$$= \left(\left(D_r - \frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^2\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}\right)^{3/2}. \tag{113}$$

It follows from Eqs. (105), (110), & (113) that

$$\left(D_{r+1}-\frac{1}{\sqrt{D_{r+1}\frac{\delta_{r+1,2}\ln(3/\delta_{r+1,1})}{\ln(3/\delta_{r+2,1})}}-1}\right)\frac{\phi^2\delta_{r+1,2}\ln\frac{3}{\delta_{r+1,1}}}{\ln\frac{3}{\delta_{r+2,1}}}\geq\left(\left(D_r-\frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^2\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}}\right)^{3/2}. \tag{114}$$

Then by an induction argument using Eq. (114), we can show that for every $r\geq 1$,

$$\left(D_r-\frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\right)\frac{\phi^2\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}{\ln\frac{3}{\delta_{r+1,1}}} \tag{115}$$

$$\geq\left(\left(D_1-\frac{1}{\sqrt{D_1\frac{\delta_{1,2}\ln(3/\delta_{1,1})}{\ln(3/\delta_{2,1})}}-1}\right)\frac{\phi^2\delta_{1,2}\ln\frac{3}{\delta_{1,1}}}{\ln\frac{3}{\delta_{2,1}}}\right)^{(3/2)^{r-1}} \tag{116}$$

$$=A^{(3/2)^{r-1}}. \tag{117}$$

Hence,

$$D_r>D_r-\frac{1}{\sqrt{D_r\frac{\delta_{r,2}\ln(3/\delta_{r,1})}{\ln(3/\delta_{r+1,1})}}-1}\geq\frac{\ln\frac{3}{\delta_{r+1,1}}}{\phi^2\delta_{r,2}\ln\frac{3}{\delta_{r,1}}}A^{(3/2)^{r-1}}\geq\frac{1}{\phi^2\delta_{1,2}}A^{(3/2)^{k-1}}=\frac{A^{(3/2)^{r-1}}}{(1-\theta)(1-\phi)\phi^2\delta}. \tag{118}$$

Finally, recall the assumption Eq. (79):

$$|\nabla_j f(\boldsymbol{x})|>\Big(C_1D+\frac{1}{D}\Big)\sqrt{2\ln\frac{3}{\theta(1-\phi)\delta}}\|\nabla_{S\setminus\{j\}}f(\boldsymbol{x})\|_2+\lambda_{1,|S|}\cdot\epsilon \tag{119}$$

$$=\Big(C_1D_1+\frac{1}{D_1}\Big)\sqrt{2\ln\frac{3}{\delta_{1,1}}}\|\nabla_{S\setminus\{j\}}f(\boldsymbol{x})\|_2+\lambda_{1,|S|}\cdot\epsilon. \tag{120}$$

Then, applying Lemma 3.1 with $(S,D_1,\delta_{1,1},\delta_{1,2})$ gives a set $S_1\subseteq S$ with

$$|S_1\setminus\{j\}|\leq\frac{|S\setminus\{j\}|}{D_1},\qquad\|\nabla_{S_1\setminus\{j\}}f(\boldsymbol{x})\|_2\leq\frac{\|\nabla_{S\setminus\{j\}}f(\boldsymbol{x})\|_2}{\sqrt{D_1\delta_{1,2}}}. \tag{121}$$

By Eqs. (79) & (121),

$$|\nabla_j f(\boldsymbol{x})|>\Big(C_1D_1+\frac{1}{D_1}\Big)\sqrt{2\ln\frac{3}{\delta_{1,1}}}\|\nabla_{S\setminus\{j\}}f(\boldsymbol{x})\|_2+\lambda_{1,|S|}\cdot\epsilon \tag{122}$$

$$\geq\Big(C_1D_1+\frac{1}{D_1}\Big)\sqrt{2\ln\frac{3}{\delta_{1,1}}}\sqrt{D_1\delta_{1,2}}\|\nabla_{S_1\setminus\{j\}}f(\boldsymbol{x})\|_2+\lambda_{1,|S|}\cdot\epsilon \tag{123}$$

$$=\sqrt{2}\Big(C_1+\frac{1}{D_1^2}\Big)D_1^{3/2}\sqrt{\delta_{1,2}\ln\frac{3}{\delta_{1,1}}}\|\nabla_{S_1\setminus\{j\}}f(\boldsymbol{x})\|_2+\lambda_{1,|S|}\cdot\epsilon. \tag{124}$$

Since we have shown that $\{D_r\}_{r\geq 1}$ is non-decreasing, it follows that

$$|\nabla_j f(\boldsymbol{x})| > \sqrt{2}\Big(C_1 + \frac{1}{D_1^2}\Big) D_1^{3/2} \sqrt{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}} \|\nabla_{S_1\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon \tag{125}$$

$$\geq \sqrt{2}\Big(C_1 + \frac{1}{D_2^2}\Big) D_1^{3/2} \sqrt{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}} \|\nabla_{S_1\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon \tag{126}$$

$$= \sqrt{2}\Big(C_1 + \frac{1}{D_2^2}\Big) D_1^{3/2} \sqrt{\frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}} \sqrt{\ln \frac{3}{\delta_{2,1}}} \|\nabla_{S_1\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon \tag{127}$$

$$\geq \sqrt{2}\Big(C_1 + \frac{1}{D_2^2}\Big) \Big\lfloor D_1^{3/2} \sqrt{\frac{\delta_{1,2} \ln \frac{3}{\delta_{1,1}}}{\ln \frac{3}{\delta_{2,1}}}} \Big\rfloor \sqrt{\ln \frac{3}{\delta_{2,1}}} \|\nabla_{S_1\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon \tag{128}$$

$$= \sqrt{2}\Big(C_1 + \frac{1}{D_2^2}\Big) D_2 \sqrt{\ln \frac{3}{\delta_{2,1}}} \|\nabla_{S_1\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon \tag{129}$$

$$= \Big(C_1 D_2 + \frac{1}{D_2}\Big) \sqrt{2 \ln \frac{3}{\delta_{2,1}}} \|\nabla_{S_1\setminus\{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S|}\cdot\epsilon. \tag{130}$$

Thus, we can apply Lemma 3.1 again to (S_1, D_2). In fact, the same argument holds for all $r \geq 1$. We can repeat Lemma 3.1 with division schedule $\{D_r\}_{r\geq 1}$ for

$$R := \lceil \log_{3/2} \log_A(\phi^2 \delta_{1,2} |S\setminus\{j\}|) \rceil + 2 = O(\log_{3/2} \log_A |S|) \tag{131}$$

times to get $S_R \subseteq S$, and we have

$$|S_R \setminus \{j\}| \leq \frac{|S\setminus\{j\}|}{D_1 \cdots D_R} \leq \frac{|S\setminus\{j\}|}{D_R} < 1. \tag{132}$$

This implies $S_R = \{j\}$, so we have found j. The total probability of failure is at most

$$\sum_{r=1}^{R} (\delta_{r,1} + \delta_{r,2}) = \sum_{r=1}^{R} (\theta(1-\phi)\phi^{r-1}\delta + (1-\theta)(1-\phi)\phi^{r-1}\delta) \tag{133}$$

$$= \sum_{r=1}^{R} (1-\phi)\phi^{r-1}\delta = (1-\phi^R)\delta < \delta. \qquad \square$$

Proof of Lemma 3.2. We can use Lemma A.4 with $D = 18$, $\delta = 1/2$, $\phi = 0.64$, and $\theta = 0.08$, and it gives

$$C_2 := \Big(C_1 D + \frac{1}{D}\Big) \sqrt{2 \ln \frac{3}{\theta(1-\phi)\delta}} \approx 134.88. \qquad \square$$

Remark A.6. Our constant C_2 here is nearly 4300 times smaller than the corresponding constant $C_2' \approx 579263$ of the IPW algorithm.

A.4. Proof of Lemma 3.3

Our Lemma 3.3 is an improvement over Lemma 3.6 in Indyk et al. (2011). Our main improvement here is by using dependent random partition to bound the worst-case size of candidate groups while Indyk et al. (2011) used independent subsampling. A main difference here is our Lemma A.7 for analyzing dependent random partition.

Before proving Lemma 3.3, we show two technical lemmas. Let $(n)_m := \prod_{k=0}^{m-1}(n-k)$ denote the falling factorial.

Lemma A.7. *Let ω be a random permutation of $[d]$. Given $1 \le n \le d$, $1 \le k \le \lceil \frac{d}{n} \rceil$, define a random set*

$$S := \Big\{ j \in [d] : \Big\lceil \frac{\omega(j)}{n} \Big\rceil = k \Big\}. \tag{134}$$

Note that $|S| = \sum_{q \in [d]} 1_{[\lceil q/n \rceil = k]}$ is not random. Then, given any $H \subset [d]$ and any $j \in H$,

$$\mathbb{P}[S \cap H = \{j\}] = \frac{|S|}{d} \cdot \frac{(d - |S|)_{|H|-1}}{(d-1)_{|H|-1}}. \tag{135}$$

Besides that, given any $H \subset [d]$, any $J \subseteq H$ with $|J| \le |S|$, and any $i \in [d] \setminus H$

$$\mathbb{P}[i \in S \mid S \cap H = J] = \frac{|S| - |J|}{d - |H|}. \tag{136}$$

Proof. W.l.o.g., suppose that $H = [|H|]$, $j = 1 \in H$, $i = |H| + 1 \in [d] \setminus H$. Recall the brute-force algorithm for generating a random permutation (shown Algorithm 3). To calculate the desired probabilities, we suppose that the random permutation ω is indeed generated via Algorithm 3 from now on.

Algorithm 3 Generating a random permutation

Input: the number d of elements
Output: a random permutation of $[d]$
1: **for** $r \leftarrow 1, 2, \ldots, d$ **do**
2: $\quad \omega(r) \leftarrow \mathsf{Unif}([d] \setminus \{\omega(1), \omega(2), \ldots, \omega(r-1)\})$
3: **end for**
4: **return** ω

Note that $S = \{r \in [d] : \omega(r) \in Q\}$, where $Q := \{q \in [d] : \lceil q/n \rceil = k\}$. We will use this set Q to rewrite events.

First, let's calculate $\mathbb{P}[S \cap H = \{j\}]$. For $j = 1$, the event that $j \in S$ is equivalent to the event that $\omega(j) \in Q$; for all other $r \in H \setminus \{j\}$, the event that $r \notin S$ is equivalent to the event that $\omega(j) \notin Q$. Hence, by the chain rule,

$$\begin{aligned}
\mathbb{P}[S \cap H = \{j\}] &= \mathbb{P}[\omega(1) \in Q, \omega(2), \ldots, \omega(|H|) \notin Q] && (137)\\
&= \mathbb{P}[\omega(1) \in Q] \cdot \prod_{r=2}^{|H|} \mathbb{P}[\omega(r) \notin Q \mid \omega(1) \in Q, \omega(2), \ldots, \omega(r-1) \notin Q] && (138)\\
&= \frac{|Q|}{d} \cdot \prod_{r=2}^{|H|} \frac{d - |Q| - (r-2)}{d - (r-1)} = \frac{|Q|}{d} \cdot \frac{(d-|Q|)_{|H|-1}}{(d-1)_{|H|-1}} = \frac{|S|}{d} \cdot \frac{(d-|S|)_{|H|-1}}{(d-1)_{|H|-1}}. && (139)
\end{aligned}$$

Next, let's calculate $\mathbb{P}[i \in S \mid S \cap H = J]$. Since $\omega(i) = \omega(|H| + 1)$ is determined immediately after $\omega(H)$, then given any $\omega(H)$,

$$\mathbb{P}[\omega(i) \in Q \mid \omega(H)] = \frac{|Q \setminus \omega(H)|}{|[d] \setminus \omega(H)|} = \frac{|Q \setminus \omega(H)|}{|[d]| - |\omega(H)|} = \frac{|Q \setminus \omega(H)|}{d - |H|}. \tag{140}$$

Therefore, by the law of total probability and the fact that $|Q| = |S|$,

$$\begin{aligned}
\mathbb{P}[i \in S \mid S \cap H = J] &= \mathbb{P}[\omega(i) \in Q \mid \omega(H) \cap Q = \omega(J)] && (141)\\
&= \mathbb{E}_{\omega(H)} \big[\mathbb{P}[\omega(i) \in Q \mid \omega(H)] \,\big|\, \omega(H) \cap Q = \omega(J) \big] && (142)\\
&= \mathbb{E}_{\omega(H)} \Big[\frac{|Q \setminus \omega(H)|}{d - |H|} \,\Big|\, \omega(H) \cap Q = \omega(J) \Big] && (143)\\
&= \mathbb{E}_{\omega(H)} \Big[\frac{|Q| - |\omega(J)|}{d - |H|} \,\Big|\, \omega(H) \cap Q = \omega(J) \Big] && (144)\\
&= \mathbb{E}_{\omega(H)} \Big[\frac{|S| - |J|}{d - |H|} \,\Big|\, \omega(H) \cap Q = \omega(J) \Big] && (145)\\
&= \frac{|S| - |J|}{d - |H|}. && \square
\end{aligned}$$

Lemma A.8. *For any $1 \le s \le d$ and any $0 < \gamma < 1$,*

$$\frac{\left(d - \left\lfloor \frac{\gamma d}{s} \right\rfloor\right)_{s-1}}{(d-1)_{s-1}} \ge \mathrm{e}^{-\gamma}.$$

Proof. Case 1: If $s = 1$, then

$$\frac{\left(d - \left\lfloor \frac{\gamma d}{s} \right\rfloor\right)_{s-1}}{(d-1)_{s-1}} = \frac{\left(d - \left\lfloor \frac{\gamma d}{s} \right\rfloor\right)_{0}}{(d-1)_{0}} = \frac{1}{1} = 1 \ge \mathrm{e}^{-\gamma}. \tag{146}$$

Case 2: If $s > 1$ and $s \ge \gamma d$, then

$$\frac{\left(d - \left\lfloor \frac{\gamma d}{s} \right\rfloor\right)_{s-1}}{(d-1)_{s-1}} = \prod_{k=0}^{s-2} \frac{d - \left\lfloor \frac{\gamma d}{s} \right\rfloor - k}{d-1-k} = \prod_{k=0}^{s-2} \left(1 + \frac{1 - \left\lfloor \frac{\gamma d}{s} \right\rfloor}{d-1-k}\right) \ge 1 \ge \mathrm{e}^{-\gamma}. \tag{147}$$

Case 3: If $1 < s < \gamma d$, since $\frac{s}{\gamma} > s > s-1$, then

$$\begin{aligned}
\frac{\left(d - \left\lfloor \frac{\gamma d}{s} \right\rfloor\right)_{s-1}}{(d-1)_{s-1}} &= \prod_{k=0}^{s-2} \frac{d - \left\lfloor \frac{\gamma d}{s} \right\rfloor - k}{d-1-k} \ge \prod_{k=0}^{s-2} \frac{d - \frac{\gamma d}{s} - k}{d-1-k} && (148)\\
&= \prod_{k=0}^{s-2} \left(1 - \frac{\gamma}{s}\left(1 - \frac{\frac{s}{\gamma} - (1+k)}{d-1-k}\right)\right) && (149)\\
&\ge \prod_{k=0}^{s-2} \left(1 - \frac{\gamma}{s}\left(1 - \frac{\frac{s}{\gamma} - (s-1)}{d-1-k}\right)\right) && (150)\\
&\ge \prod_{k=0}^{s-2} \left(1 - \frac{\gamma}{s}\right) = \left(1 - \frac{\gamma}{s}\right)^{s-1} && (151)\\
&\ge \lim_{s' \to +\infty} \left(1 - \frac{\gamma}{s'}\right)^{s'-1} = \mathrm{e}^{-\gamma}. && (152)
\end{aligned}$$

The last inequality uses the fact that $\left(1 - \frac{\gamma}{s'}\right)^{s'-1}$ is decreasing w.r.t s' for $0 < \gamma < 1$. □

Now we are ready to prove Lemma 3.3.

Proof of Lemma 3.3. Let $0 < \nu < 1$ be an absolute constant, and let

$$\beta := \min\left\{1, \frac{\rho - \alpha}{1 - \rho}\left(1 - \frac{1}{-W_{-1}\left(-\frac{(\rho-\alpha)\delta}{\mathrm{e}\rho}\right)}\right)\right\}, \qquad \gamma := \frac{(1-\nu)\beta}{C_2^2} < 1, \tag{153}$$

where $W_{-1} : \left[-\frac{1}{\mathrm{e}}, 0\right) \to (-\infty, -1]$ is the (-1) branch of the Lambert product logarithm function.

If $s > \gamma d = \Omega(d)$, then we can simply let $J := [d]$. Otherwise, let

$$n := \left\lfloor \frac{\gamma d}{s} \right\rfloor, \qquad K := \left\lceil \frac{d}{n} \right\rceil, \qquad m := \left\lceil \log_{\frac{1}{1 - \exp(-(1-\nu)/C_2^2)\nu/2}} \frac{\rho}{(\rho - \alpha - (1-\rho)\beta)\delta} \right\rceil. \tag{154}$$

Consider the following procedure, which is equivalent to the procedure in Algorithm 1. Randomly generate m permutations $\omega_1, \ldots, \omega_m$ of $[d]$. Apply the algorithm in Lemma 3.2 independently to each $S_{l,k} := \{i \in [d] : \lceil \omega_l(i)/n \rceil = k\}$ get a $j_{l,k} \in S_{l,k}$ for each $l \in [m]$ and $k \in [K]$. We will show that $J := \{j_{l,k}\}_{l \in [m], k \in [K]}$ is the desired set.

Let

$$H := \operatorname*{arg\,max}_{\substack{I \subseteq [d] \\ |I| = s}} \|\nabla_I f(\boldsymbol{x})\|_2^2, \tag{155}$$

$$H^* := \left\{ i \in H : |\nabla_i f(\boldsymbol{x})| > \sqrt{\frac{\beta}{s}} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2 + \lambda_{1,n} \cdot \epsilon \right\}. \tag{156}$$

We remark that $H^* \subseteq H$ and that the number of large-gradient dimensions H^* can be less than s.

Fix an $l \in [m]$. For any $k \in [K]$ and any $j \in H$, since $|S_{l,k}| \leq n = \lfloor \frac{\gamma d}{s} \rfloor \leq \frac{\gamma d}{s}$, then by Lemma A.7 and Lemma A.8,

$$\mathbb{P}[S_{l,k} \cap H = \{j\}] = \frac{|S_{l,k}|}{d} \cdot \frac{(d-|S_{l,k}|)_{s-1}}{(d-1)_{s-1}} \geq \frac{|S_{l,k}|}{d} \cdot \frac{\left(d - \lfloor \frac{\gamma d}{s} \rfloor\right)_{s-1}}{(d-1)_{s-1}} \tag{157}$$

$$\geq \frac{|S_{l,k}|}{d} \cdot \mathrm{e}^{-\gamma} = \frac{|S_{l,k}|}{d} \cdot \mathrm{e}^{-(1-\nu)\beta/C_2^2} \geq \frac{|S_{l,k}|}{d} \cdot \mathrm{e}^{-(1-\nu)/C_2^2}. \tag{158}$$

Besides that, by Lemma A.7,

$$\mathbb{P}[i \in S_{l,k} \mid S_{l,k} \cap H = \{j\}] = \frac{|S_{l,k}| - |\{j\}|}{d - |H|} = \frac{|S_{l,k}| - 1}{d - s}. \tag{159}$$

Thus,

$$\mathbb{E}[\|\nabla_{S_{l,k} \setminus H} f(\boldsymbol{x})\|_2^2 \mid \{S_{l,k} \cap H = \{j\}\}] \tag{160}$$

$$= \sum_{i \in [d] \setminus H} (\nabla_i f(\boldsymbol{x}))^2 \cdot \mathbb{P}[i \in S_{l,k} \setminus H \mid S_{l,k} \cap H = \{j\}] \tag{161}$$

$$= \sum_{i \in H} (\nabla_i f(\boldsymbol{x}))^2 \cdot 0 + \sum_{i \in [d] \setminus H} (\nabla_i f(\boldsymbol{x}))^2 \cdot \mathbb{P}[i \in S_{l,k} \mid S_{l,k} \cap H = \{j\}] \tag{162}$$

$$= 0 + \sum_{i \in [d] \setminus H} (\nabla_i f(\boldsymbol{x}))^2 \cdot \frac{|S_{l,k}| - 1}{d - s} = \frac{|S_{l,k}| - 1}{d - s} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2 \tag{163}$$

$$\leq \frac{\frac{\gamma d}{s} - 1}{d - s} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2 = \frac{\gamma d - s}{s(d-s)} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2. \tag{164}$$

By Markov's inequality and Eq. (164),

$$\mathbb{P}\Big[\|\nabla_{S_{l,k} \setminus H} f(\boldsymbol{x})\|_2^2 \leq \frac{\beta}{C_2^2 s} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2 \,\Big|\, S_{l,k} \cap H = \{j\}\Big] \tag{165}$$

$$\geq 1 - \frac{\mathbb{E}[\|\nabla_{S_{l,k} \setminus H} f(\boldsymbol{x})\|_2^2 \mid \{S_{l,k} \cap H = \{j\}\}]}{\frac{\beta}{C_2^2 s} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2} \geq 1 - \frac{\frac{\gamma d - s}{s(d-s)} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2}{\frac{\beta}{C_2^2 s} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2} \tag{166}$$

$$= 1 - \frac{C_2^2(\gamma d - s)}{\beta(d-s)} = 1 - \frac{(1-\nu)(\gamma d - s)}{\gamma(d-s)} \geq 1 - (1-\nu) = \nu. \tag{167}$$

Thus, for any $j \in H^*$ and any $k \in [K]$, by the definition of H^* and Eqs. (158) & (167),

$$\mathbb{P}\{j = j_{l,k}\} = \mathbb{P}\{j \in S_{l,k}, \text{ and Lemma 3.2 finds } j\} \tag{168}$$

$$\geq \mathbb{P}\{j \in S_{l,k}, |\nabla_j f(\boldsymbol{x})| > C_2 \|\nabla_{S_{l,k} \setminus \{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S_{l,k}|} \cdot \epsilon, \text{ and Lemma 3.2 finds } j\} \tag{169}$$

$$\geq \mathbb{P}\{j \in S_{l,k}, |\nabla_j f(\boldsymbol{x})| > C_2 \|\nabla_{S_{l,k} \setminus \{j\}} f(\boldsymbol{x})\|_2 + \lambda_{1,|S_{l,k}|} \cdot \epsilon\} \cdot \frac{1}{2} \tag{170}$$

$$\geq \mathbb{P}\{S_{l,k} \cap H = \{j\}, |\nabla_j f(\boldsymbol{x})| > C_2 \|\nabla_{S_{l,k} \setminus H} f(\boldsymbol{x})\|_2 + \lambda_{1,|S_{l,k}|} \cdot \epsilon\} \cdot \frac{1}{2} \tag{171}$$

$$\geq \mathbb{P}\{S_{l,k} \cap H = \{j\}, |\nabla_j f(\boldsymbol{x})| > C_2 \|\nabla_{S_{l,k} \setminus H} f(\boldsymbol{x})\|_2 + \lambda_{1,n} \cdot \epsilon\} \cdot \frac{1}{2} \tag{172}$$

$$\geq \mathbb{P}\Big\{S_{l,k} \cap H = \{j\}, \|\nabla_{S_{l,k} \setminus H} f(\boldsymbol{x})\|_2 \leq \frac{\beta}{C_2^2 s} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2\Big\} \cdot \frac{1}{2} \tag{173}$$

$$= \mathbb{P}[S_{l,k} \cap H = \{j\}] \cdot \mathbb{P}\Big[\|\nabla_{S_{l,k} \setminus H} f(\boldsymbol{x})\|_2^2 \leq \frac{\beta}{C_2^2 s} \|\nabla_{[d] \setminus H} f(\boldsymbol{x})\|_2^2 \,\Big|\, S_{l,k} \cap H = \{j\}\Big] \cdot \frac{1}{2} \tag{174}$$

$$\geq \Big(\frac{|S_{l,k}|}{d} \cdot \mathrm{e}^{-(1-\nu)/C_2^2}\Big) \cdot \nu \cdot \frac{1}{2}, \tag{175}$$

Since $\sum_{k\in[K]}|S_{l,k}| = d$ by definition, then by Eq. (175),

$$\mathbb{P}\{j \in \{j_{l,k}\}_{k\in[K]}\} = \sum_{k\in[K]} \mathbb{P}\{j = j_{l,k}\} \geq \sum_{k\in[K]} \Big(\frac{|S_{l,k}|}{d} \cdot \mathrm{e}^{-(1-\nu)/C_2^2}\Big) \cdot \nu \cdot \frac{1}{2} = \mathrm{e}^{-(1-\nu)/C_2^2} \cdot \frac{\nu}{2}. \tag{176}$$

By Eq. (176) and the definition of m,

$$\begin{aligned}
\mathbb{E}[\|\nabla_{H^*\setminus J} f(\boldsymbol{x})\|_2^2] &= \sum_{j\in H^*} (\nabla_j f(\boldsymbol{x}))^2 \cdot \mathbb{P}\{j \notin J\} && (177)\\
&= \sum_{j\in H^*} (\nabla_j f(\boldsymbol{x}))^2 \prod_{l\in[m]} (1 - \mathbb{P}\{j \in \{j_{l,k}\}_{k\in[K]}\}) && (178)\\
&\leq \sum_{j\in H^*} (\nabla_j f(\boldsymbol{x}))^2 \Big(1 - \mathrm{e}^{-(1-\nu)/C_2^2} \cdot \frac{\nu}{2}\Big)^m && (179)\\
&= \Big(1 - \mathrm{e}^{-(1-\nu)/C_2^2} \cdot \frac{\nu}{2}\Big)^m \|\nabla_{H^*} f(\boldsymbol{x})\|_2^2 && (180)\\
&\leq \frac{(\rho - \alpha - (1-\rho)\beta)\delta}{\rho} \|\nabla_{H^*} f(\boldsymbol{x})\|_2^2. && (181)
\end{aligned}$$

Hence, by Markov's inequality, with probability at least $1-\delta$,

$$\begin{aligned}
\|\nabla_{H^*\setminus J} f(\boldsymbol{x})\|_2^2 &\leq \frac{\mathbb{E}[\|\nabla_{H^*\setminus J} f(\boldsymbol{x})\|_2^2]}{\delta} && (182)\\
&\leq \frac{\rho - \alpha - (1-\rho)\beta}{\rho} \|\nabla_{H^*} f(\boldsymbol{x})\|_2^2 && (183)\\
&\leq \frac{\rho - \alpha - (1-\rho)\beta}{\rho} \|\nabla_{H} f(\boldsymbol{x})\|_2^2. && (184)
\end{aligned}$$

In addition to that, by the definition of H^*, Lemma A.2, and convexity and monotonicity of $\lambda_{1,n}^2$ w.r.t. n,

$$\begin{aligned}
\|\nabla_{H\setminus H^*} f(\boldsymbol{x})\|_2^2 &\leq s\|\nabla_{H\setminus H^*} f(\boldsymbol{x})\|_\infty^2 && (185)\\
&\leq s\Big(\sqrt{\frac{\beta}{s}}\|\nabla_{[d]\setminus H} f(\boldsymbol{x})\|_2 + \lambda_{1,n}\cdot\epsilon\Big)^2 && (186)\\
&= \beta\|\nabla_{[d]\setminus H} f(\boldsymbol{x})\|_2^2 + 2\sqrt{s\beta}\|\nabla_{[d]\setminus H} f(\boldsymbol{x})\|_2\lambda_{1,n}\epsilon + s\lambda_{1,n}^2\epsilon^2 && (187)\\
&\leq \beta\|\nabla_{[d]\setminus H} f(\boldsymbol{x})\|_2^2 + 2\sqrt{s1}L_0\lambda_{1,n}\epsilon + s\lambda_{1,n}^2\epsilon^2 && (188)\\
&\leq \beta\|\nabla_{[d]\setminus H} f(\boldsymbol{x})\|_2^2 + 2L_0\lambda_{1,sn}\epsilon + \lambda_{1,sn}^2\epsilon^2 && (189)\\
&\leq \beta\|\nabla_{[d]\setminus H} f(\boldsymbol{x})\|_2^2 + 2L_0\lambda_{1,\gamma d}\epsilon + \lambda_{1,\gamma d}^2\epsilon^2 && (190)\\
&\leq \beta\|\nabla_{[d]\setminus H} f(\boldsymbol{x})\|_2^2 + 2L_0\lambda_{1,d}\epsilon + \lambda_{1,d}^2\epsilon^2 && (191)\\
&= \beta(\|\nabla f(\boldsymbol{x})\|_2^2 - \|\nabla_H f(\boldsymbol{x})\|_2^2) + \lambda_{2,d}\epsilon + \lambda_{1,d}^2\epsilon^2. && (192)
\end{aligned}$$

It follows from Eqs. (192) & (184) and Assumption A that

$$\begin{aligned}
\|\nabla_J f(\boldsymbol{x})\|_2^2 &\geq \|\nabla_H f(\boldsymbol{x})\|_2^2 - \|\nabla_{H\setminus H^*} f(\boldsymbol{x})\|_2^2 - \|\nabla_{H^*\setminus J} f(\boldsymbol{x})\|_2^2 && (193)\\
&\geq \|\nabla_H f(\boldsymbol{x})\|_2^2 - (\beta(\|\nabla f(\boldsymbol{x})\|_2^2 - \|\nabla_H f(\boldsymbol{x})\|_2^2) + \lambda_{2,d}\epsilon + \lambda_{1,d}^2\epsilon^2) - \frac{\rho-\alpha-(1-\rho)\beta}{\rho}\|\nabla_H f(\boldsymbol{x})\|_2^2 && (194)\\
&= \frac{\alpha+\beta}{\rho}\|\nabla_H f(\boldsymbol{x})\|_2^2 - \beta\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 && (195)\\
&\geq \frac{\alpha+\beta}{\rho}\rho\|\nabla f(\boldsymbol{x})\|_2^2 - \beta\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 && (196)\\
&= \alpha\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2. && (197)
\end{aligned}$$

Since $K = O(\gamma s)$, then the size of J is at most $mK \le C_{\rho,\alpha,\delta} s$ where $C_{\rho,\alpha,\delta} := O(m\gamma)$, and the total number of queries is at most

$$O\Big(\sum_{l\in[m]}\sum_{k\in[K]} \log\log|S_{l,k}|\Big) \le O\Big(mK\log\log\Big\lfloor\frac{\gamma d}{s}\Big\rfloor\Big) = O\Big(C_{\rho,\alpha,\delta} s\log\log\frac{d}{s}\Big). \qquad \square$$

A.5. Proof of Theorem 3.4

While previous ZOO works try to upper-bound $\|\boldsymbol{g} - \nabla f(\boldsymbol{x})\|_2$, here we only lower-bound $\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle$ because ensuring this weaker condition suffices for the $O\big(\frac{1}{T}\big)$ convergence and needs fewer queries than ensuring $\|\boldsymbol{g} - \nabla f(\boldsymbol{x})\|_2$ to be small. This insight also helps us to simplify the proofs of Theorems 3.4 & 4.1.

Proof of Theorem 3.4. Since $\alpha < \rho$, we can pick α' such that $\alpha < \alpha' < \rho$. Let $\delta := 1 - \frac{\alpha}{\alpha'}$.

Define $C_{\rho,\alpha} := C_{\rho,\alpha',\delta}$, where $C_{\rho,\alpha',\delta}$ is the constant in the proof of Lemma 3.3. Using Lemma 3.3 w.r.t. $(\boldsymbol{x}, \alpha', \delta)$, we can find a set $J \subseteq [d]$ of size at most $C_{\rho,\alpha',\delta} s = C_{\rho,\alpha} s$ such that with probability at least $1-\delta$, Eq. (197) holds.

For each dimension $i \in [d] \setminus J$, let $g_i := 0$. For each dimension $j \in J$, recall that

$$g_j := \frac{f(\boldsymbol{x} + \epsilon \boldsymbol{e}_j) - f(\boldsymbol{x})}{\epsilon}. \tag{198}$$

By Lemma A.1, for every $j \in J$,

$$|g_j - \nabla_j f(\boldsymbol{x})| = \frac{|f(\boldsymbol{x}+\epsilon\boldsymbol{e}_j) - f(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \epsilon\boldsymbol{e}_j\rangle|}{\epsilon} \le \frac{L_1\epsilon^2/2}{\epsilon} = \frac{L_1\epsilon}{2}. \tag{199}$$

This implies

$$\|\boldsymbol{g}_J - \nabla_J f(\boldsymbol{x})\|_\infty \le \frac{L_1\epsilon}{2}. \tag{200}$$

It follows that

$$\|\boldsymbol{g}_J - \nabla_J f(\boldsymbol{x})\|_2 \le \sqrt{|J|}\cdot\|\nabla_J f(\boldsymbol{x}) - \boldsymbol{g}\|_\infty \le \sqrt{C_{\rho,\alpha} s}\cdot\frac{L_1\epsilon}{2} = \frac{\sqrt{C_{\rho,\alpha}}L_1}{2}\sqrt{s}\epsilon. \tag{201}$$

Hence,

$$\|\boldsymbol{g}\|_2 = \|\boldsymbol{g}_J\|_2 \le \|\nabla_J f(\boldsymbol{x})\|_2 + \|\boldsymbol{g}_J - \nabla_J f(\boldsymbol{x})\|_2 \le \|\nabla f(\boldsymbol{x})\|_2 + \frac{\sqrt{C_{\rho,\alpha}}L_1}{2}\sqrt{s}\epsilon. \tag{202}$$

Besides that, since $\boldsymbol{g}_{[d]\setminus J} = \boldsymbol{0}$, then by the Cauchy–Schwarz inequality and Eq. (201),

$$\begin{aligned}
\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle &= \langle \nabla_J f(\boldsymbol{x}), \boldsymbol{g}_J\rangle && (203)\\
&= \langle \nabla_J f(\boldsymbol{x}), \nabla_J f(\boldsymbol{x})\rangle - \langle \nabla_J f(\boldsymbol{x}), \nabla_J f(\boldsymbol{x}) - \boldsymbol{g}_J\rangle && (204)\\
&= \|\nabla_J f(\boldsymbol{x})\|_2^2 - \langle \nabla_J f(\boldsymbol{x}), \nabla_J f(\boldsymbol{x}) - \boldsymbol{g}_J\rangle && (205)\\
&\ge \|\nabla_J f(\boldsymbol{x})\|_2^2 - \|\nabla_J f(\boldsymbol{x})\|_2\cdot\|\nabla_J f(\boldsymbol{x}) - \boldsymbol{g}_J\|_2 && (206)\\
&\ge \|\nabla_J f(\boldsymbol{x})\|_2^2 - \|\nabla f(\boldsymbol{x})\|_2\cdot\|\nabla_J f(\boldsymbol{x}) - \boldsymbol{g}_J\|_2 && (207)\\
&\ge \|\nabla_J f(\boldsymbol{x})\|_2^2 - L_0\cdot\Big(\frac{\sqrt{C_{\rho,\alpha}}L_1}{2}\sqrt{s}\epsilon\Big) && (208)\\
&= \|\nabla_J f(\boldsymbol{x})\|_2^2 - \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon && (209)\\
&\ge -\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon. && (210)
\end{aligned}$$

Furthermore, recall that Lemma 3.3 succeeds with probability at least $1-\delta$:

$$\mathbb{P}[\|\nabla_J f(\boldsymbol{x})\|_2^2 \geq \alpha'\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 \mid \boldsymbol{x}] \geq 1-\delta. \tag{211}$$

Then by Eq. (209), the event

$$E := \left[\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle \geq \alpha'\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 - \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon\right] \tag{212}$$

has probability

$$\mathbb{P}[E \mid \boldsymbol{x}] \geq \mathbb{P}\big[\|\nabla_J f(\boldsymbol{x})\|_2^2 \geq \alpha'\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 \,\big|\, \boldsymbol{x}\big] \geq 1-\delta. \tag{213}$$

Finally, by the decomposition $1 = 1_E + 1_{E^\mathsf{c}}$,

$$\mathbb{E}[\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle \mid \boldsymbol{x}] = \mathbb{E}[\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle 1_E \mid \boldsymbol{x}_t] + \mathbb{E}[\langle \nabla f(\boldsymbol{x}), \boldsymbol{g}\rangle 1_{E^\mathsf{c}} \mid \boldsymbol{x}] \tag{214}$$

$$\geq \mathbb{E}\left[\left(\alpha'\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 - \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon\right)1_E \,\middle|\, \boldsymbol{x}\right] + \mathbb{E}\left[-\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon 1_{E^\mathsf{c}} \,\middle|\, \boldsymbol{x}\right] \tag{215}$$

$$= \left(\alpha'\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 - \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon\right)\cdot \mathbb{P}[E \mid \boldsymbol{x}] - \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon \cdot \mathbb{P}[E^\mathsf{C} \mid \boldsymbol{x}] \tag{216}$$

$$\geq \left(\alpha'\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{2,d}\epsilon - \lambda_{1,d}^2\epsilon^2 - \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon\right)\cdot (1-\delta) - \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon\cdot\delta \tag{217}$$

$$= \alpha'(1-\delta)\|\nabla f(\boldsymbol{x})\|_2^2 - \left((1-\delta)\lambda_{2,d} + \frac{\sqrt{C_{\rho,\alpha}}}{2}L_0L_1\sqrt{s}\right)\epsilon - (1-\delta)\lambda_{1,d}^2\epsilon^2 \tag{218}$$

$$= \alpha\|\nabla f(\boldsymbol{x})\|_2^2 - \lambda_{3,d,s}\epsilon - \lambda_{4,d}\epsilon^2, \tag{219}$$

with $\lambda_{3,d,s} = O(\lambda_{2,d} + L_0L_1\sqrt{s})$ and $\lambda_{4,d} = O(\lambda_{1,d}^2)$. □

A.6. Proof of Theorem 4.1

With the help of Theorem 3.4, a standard argument (e.g., Section 1.2.3, Nesterov, 2018) shows the $O\big(\frac{1}{T}\big)$ rate of convergence.

Proof of Theorem 4.1. Pick $0 < \alpha < \rho$, and apply Theorem 3.4 with (ρ, α) to get gradient estimates $\{\boldsymbol{g}_t\}_{t\geq 1}$. Let $C_{\rho,\alpha}$ denote the constant in the proof of Theorem 3.4, and let $C_\rho := C_{\rho,\alpha}$.

For every $t \geq 1$, by Lemma A.1 and Theorem 3.4,

$$f(\boldsymbol{x}_{t+1}) = f(\boldsymbol{x}_t - \eta\boldsymbol{g}_t) \leq f(\boldsymbol{x}_t) - \eta\langle\nabla f(\boldsymbol{x}_t), \boldsymbol{g}_t\rangle + \frac{L_1\eta^2}{2}\|\boldsymbol{g}_t\|_2^2 \tag{220}$$

$$\leq f(\boldsymbol{x}_t) - \eta\langle\nabla f(\boldsymbol{x}_t), \boldsymbol{g}_t\rangle + \frac{L_1\eta^2}{2}\left(\|\nabla f(\boldsymbol{x}_t)\|_2 + \frac{\sqrt{C_{\rho,\alpha}}L_1}{2}\sqrt{s}\epsilon\right)^2 \tag{221}$$

$$= f(\boldsymbol{x}_t) - \eta\langle\nabla f(\boldsymbol{x}_t), \boldsymbol{g}_t\rangle + \frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2 + \frac{\sqrt{C_\rho}\eta^2L_1^2\|\nabla f(\boldsymbol{x}_t)\|_2}{2}\sqrt{s}\epsilon_t + \frac{C_\rho\eta^2L_1^3}{8}s\epsilon_t^2 \tag{222}$$

$$\leq f(\boldsymbol{x}_t) - \eta\langle\nabla f(\boldsymbol{x}_t), \boldsymbol{g}_t\rangle + \frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2 + \frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\epsilon_t + \frac{C_\rho\eta^2L_1^3}{8}s\epsilon_t^2. \tag{223}$$

Rearranging the terms gives

$$\eta\langle\nabla f(\boldsymbol{x}_t), \boldsymbol{g}_t\rangle - \frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2 \leq (f(\boldsymbol{x}_t) - f(\boldsymbol{x}_{t+1})) + \frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\epsilon_t + \frac{C_\rho\eta^2L_1^3}{8}s\epsilon_t^2. \tag{224}$$

One the one hand, by a telescoping sum of (224),

$$\sum_{t=1}^{T}\left(\eta\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle-\frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right) \tag{225}$$

$$\leq \sum_{t=1}^{T}(f(\boldsymbol{x}_t)-f(\boldsymbol{x}_{t+1}))+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\sum_{t=1}^{T}\epsilon_t+\frac{C_\rho\eta^2L_1^3}{8}s\sum_{t=1}^{T}\epsilon_t^2 \tag{226}$$

$$= f(\boldsymbol{x}_1)-f(\boldsymbol{x}_{T+1})+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\sum_{t=1}^{T}\epsilon_t+\frac{C_\rho\eta^2L_1^3}{8}s\sum_{t=1}^{T}\epsilon_t^2 \tag{227}$$

$$\leq f(\boldsymbol{x}_1)-f_*+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\sum_{t=1}^{T}\epsilon_t+\frac{C_\rho\eta^2L_1^3}{8}s\sum_{t=1}^{T}\epsilon_t^2. \tag{228}$$

On the other hand, by the law of total expectation and Eq. (219) in Theorem 3.4,

$$\mathbb{E}\left[\sum_{t=1}^{T}\left(\eta\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle-\frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right)\right] \tag{229}$$

$$= \sum_{t=1}^{T}\mathbb{E}\left[\eta\,\mathbb{E}[\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle\mid\boldsymbol{x}_t]-\frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right] \tag{230}$$

$$\geq \sum_{t=1}^{T}\mathbb{E}\left[\eta(\alpha\|\nabla f(\boldsymbol{x}_t)\|_2^2-\lambda_{3,d,s}\epsilon_t-\lambda_{4,d}\epsilon_t^2)-\frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right] \tag{231}$$

$$= \left(\alpha\eta-\frac{L_1\eta^2}{2}\right)\mathbb{E}\left[\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right]-\eta\lambda_{3,d,s}\sum_{t=1}^{T}\epsilon_t-\eta\lambda_{4,d}\sum_{t=1}^{T}\epsilon_t^2. \tag{232}$$

Let the step size $\eta:=\frac{\alpha}{L_1}$, so $\alpha\eta-\frac{L_1\eta^2}{2}=\frac{\alpha^2}{2L_1}$. Combining Eqs. (228) & (232) gives

$$\frac{\alpha^2}{2L_1}\mathbb{E}\left[\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right]-\eta\lambda_{3,d,s}\sum_{t=1}^{T}\epsilon_t-\eta\lambda_{4,d}\sum_{t=1}^{T}\epsilon_t^2\leq f(\boldsymbol{x}_1)-f_*+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\sum_{t=1}^{T}\epsilon_t+\frac{C_\rho\eta^2L_1^3}{8}s\sum_{t=1}^{T}\epsilon_t^2.$$

Rearranging the terms gives

$$\mathbb{E}\left[\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right]\leq\frac{2L_1}{\alpha^2}(f(\boldsymbol{x}_1)-f_*)+\frac{2L_1\left(\eta\lambda_{3,d,s}+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\right)}{\alpha^2}\sum_{t=1}^{T}\epsilon_t+\frac{2L_1\left(\eta\lambda_{4,d}+\frac{C_\rho\eta^2L_1^3}{8}s\right)}{\alpha^2}\sum_{t=1}^{T}\epsilon_t^2. \tag{233}$$

For each $t\geq1$, let $\epsilon_t:=\epsilon_1/t^2$ where

$$\epsilon_1:=\frac{-\frac{\pi^2}{6}\cdot\frac{2L_1\left(\eta\lambda_{3,d,s}+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\right)}{\alpha^2}+\sqrt{\left(\frac{\pi^2}{6}\cdot\frac{2L_1\left(\eta\lambda_{3,d,s}+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\right)}{\alpha^2}\right)^2+4\cdot\frac{\pi^4}{90}\cdot\frac{2L_1\left(\eta\lambda_{4,d}+\frac{C_\rho\eta^2L_1^3s}{8}\right)}{\alpha^2}\cdot\Delta}}{2\cdot\frac{\pi^4}{90}\cdot\frac{2L_1\left(\eta\lambda_{4,d}+\frac{C_\rho\eta^2L_1^3s}{8}\right)}{\alpha^2}}. \tag{234}$$

It is clear that $\epsilon_1>0$. Since $\sum_{t=1}^{T}\frac{1}{t^2}\leq\frac{\pi^2}{6}$, and $\sum_{t=1}^{T}\frac{1}{t^4}\leq\frac{\pi^4}{90}$, then by Eq. (233),

$$\mathbb{E}\left[\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t)\|_2^2\right]\leq\frac{2L_1}{\alpha^2}(f(\boldsymbol{x}_1)-f_*)+\frac{2L_1\left(\eta\lambda_{3,d,s}+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\right)}{\alpha^2}\cdot\frac{\pi^2}{6}\epsilon_1+\frac{2L_1\left(\eta\lambda_{4,d}+\frac{C_\rho\eta^2L_1^3}{8}s\right)}{\alpha^2}\cdot\frac{\pi^4}{90}\epsilon_1^2 \tag{235}$$

$$= \frac{2L_1}{\alpha^2}(f(\boldsymbol{x}_1)-f_*)+\Delta. \tag{236}$$

It follows that

$$\mathbb{E}\Big[\min_{t=1,\dots,T}\|\nabla f(\boldsymbol{x}_t)\|_2^2\Big] \le \mathbb{E}\bigg[\frac{1}{T}\sum_{t=1}^{T}\|\nabla f(\boldsymbol{x}_t)\|_2^2\bigg] \le \frac{\frac{2L_1}{\alpha^2}(f(\boldsymbol{x}_1)-f_*)+\Delta}{T}. \qquad \square$$

A.7. Proof of Theorem 4.2

Note that the proof of Theorem 4.1 implies that $\mathbb{E}[\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2]$ is bounded. Thus, we can give a high-probability convergence analysis through a similar argument with the proof of Theorem 4.1 and by applying Markov's inequality w.r.t. a constructed non-negative random variable.

Proof. Let $\alpha := L_1\eta < \rho$, and let $C_{\rho,\alpha}$ denote the constant in the proof of Theorem 3.4. For each $t \ge 1$, let $\epsilon_t := \epsilon_1/t^2$ with $\epsilon_1 > 0$ to be determined later.

Let $\mu_1 := \frac{2L_1\left(\eta\lambda_{3,d,s}+\frac{\sqrt{C_{\rho,\alpha}}\eta^2L_0L_1^2}{2}\sqrt{s}\right)}{\alpha^2}\cdot\frac{\pi^2}{6}$ and $\mu_2 := \frac{2L_1\left(\eta\lambda_{4,d}+\frac{C_{\rho,\alpha}\eta^2L_1^3}{8}s\right)}{\alpha^2}\cdot\frac{\pi^4}{90}$. Then by Eq. (235),

$$\mathbb{E}\bigg[\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2\bigg] \le \frac{2L_1}{\alpha^2}(f(\boldsymbol{x}_1)-f_*)+\mu_1\epsilon_1+\mu_2\epsilon_1^2 = \frac{2}{L_1\eta^2}(f(\boldsymbol{x}_1)-f_*)+\mu_1\epsilon_1+\mu_2\epsilon_1^2 < \infty. \tag{237}$$

Let J_t be the candidate set in step t. Then by the Cauchy–Schwarz inequality and Eq. (201),

$$\begin{aligned}
\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle &= \langle\nabla_{J_t}f(\boldsymbol{x}_t),(\boldsymbol{g}_t)_{J_t}\rangle && (238)\\
&= \langle\nabla_{J_t}f(\boldsymbol{x}_t),\nabla_{J_t}f(\boldsymbol{x}_t)\rangle - \nabla_{J_t}f(\boldsymbol{x}_t),\nabla_{J_t}f(\boldsymbol{x}_t)-(\boldsymbol{g}_t)_{J_t}\rangle && (239)\\
&= \|\nabla_{J_t}f(\boldsymbol{x}_t)\|_2^2 - \langle\nabla_{J_t}f(\boldsymbol{x}_t),\nabla_{J_t}f(\boldsymbol{x}_t)-(\boldsymbol{g}_t)_{J_t}\rangle && (240)\\
&\le \|\nabla_{J_t}f(\boldsymbol{x}_t)\|_2^2 + \|\nabla_{J_t}f(\boldsymbol{x}_t)\|_2\cdot\|\nabla_{J_t}f(\boldsymbol{x}_t)-(\boldsymbol{g}_t)_{J_t}\|_2 && (241)\\
&\le \|\nabla f(\boldsymbol{x}_t)\|_2^2 + \|\nabla f(\boldsymbol{x})\|_2\cdot\|\nabla_{J_t}f(\boldsymbol{x}_t)-(\boldsymbol{g}_t)_{J_t}\|_2 && (242)\\
&\le \|\nabla f(\boldsymbol{x}_t)\|_2^2 + L_0\cdot\Big(\frac{\sqrt{C_{\rho,\alpha}}L_1}{2}\sqrt{s}\epsilon_t\Big) && (243)\\
&= \|\nabla f(\boldsymbol{x}_t)\|_2^2 + \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon_t. && (244)
\end{aligned}$$

This implies that

$$\begin{aligned}
&\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2 - \sum_{t=1}^{\infty}\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle + \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\cdot\frac{\pi^2}{6}\epsilon_1 && (245)\\
=&\sum_{t=1}^{\infty}\Big(\|\nabla f(\boldsymbol{x}_t)\|_2^2 - \langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle + \frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\epsilon_t\Big) \ge 0. && (246)
\end{aligned}$$

Thus, by Markov's inequality and Eq. (219), with probability at least $1-\beta$,

$$\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2-\sum_{t=1}^{\infty}\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle+\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\cdot\frac{\pi^2}{6}\epsilon_1 \tag{247}$$

$$\leq\frac{1}{\beta}\mathbb{E}\bigg[\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2-\sum_{t=1}^{\infty}\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle+\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\cdot\frac{\pi^2}{6}\epsilon_1\bigg] \tag{248}$$

$$=\frac{1}{\beta}\bigg(\mathbb{E}\bigg[\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2\bigg]-\mathbb{E}\bigg[\sum_{t=1}^{\infty}\mathbb{E}[\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle\mid\boldsymbol{x}_t]\bigg]+\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\cdot\frac{\pi^2}{6}\epsilon_1\bigg) \tag{249}$$

$$\leq\frac{1}{\beta}\bigg(\mathbb{E}\bigg[\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2\bigg]-\mathbb{E}\bigg[\sum_{t=1}^{\infty}(\alpha\|\nabla f(\boldsymbol{x}_t)\|_2^2-\lambda_{3,d,s}\epsilon_t-\lambda_{4,d}\epsilon_t^2)\bigg]+\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\cdot\frac{\pi^2}{6}\epsilon_1\bigg) \tag{250}$$

$$=\frac{1}{\beta}\bigg((1-\alpha)\mathbb{E}\bigg[\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2\bigg]+\Big(\lambda_{3,d,s}+\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\Big)\cdot\frac{\pi^2}{6}\epsilon_1+\lambda_{4,d}\cdot\frac{\pi^4}{90}\epsilon_1^2\bigg) \tag{251}$$

$$\leq\frac{1}{\beta}\bigg((1-\alpha)\Big(\frac{2}{L_1\eta^2}(f(\boldsymbol{x}_1)-f_*)+\mu_1\epsilon_1+\mu_2\epsilon_1^2\Big)+\Big(\lambda_{3,d,s}+\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\Big)\cdot\frac{\pi^2}{6}\epsilon_1+\lambda_{4,d}\cdot\frac{\pi^4}{90}\epsilon_1^2\bigg). \tag{252}$$

With $\mu_3:=\frac{\mu_1}{\beta}+\big(\lambda_{3,d,s}+\big(\frac{1}{\beta}-1\big)\frac{\sqrt{C_{\rho,\alpha}}L_0L_1}{2}\sqrt{s}\big)\cdot\frac{\pi^2}{6}$ and $\mu_4:=\frac{\mu_2}{\beta}+\frac{\lambda_{4,d}}{\beta}\cdot\frac{\pi^4}{90}$, rearranging the terms in Eq. (252) gives

$$\sum_{t=1}^{\infty}\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle\geq\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2-\frac{2(1-\alpha)}{L_1\eta^2\beta}(f(\boldsymbol{x}_1)-f_*)-\mu_3\epsilon_1-\mu_4\epsilon_1^2 \tag{253}$$

$$=\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2-\frac{2(1-L_1\eta)}{L_1\eta^2\beta}(f(\boldsymbol{x}_1)-f_*)-\mu_3\epsilon_1-\mu_4\epsilon_1^2. \tag{254}$$

By Eqs. (254) & (228),

$$\Big(\eta-\frac{L_1\eta^2}{2}\Big)\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2=\sum_{t=1}^{\infty}\eta\|\nabla f(\boldsymbol{x}_t)\|_2^2-\sum_{t=1}^{\infty}\frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2 \tag{255}$$

$$\leq\sum_{t=1}^{\infty}\eta\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle+\frac{2(1-L_1\eta)}{L_1\eta\beta}(f(\boldsymbol{x}_1)-f_*)+\eta\mu_3\epsilon_1+\eta\mu_4\epsilon_1^2-\sum_{t=1}^{\infty}\frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2 \tag{256}$$

$$=\sum_{t=1}^{\infty}\Big(\eta\langle\nabla f(\boldsymbol{x}_t),\boldsymbol{g}_t\rangle-\frac{L_1\eta^2}{2}\|\nabla f(\boldsymbol{x}_t)\|_2^2\Big)+\frac{2(1-L_1\eta)}{L_1\eta\beta}(f(\boldsymbol{x}_1)-f_*)+\eta\mu_3\epsilon_1+\eta\mu_4\epsilon_1^2 \tag{257}$$

$$\leq f(\boldsymbol{x}_1)-f_*+\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\sum_{t=1}^{\infty}\epsilon_t+\frac{C_\rho\eta^2L_1^3}{8}s\sum_{t=1}^{\infty}\epsilon_t^2+\frac{2(1-L_1\eta)}{L_1\eta\beta}(f(\boldsymbol{x}_1)-f_*)+\eta\mu_3\epsilon_1+\eta\mu_4\epsilon_1^2 \tag{258}$$

$$=\Big(1+\frac{2(1-L_1\eta)}{L_1\eta\beta}\Big)(f(\boldsymbol{x}_1)-f_*)+\Big(\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\cdot\frac{\pi^2}{6}+\eta\mu_3\Big)\epsilon_1+\Big(\frac{C_\rho\eta^2L_1^3}{8}s\cdot\frac{\pi^4}{90}+\eta\mu_4\Big)\epsilon_1^2. \tag{259}$$

With $\mu_5:=\frac{\frac{\sqrt{C_\rho}\eta^2L_0L_1^2}{2}\sqrt{s}\cdot\frac{\pi^2}{6}+\eta\mu_3}{\eta-\frac{L_1\eta^2}{2}}$ and $\mu_6:=\frac{\frac{C_\rho\eta^2L_1^3}{8}s\cdot\frac{\pi^4}{90}+\eta\mu_4}{\eta-\frac{L_1\eta^2}{2}}$, dividing Eq. (259) by $\eta-\frac{L_1\eta^2}{2}$ gives

$$\sum_{t=1}^{\infty}\|\nabla f(\boldsymbol{x}_t)\|_2^2\leq\frac{1+\frac{2(1-L_1\eta)}{L_1\eta\beta}}{\eta-\frac{L_1\eta^2}{2}}(f(\boldsymbol{x}_1)-f_*)+\mu_5\epsilon_1+\mu_6\epsilon_1^2. \tag{260}$$

Table 3. Summary of hyperparameters (q: queries per step; N/A means that q is decided by the algorithm).

Type	Method	DISTANCE			MAGNITUDE			ATTACK		
		η	q	T	η	q	T	η	q	T
Full Gradient	RS	0.0001	N/A	3000	0.0005	N/A	775	0.002	N/A	10000
	TPGE	0.0001	N/A	2000	0.0005	N/A	517	0.002	N/A	6667
	RSPG	0.005	58	100	0.02	31	50	0.1	199	100
	ZO-signSGD	0.001	58	100	0.002	31	50	0.001	199	100
	ZO-AdaMM	0.001	N/A	3000	0.001	N/A	775	0.002	N/A	10000
	GLD	0.001	4	1200	0.002	4	310	0.01	4	5000
Sparse Gradient	LASSO	0.005	58	100	0.02	31	50	0.2	199	100
	SparseSZO	0.01	58	100	0.05	31	50	0.2	199	100
	TruncZSGD	0.02	58	100	0.2	31	50	0.02	199	100
	ZORO	0.02	58	100	0.5	31	50	0.0002	199	100
	ZO-BCD	0.5	58	100	0.5	31	50	0.002	199	100
	SZOHT	0.005	58	100	0.5	31	50	0.2	199	100
	GraCe (ours)	0.5	N/A	100	0.5	N/A	50	0.5	N/A	100

Table 4. Performance of our method w.r.t. various $\widehat{s}$ on DISTANCE with true sparsity $s = 10$.

$s=6$	$s=7$	$s=8$	$s=9$	$s=10$	$s=11$	$s=12$	$s=13$	$s=14$	$s=15$
0.0094	0.0097	0.0053	0.0053	0.0051	0.0045	0.0040	0.0040	0.0056	0.0036

Let $\epsilon_1 := \frac{-\mu_6+\sqrt{\mu_6^2+4\mu_5\Delta}}{2\mu_5} > 0$, which has $\mu_5\epsilon_1 + \mu_6\epsilon_1^2 = \Delta$. It follows that for all $T \geq 1$ simultaneously,

$$\min_{t=1,\dots,T} \|\nabla f(\boldsymbol{x}_t)\|_2^2 \leq \frac{1}{T}\sum_{t=1}^{T} \|\nabla f(\boldsymbol{x}_t)\|_2^2 \leq \frac{1}{T}\sum_{t=1}^{\infty} \|\nabla f(\boldsymbol{x}_t)\|_2^2 \tag{261}$$
$$\leq \frac{\frac{1+\frac{2(1-L_1\eta)}{L_1\eta\beta}}{\eta-\frac{L_1\eta^2}{2}}(f(\boldsymbol{x}_1) - f_*) + \Delta}{T}.$$

□

B. Experiments (Continued)

B.1. Additional Implementation Details

The hyperparameters of all methods are summarized in Table 3. To ensure fair comparison, we let all methods have the query budget $(q+1)T$ at least that of GraCe. Since RS, TPGE, ZO-AdaMM, and GLD use $O(1)$ queries per step, we adjust their number T of steps so that their total number of queries matches that of our GraCe; for other methods, we use the same number of queries per step as that of our GraCe. For each method, we choose the best step size η among $\{0.5, 0.2, 0.1, 0.05, 0.02, 0.01, \dots\}$.

B.2. Performance under Inexact Sparsity

In practice, when the true sparsity s is unknown, we may use an estimated sparsity $\widehat{s}$ instead of the true sparsity s. We show in Table 4 the normalized objective of our method w.r.t. various $\widehat{s}$ on DISTANCE with true sparsity $s = 10$. We can see that even when $\widehat{s}$ is slightly smaller than the true sparsity $s = 10$ (especially when $\widehat{s} \geq 8$), our method still achieves strong performance. Furthermore, slight overestimation of $\widehat{s}$ can even improve the performance. The results demonstrate that our method is fairly robust w.r.t. inexact sparsity. Therefore, as long as the estimated $\widehat{s}$ is not too far from the true sparsity s, we can expect that our method should still have good performance.

Table 5. Performance under non-sparse gradients.

FDSA	RS	TPGE	RSPG	ZO-signSGD	ZO-AdaMM	GLD
0.0008	0.0021	0.0056	0.0012	0.1002	0.0023	0.3392
LASSO	SparseSZO	TruncZSGD	ZORO	ZO-BCD	SZOHT	GraCe (ours)
0.0013	0.0448	0.0017	0.0792	0.0187	0.0015	**0.0007**

Table 6. Worst-case numbers of queries per step under various d and s.

$\backslash$	$d = 10^2$	$d = 10^3$	$d = 10^4$	$d = 10^5$	$d = 10^6$	$d = 10^7$	$d = 10^8$
$s = 1$	11	15	15	17	19	19	19
$s = 2$	16	19	22	22	28	28	28
$s = 3$	26	26	36	36	46	46	46
$s = 4$	25	31	43	43	55	55	55
$s = 5$	33	41	57	57	73	73	73

B.3. Performance under Non-Sparse Gradients

We further investigate the performance of our method when $s = \Theta(d)$ (i.e., the function has non-sparse gradients). Note that GraCe reduces to the classic method FDSA (Kiefer & Wolfowitz, 1952) when $s = d$, so we also include FDSA in our comparison. Here we use DISTANCE with $d = 10000$ and $s = 2500$ and report the normalized objective under a query budget of 250000. The results are presented in Table 5. From the table, we can see that our GraCe and FDSA achieve the best performance and significantly outperform all other methods. Therefore, it is actually beneficial that our method reduces to FDSA when $s = \Theta(d)$. The results are not surprising because it has been recently shown that $\Omega(d)$ queries are required if the gradient is not sparse (Theorem 1, Alabdulkareem & Honorio, 2021), which implies that any ZOO method should not have significant advantage over FDSA under non-sparse gradients in the worst case.

B.4. Validation of Query Complexity

To validate the query complexity $O\big(s \log\log \frac{d}{s}\big)$, we present the worst-case numbers of queries under various d and s in Table 6. From the table, we can see that the number of queries grows very slowly w.r.t. d. For instance, GraCe needs at most 19 queries for $d = 10^8$ and $s = 1$. This provides strong evidence that the number of queries does scale as $O\big(s \log\log \frac{d}{s}\big)$.