

Zero-Order Stochastic Conditional Gradient Sliding Method for Non-smooth Convex Optimization^{*}

Aleksandr Lobanov^{1,2,3}[0000-0003-1620-9581], Anton Anikin⁴[0000-0002-7681-2481], Alexander Gasnikov^{1,5,6}[0000-0002-7386-039X], Alexander Gornov⁴[0000-0002-8340-5729], and Sergey Chukanov⁷

¹ Moscow Institute of Physics and Technology, Dolgoprudny, Russia
 lobanov.av@mipt.ru, gasnikov.av@mipt.ru

² ISP RAS Research Center for Trusted Artificial Intelligence, Moscow, Russia

³ Moscow Aviation Institute, Moscow, Russia

⁴ Matrosov Institute for System Dynamics and Control Theory, Irkutsk, Russia
 anikin@icc.ru, gornov@icc.ru

⁵ Institute for Information Transmission Problems RAS, Moscow, Russia

⁶ Caucasus Mathematical Center, Adyghe State University, Maikop, Russia

⁷ Federal Research Center "Computer Science and Control" RAS, Moscow, Russia
 chukanov47@mail.ru

Abstract. The conditional gradient idea proposed by Marguerite Frank and Philip Wolfe in 1956 was so well received by the community that new algorithms (also called Frank–Wolfe type algorithms) are still being actively created. In this paper, we study a non-smooth stochastic convex optimization problem with constraints. Using a smoothing technique and based on an accelerated batched first-order Stochastic Conditional Gradient Sliding method, we propose a novel gradient-free Frank–Wolfe type algorithm called Zero-Order Stochastic Conditional Gradient Sliding (ZO-SCGS). This algorithm is robust not only for the class of non-smooth problems, but surprisingly also for the class of smooth black box problems, outperforming the SOTA algorithms in the smooth case in term oracle calls. In practical experiments we confirm our theoretical results.

Keywords: Frank–Wolfe type algorithms · Non-smooth convex optimization · Gradient-free method.

1 Introduction

The history of the conditional gradient method begins with the Frank–Wolfe algorithm proposed in 1956 [16]. Marguerite Frank and Philip Wolfe proposed an alternative to the gradient descent method for solving a class of quadratic constrained optimization problems that uses linear optimization on a convex

^{*} The work was supported by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye) 075-00337-20-03, project No. 0714-2020-0005.

compact set, avoiding projection. A little later in 1966, Evgenii Levitin and Boris Polyak in [34] investigated the Frank–Wolfe method (named Conditional Gradient), obtaining the rate of convergence and showed that this rate is optimal for the class of smooth convex problems and for all algorithms that use linear minimization oracle. Since then, the conditional gradient algorithm has gained much interest in the community, because in some cases it is computationally cheaper to solve the linear minimization problem over the feasible set (thereby guaranteeing a presence over the feasible set) than to perform a projection over the feasible set. Currently, the conditional gradient method is actively used in solving practical problems of network routing [35, 38, 21], matrix completion [17, 20], as well as in problems of machine learning [30, 40], federated learning [12], on-line optimization [28, 9, 18], standard optimization [24, 39, 19] and huge-scale optimization [7, 11, 3].

However, as far as we know, there are no gradient-free algorithms (based on the conditional gradient method) to solve the black box problem in the non-smooth case. Where the black box problem means that only the zero-order oracle [42] is available to us, i.e. we have access to the value of the objective function, not its gradient. This class of problems is a particular case of the practical problems above, when the gradient calculation procedure is too expensive [43, 1] or not available at all [8, 14]. Already in November 2022, a survey appeared [23], which provides various techniques for creating optimal gradient-free algorithms (based generally on accelerated batched first-order methods) to solve the black-box problem. The optimal for a gradient-free algorithm is usually understood by three criteria: iteration complexity, oracle complexity, and maximum level of adversary noise. Thus, by choosing the accelerated batched conditional gradient method and using the smoothing technique from the survey, it is possible to develop a gradient-free algorithm to solve black-box problem in non-smooth case.

In this paper, we focus on black-box problems in the non-smooth case, namely, non-smooth convex stochastic optimization problems. To solve this problem, we use a smoothing scheme approach with l_2 randomization. Based on the accelerated batched conditional gradient method, also known as the Stochastic Conditional Gradient Sliding Method from [33], we create an algorithm and derive optimal estimates: iteration complexity, oracle complexity, and maximum adversary noise level. As far as we know, this is the first gradient-free algorithm for solving a non-smooth convex optimization problem. We show in theory that Zero-Order Stochastic Conditional Gradient Sliding Method outperforms the oracle complexity of gradient-free algorithms (which are state of the art algorithms) in a smooth setting, which is a surprising fact. In practical experiments we confirm our theoretical results.

1.1 Our Contributions

Our contributions can be summarized as:

- We present the first gradient-free algorithm based on the conditional gradient method "Zero-Order Stochastic Conditional Gradient Sliding Method" (ZO-

SCGS) for solving a non-smooth convex stochastic optimization problem with constraints.

- Our theoretical results show that the algorithm is robust for black-box problems not only in the non-smooth case, but also for the smooth setting case. That is, our algorithm outperforms state of the art algorithms on oracle calls. In particular, the SOTA algorithm Zero-Order Conditional Gradient Method (ZSCG) from [5] has an estimation of oracle complexity $\sim \varepsilon^{-3}$, while our algorithm has an estimation of oracle complexity $\sim \varepsilon^{-2}$.
- We empirically test our theoretical results by comparing the Zero-Order Stochastic Conditional Gradient Sliding Method (ZO-SCGS) with the Zero-Order Conditional Gradient Method (ZSCG) on a model case in a smooth setting. We explain the reason for the advantage of the proposed algorithm.

1.2 Paper Organization

This paper has the following structure. In Section 2 we provide related works. In Section 3 we consider the formulation of the problem. We present the novel gradient-free algorithm in Section 4. In Section 5 we discuss the theoretical results obtained. We verify our results with a model experiment in Section 6. While Section 7 concludes the paper. We provide a detailed proof of the Theorem 1 in the supplementary materials (Appendix A)[†].

2 Related Works

Conditional gradient methods. There are many works [24, 39, 33, 5, 29, 49, 50, 13, 37, 10] in the field of conditional gradient methods research. The latest research results in this area are presented in a recent survey on conditional gradient methods [6]. For instance, the Stochastic Frank–Wolf algorithm from [29], which is a generalization of the Frank–Wolf algorithm to stochastic optimization by replacing the gradient in the update with its stochastic approximation, requires $\sim \varepsilon^{-3}$ calls of stochastic gradients and performing $\sim \varepsilon^{-1}$ linear optimization. Also, for instance, the Stochastic Away Frank–Wolfe algorithm from [24], which is derived from combining the Away-Step Stochastic Frank–Wolfe algorithm [27] and the Pairwise Stochastic Frank–Wolfe algorithm [32], requires $\sim \varepsilon^{-4} \log^6(\varepsilon^{-1})$ calls of stochastic gradients and performing $\sim \log(\varepsilon^{-1})$ linear optimization. In another work [39], the Momentum Stochastic Frank–Wolf algorithm, which is obtained from the Stochastic Frank–Wolfe algorithm by replacing the gradient estimator with the momentum estimator, requires $\sim \varepsilon^{-3}$ calls of stochastic gradients and linear optimization. And in [33] the Stochastic Conditional Gradient Sliding algorithm was proposed, which is an accelerated batched method and requires $\sim \varepsilon^{-2}$ calls of stochastic gradients and performing $\sim \varepsilon^{-1}$ linear optimization. The above algorithms solve the problem of convex stochastic optimization and are first-order methods, but the Zeroth-Order

[†] The full version of this article, which includes the Appendix A can be found at the following link: <https://arxiv.org/abs/2303.02778>.

Stochastic Conditional Gradient Method from [5], which solves the black box problem in the smooth case, requires $\sim \varepsilon^{-3}$ calls of stochastic gradients and performing $\sim \varepsilon^{-1}$ linear optimization. In this paper, we choose the accelerated batched first order method: Stochastic Conditional Gradient Sliding algorithm from [33] as the basis for creating a novel gradient-free algorithm, since it has the best number of stochastic gradient calls presented. We will compare the efficiency of our algorithm to the Zeroth-Order Stochastic Conditional Gradient Method from [5], which is one of the SOTA algorithms.

Gradient-free methods. The research field of gradient-free algorithms can be traced back to at least 1952 [31]. Recent works [14, 47, 15, 4, 22, 2, 36] are heavily focused on creating optimal gradient-free algorithms based on three criteria: iteration complexity, oracle complexity, and maximum level of adversary noise. For black-box problems, a gradient approximation is usually used instead of an exact gradient in first-order algorithms. For instance, work [47] investigated gradient approximation via coordinate-wise randomization, and work [15] investigated gradient approximation via random search randomization. Also, for instance, in [4] the gradient approximation via a "kernel-based" approximation is studied, the feature of which is to take into account the advantages for the case of increased smoothness. Some works use smoothing schemes via l_1 or l_2 randomization. For instance, paper [2] studied l_1 randomization as an alternative to the exact gradient for solving smooth optimization problems. Another paper [22] explained the advantages of solving non-smooth problems using a smoothing scheme with l_2 randomization. And in [36] the smoothing scheme through l_1 randomization for non-smooth optimization problems is investigated and it is shown that in practice there are no significant advantages of l_1 randomization over l_2 randomization. In this paper, we use a smoothing scheme with l_2 randomization to create a gradient-free algorithm for solving a non-smooth convex stochastic optimization problem.

3 Setup

We study a non-smooth convex stochastic optimization problem with constraints

$$f^* := \min_{x \in Q} [f(x) := \mathbb{E}_\xi [f(x, \xi)]] \quad (1)$$

where $Q \subseteq \mathbb{R}^d$ is a convex compact set and $f : Q \rightarrow \mathbb{R}$ is a convex function. This problem is also known as the black box problem, where a zero-order (gradient-free) oracle returns a function value $f(x, \xi)$ at the requested point x , possibly with some adversarial noise $\delta(x)$. We now formally introduce the definition of a gradient-free oracle.

Definition 1 (Gradient-free oracle). *Let gradient-free oracle returns a noise value of $f(x, \xi)$, i.e. for all $x \in Q$*

$$f_\delta(x, \xi) := f(x, \xi) + \delta(x).$$

Next, we consider the assumptions we use in our theoretical results.

3.1 Assumptions

We assume that the function is Lipschitz continuous and is convex on set Q_γ .

Assumption 1 (Lipschitz continuity of the function). *Function $f(x, \xi)$ is Lipschitz continuous with constant M , i.e. for all $x, y \in Q$:*

$$|f(y, \xi) - f(x, \xi)| \leq M(\xi) \|y - x\|_p.$$

Moreover, there exists a positive constant M such that $\mathbb{E} [M^2(\xi)] \leq M^2$.

Assumption 2 (Convexity on the set Q_γ). *Let $\gamma > 0$ a small number to be defined later and $Q_\gamma := Q + B_2^d(\gamma)$, then the function f is convex on the set Q_γ .*

We also assume that adversarial noise is bounded.

Assumption 3 (Boundedness of noise). *For all $x \in Q$, it holds $|\delta(x)| \leq \Delta$.*

Our Assumption 1 of a Lipschitz continuity of the function is similar as in [22] and generalizes to a stochastic setting. For the special case when $p = 2$ we use the notation M_2 for the Lipschitz constant (see e.g. [14]). Assumption 2 is quite standard in the literature (see e.g. [48, 41]). We used l_2 -ball here since we use l_2 randomization in this paper. In more general the formulation of the assumption depends on the choice of gradient approximation (see e.g. [36]). So much prior work in the context of stochastic optimization often assumed the boundedness of stochastic or deterministic noise (such as e.g. [1, 44, 46]). In Assumption 3, we consider bounded deterministic noise.

3.2 Notation

We use $\langle x, y \rangle := \sum_{i=1}^d x_i y_i$ to denote standard inner product of $x, y \in \mathbb{R}^d$, where x_i and y_i are the i -th component of x and y respectively. We denote l_p -norms (for $p \geq 1$) in \mathbb{R}^d as $\|x\|_p := \left(\sum_{i=1}^d |x_i|^p \right)^{1/p}$. Particularly for l_2 -norm in \mathbb{R}^d it follows $\|x\|_2 := \sqrt{\langle x, x \rangle}$. We denote l_p -ball as $B_p^d(r) := \{x \in \mathbb{R}^d : \|x\|_p \leq r\}$ and l_p -sphere as $S_p^d(r) := \{x \in \mathbb{R}^d : \|x\|_p = r\}$. Operator $\mathbb{E}[\cdot]$ denotes full mathematical expectation. We notation $\tilde{O}(\cdot)$ to hide logarithmic factors. To define the diameter of the set Q we introduce $D := \max_{x, y \in Q} \|x - y\|_p$.

4 Main Result

In this section, we present a novel algorithm (see Algorithm 1) that is optimal in terms of iterative complexity, the number of gradient-free oracle calls, and the maximum value of adversarial noise. This algorithm is based on an accelerated first-order Stochastic Conditional Gradient Sliding (SCGD) method from [33]. This section is structured as follows: in Subsection 4.1 we introduce the basic idea of the smoothing scheme, in Subsections 4.2 and 4.3 we consider the main elements of the smoothing scheme via l_2 randomization, and in Subsection 4.4 we present the new gradient-free method (see Algorithm 1 for more details).

We start with the main idea of solving problem (1) via the smoothing scheme.

4.1 Smoothing scheme intuition

The main idea of solving problem (1) via the smoothing scheme is to replace the problem. That is, instead of solving the non-smooth problem we will solve its smoothed problem:

$$\min_{x \in Q} f_\gamma(x), \quad (2)$$

where f_γ a smooth approximation of the non-smooth function f , which we define below. Thus, to solve the smooth problem (2) it is sufficient to choose the accelerated batched algorithm $\mathbf{A}(L_{f_\gamma}, \sigma^2)$. Next, we introduce the assumptions of smoothness of the function f_γ and bounded variance of the gradient $\nabla f_\gamma(x, \psi)$.

Assumption 4 (L_{f_γ} -smoothness). *Function $f_\gamma(x)$ is differentiable and there exists a constant $L_{f_\gamma} \geq 0$ such that for $x, y \in Q$:*

$$\|\nabla f_\gamma(y) - \nabla f_\gamma(x)\|_q \leq L_{f_\gamma} \|y - x\|_p.$$

Assumption 5 (Bounded variance and unbiased). *Gradient $\nabla f_\gamma(x, \xi)$ has bounded variance such that for $x \in Q$:*

$$\mathbb{E}_\psi [\|\nabla_x f_\gamma(x, \psi) - \nabla f_\gamma(x)\|_q^2] \leq \sigma^2, \quad \mathbb{E}_\psi [\nabla f_\gamma(x, \psi)] = \nabla f_\gamma(x).$$

Assumptions 4 and 5 are quite common in the literature (see e.g. [33, 44, 26]). Here q is such that $1/p + 1/q = 1$. And a random variable ψ we define below.

The connection between Problems (1) and (2) is as follows: to solve a non-smooth problem with ε -accuracy, it is necessary to solve a smooth problem with $(\varepsilon/2)$ -accuracy, where ε -suboptimality is the accuracy of the solution in terms of expectation (see Appendix A for the proof of this statement). So, to solve Problem (1) (under the Assumption 4 and 5) with Algorithm $\mathbf{A}(L_{f_\gamma}, \sigma^2)$, we need to know the gradient of the smoothed function $\nabla f_\gamma(x, \psi)$, L_{f_γ} -smoothness constant, and the variance estimate σ^2 .

In the following subsections we will define these elements.

4.2 Smooth approximation

Since problem (1) is non-smooth, we introduce a smooth approximation of the non-smooth function f as follows:

$$f_\gamma(x) := \mathbb{E}_{\tilde{e}} [f(x + \gamma\tilde{e})], \quad (3)$$

where $\gamma > 0$ is smoothing parameter, \tilde{e} is random vector uniformly distributed on $B_2^d(\gamma)$. Here $f_\gamma(x) := \mathbb{E}[f(x, \xi)]$. The following lemma provides the connection between non-smooth function f and smoothed function f_γ .

Lemma 1. *Let Assumptions 1, 2 it holds, then for all $x \in Q$ we have*

$$f(x) \leq f_\gamma(x) \leq f(x) + \gamma M_2.$$

Proof. For the first inequality we use the convexity of the function $f(x)$

$$f_\gamma(x) = \mathbb{E}_{\tilde{e}} [f(x + \gamma\tilde{e})] \geq \mathbb{E}_{\tilde{e}} [f(x) + \langle \nabla f(x), \gamma\tilde{e} \rangle] = \mathbb{E}_{\tilde{e}} [f(x)] = f(x).$$

For the second inequality we have

$$\begin{aligned} |f_\gamma(x) - f(x)| &= |\mathbb{E}_{\tilde{e}} [f(x + \gamma\tilde{e})] - f(x)| \leq \mathbb{E}_{\tilde{e}} [|f(x + \gamma\tilde{e}) - f(x)|] \\ &\leq \gamma M_2 \mathbb{E}_{\tilde{e}} [\|\tilde{e}\|_2] \leq \gamma M_2, \end{aligned}$$

using the fact that f is M_2 -Lipschitz function. \square

The next lemmas provide properties of the smoothed function f_γ .

Lemma 2. *Let Assumptions 1, 2 it holds, then for $f_\gamma(x)$ from (3) we have*

$$|f_\gamma(y) - f_\gamma(x)| \leq M \|y - x\|_p, \quad \forall x, y \in Q.$$

Proof. Using M -Lipschitz continuity of function f we obtain

$$|f_\gamma(y) - f_\gamma(x)| \leq \mathbb{E}_{\tilde{e}} [|f(y + \gamma\tilde{e}) - f(x + \gamma\tilde{e})|] \leq M \|y - x\|_p. \quad \square$$

Lemma 3 (Theorem 1, [22]). *Let Assumptions 1, 2 it holds, then $f_\gamma(x)$ has $L_{f_\gamma} = \frac{\sqrt{d}M}{\gamma}$ -Lipschitz gradient*

$$\|\nabla f_\gamma(y) - \nabla f_\gamma(x)\|_q \leq L_{f_\gamma} \|y - x\|_p, \quad \forall x, y \in Q.$$

4.3 Gradient via l_2 randomization

The gradient of $f_\gamma(x, \xi)$ can be estimated by the following approximation:

$$\nabla f_\gamma(x, \xi, e) = \frac{d}{2\gamma} (f_\delta(x + \gamma e, \xi) - f_\delta(x - \gamma e, \xi)) e, \quad (4)$$

where $f_\delta(x, \xi)$ is gradient-free oracle from Definition 1, e is a random vector uniformly distributed on $S_2^d(\gamma)$. The following lemma provides properties of the gradient $\nabla f_\gamma(x, \xi, e)$.

Lemma 4 (Lemma 2, [36]). *Gradient $\nabla f_\gamma(x, \xi, e)$ has bounded variance (second moment) for all $x \in Q$*

$$\mathbb{E}_{\xi, e} [\|\nabla f_\gamma(x, \xi, e)\|_q^2] \leq \kappa(p, d) \left(dM_2^2 + \frac{d^2 \Delta^2}{\sqrt{2}\gamma^2} \right),$$

where $1/p + 1/q = 1$ and

$$\kappa(p, d) = \sqrt{2} \min \{q, \ln d\} d^{1 - \frac{2}{p}}.$$

Remark 1. Using the fact that the second moment is the upper estimate of the variance for the unbiased gradient and assuming that Δ is sufficiently small we obtain the following estimate of the variance from Lemma 4:

$$\sigma^2 \leq 2\sqrt{2} \min \{q, \ln d\} d^{2 - \frac{2}{p}} M_2^2.$$

4.4 Zero-Order Stochastic Conditional Gradient Sliding Method

Now we present gradient-free algorithm (see Algorithm 1) to solve problem (1). We chose Stochastic Conditional Gradient Sliding Method as accelerated batched Algorithm $\mathbf{A}(L_{f_\gamma}, \sigma^2)$. Substituting the approximation of the gradient via l_2 randomization $\nabla f_\gamma(x, \xi, e)$ ($\nabla f_\gamma(x, \psi)$ from Subsection 4.1, where $\psi = (\xi, e)$ is not only the random value ξ , but also the randomization on the l_2 -sphere e , which was introduced in Subsection 4.3) instead of the exact gradient, we obtain a new ZO-SCGD Algorithm 1 to solve the non-smooth problem (1).

Algorithm 1 Zero-Order Stochastic Conditional Gradient Sliding (ZO-SCGS)

Input: Start point $x_0 \in Q$, maximum number of iterations $N \in \mathbb{Z}_+$.

Let stepsize $\zeta_k \in [0, 1]$, learning rate $\eta_k > 0$, accuracies β_k , batch size $B_k \in \mathbb{Z}_+$, smoothing parameter $\gamma > 0$.

Initialization: Generate independently vectors e_1, e_2, \dots uniformly distributed on unit l_2 -sphere, and set $y_0 \leftarrow x_0$

- 1: **for** $k = 1, \dots, N$ **do**
- 2: $z_k \leftarrow (1 - \zeta_k)x_{k-1} + \zeta_k y_{k-1}$
- 3: Sample $\{e_1, \dots, e_{B_k}\}$ and $\{\xi_1, \dots, \xi_{B_k}\}$ independently
- 4: $g_k \leftarrow \frac{1}{B_k} \sum_{i=1}^{B_k} \left[\frac{d}{2\gamma} (f(z_k + \gamma e_i, \xi_i) - f(z_k - \gamma e_i, \xi_i)) e_i \right]$
- 5: $y_k \leftarrow \text{CG}(g_k, y_{k-1}, \eta_k, \beta_k)$ ▷ See CG in Algorithm 2
- 6: $x_k \leftarrow (1 - \zeta_k)x_{k-1} + \zeta_k y_k$
- 7: **end for**

Output: x_N .

Algorithm 1 has such parameters as number of iterations N , batch size B , stepsize ζ , learning rate η , accuracies β . The recommendations for selecting these parameters can be found in Theorem 1. To prove theorem we also need to know the values of the following parameters: constant of Lipschitz gradient $L_{f_\gamma} = \frac{2\sqrt{d}MM_2}{\varepsilon}$ (by substituting $\gamma = \varepsilon/(2M_2)$ in Lemma 3), where constant of Lipschitz continuity M defined in Lemma 2 under Assumption 1, and estimate of the variance $\sigma^2 \leq 2\sqrt{2} \min\{q, \ln d\} d^{2-\frac{2}{p}} M_2^2$ (from Remark 1).

Next theorem provides estimates of the convergence rate of Algorithm 1.

Theorem 1. *Let ε be desired accuracy to solve problem (1) and γ be chosen as $\gamma = \varepsilon/(2M_2)$. Let function $f(x, \xi)$ satisfy the Assumptions 1, 2 and 3. Then Zero-Order Stochastic Conditional Gradient Sliding algorithm (see Algorithm 1) with $\zeta_k = 3/(k+3)$, $\eta_k = 8\sqrt{d}MM_2/(\varepsilon(k+3))$, $\beta_k = 2\sqrt{d}MM_2D^2/(\varepsilon(k+1)(k+2))$, and $B_k = \left\lceil \min\{q, \ln d\} d^{1-\frac{2}{p}} (k+3)^3 \varepsilon^2 / (MD)^2 \right\rceil$ achieves $\mathbb{E}[f(x_k)] - f^* \leq \varepsilon$ after*

$$N = \mathcal{O}\left(\frac{d^{1/4}\sqrt{MM_2}D}{\varepsilon}\right), \quad T = \mathcal{O}\left(\frac{\min\{q, \ln d\} d^{2-\frac{2}{p}} M_2^2 D^2}{\varepsilon^2}\right)$$

number of iterations and gradient-free oracle calls respectively.

See Appendix A for detailed proof.

The results of Theorem 1 show that Zero-Order Stochastic Conditional Gradient Sliding algorithm converges with ε -accuracy in $N \sim d^{1/4}\varepsilon^{-1}$ iterations. The number of solutions to linear optimization problems, also known as the linear minimization oracle (LMO), is $\mathcal{O}(\sqrt{d}\varepsilon^{-2})$. Batch size $B_k \in \mathbb{Z}_+$ must be chosen integer, so in Theorem 1 $\lceil \cdot \rceil$ denotes the whole part of the next integer number. The number of oracle calls T requiring the Algorithm 1 to solve a non-smooth problem (1) with ε -accuracy is $T \sim \min\{q, \ln d\}d^{2-\frac{2}{p}}\varepsilon^{-2}$.

Remark 2 (Smooth setting). In Theorem 1, we presented the convergence results of Algorithm 1 in the non-smooth setting, since in this paper we focus on solving non-smooth convex stochastic optimization problems. However, the algorithm proposed in this paper is robust to the smooth setting as well. To obtain similar estimates of the algorithm for smooth setting, it is sufficient not to change constant of Lipschitz gradient (i.e., it is not necessary to substitute the value obtained in Lemma 3). Therefore, Algorithm 1 with parameters $\zeta_k = 3/(k+3)$, $\eta_k = 4L/(k+3)$, $\beta_k = LD^2/((k+1)(k+2))$, and $B_k = \left\lceil \min\{q, \ln d\}d^{2-\frac{2}{p}}M_2^2(k+3)^3/(LD)^2 \right\rceil$ achieves $\mathbb{E}[f(x_k)] - f^* \leq \varepsilon$ after $N \sim \varepsilon^{-1/2}$ iterations, performs $\sim \varepsilon^{-1}$ linear optimization and requires $T \sim \min\{q, \ln d\}d^{2-\frac{2}{p}}\varepsilon^{-2}$ gradient-free oracle calls.

Remark 3. In Subsection 4.4, we focus on obtaining optimal estimates of iterative N and oracle T complexities, so in proving the Theorem 1 we considered the case $\Delta = 0$. However, an optimal estimate of the maximum adversarial noise can be obtained by performing a similar convergence analysis of the Stochastic Conditional Gradient Sliding Method for the biased stochastic oracle (see example analysis in [25]). For brevity, we omit this analysis, stating that the estimate of maximum adversarial noise is $\Delta \lesssim \varepsilon^2 d^{-1/2}$ for gradient-free algorithms created by applying smoothing scheme via l_2 randomization (see e.g. [14, 22]).

5 Discussion

As far as we know, Zero-Order Stochastic Conditional Gradient Sliding (ZO-SCGS) is the first gradient-free conditional gradient-type algorithm that solves a non-smooth convex stochastic optimization problem (1). This algorithm, as Theorem 1 shows, is robust for solving non-smooth black-box problems. But most interestingly, this algorithm is also robust for smooth black box problems, because it is superior in terms of the number of oracle calls to the state of the art algorithms. For instance, the Zeroth-Order Stochastic Conditional Gradient Method (ZSCG) from [5], which is a SOTA algorithm, has the following oracle complexity of $T \sim d \varepsilon^{-3}$ in any setting, while Algorithm 1 has oracle complexity of $T \sim d \varepsilon^{-2}$ in the Euclidean setting $p = 2$ ($q = 2$) and $T \sim \ln(d) \varepsilon^{-2}$ in the simplex setting $p = 1$ ($q = \infty$). One reason for the advantage of our algorithm may be that the ZSCG method uses Direct Finite Difference (FFD), while the ZO-SCGS method (see Algorithm 1) uses Central Finite Difference (CFD). It

is worth noting that [45] explains why it is worth estimating the gradient via central finite difference. Another possible reason may be the choice of Gaussian smoothing instead of smoothing via l_2 randomization, because in practical examples it often happens that the algorithm whose gradient is approximated over l_2 randomization works better than the algorithm whose gradient is approximated over Gaussian smoothing. Last but not least, a possible reason is that the Zeroth-Order Stochastic Conditional Gradient Method (ZSCG) used the unaccelerated first-order Stochastic Frank–Wolfe (SFW) method of [28] as its base. Since the Stochastic Conditional Gradient Method already has an estimate on the number of calls to the stochastic gradient as $\sim \varepsilon^{-3}$. It is hard to expect an improvement in estimate of oracle complexity when creating a gradient-free method based on it. Therefore, in this paper we created an optimal gradient-free method based on an accelerated batched first-order algorithm. So far we have observed theoretical advantages of Algorithm 1 (robust for solving non-smooth black box problems) in terms of oracle complexity over SOTA algorithms, which are robust for solving smooth black box problems. Therefore, in Section 6 we will verify our theoretical results with a model example of a convex stochastic optimization problem in a smooth setting.

6 Experiments

In this section we focus on verifying our theoretical results obtained in Section 4 via experiments*. In particular, we numerically compare the Zero-Order Stochastic Conditional Gradient Sliding Method (ZO-SCGS) proposed in this paper (see Algorithm 1) with the Zeroth-Order Stochastic Conditional Gradient Method (ZSCG) from [5]. We consider a standard model example of a black box problem in a smooth setting, which has the following form:

$$\min_{x \in Q} f(x) := \frac{1}{2} \langle x, Ax \rangle - \langle b, x \rangle,$$

where $Q = \{x \in \mathbb{R}^d : \|x\|_1 = 1, x \geq 0\}$ is a simplex set, $A \in \mathbb{R}^{d \times d}$ is a random positively determined matrix, $b \in \mathbb{R}^d$ is a vector such that $b = Ax_*$, and x_* is a solution to the problem $x_* = \arg \min_{x \in Q} f(x)$. In all tests, the dimensionality of the problem is $d = 100$, we fix the maximum number of calls to the gradient-free oracle $T_{max} = 10^7$, and the parameters of the algorithms are taken according to theoretical recommendations: for instance, parameters for Algorithm 1, see Remark 2, and parameters for Zeroth-Order Stochastic Conditional Gradient Method, see [5]. In Figure 1 we compare the ZO-SCGS method with the ZSCG method. In particular, Figure 1a shows the dependence of the optimal error ($f(x_k) - f^*$) on the number of calls of the gradient-free oracle T . And Figure 1b examines the dependence of the optimal error ($f(x_k) - f^*$) on the number of iterations N . We observe that Algorithm 1 significantly outperforms the ZSCG method in the number of oracle calls. Also, when the maximal value of

* Code repository link: <https://github.com/htower/zo-scgs>

the gradient-free oracle call is fixed, we see that the Algorithm 1 is first inferior to the ZSCG method in the number of iterations.

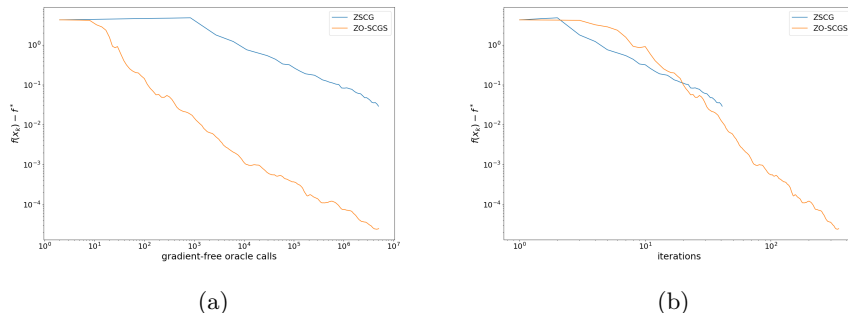


Fig. 1: Comparison of convergence result of Algorithm 1 with ZSCG method [5].

According to theoretical estimates for the ZO-SCGS and ZSCG methods, the batch size should be taken at a large size, which is a disadvantage of these algorithms. In Figure 2 we compare Algorithm 1 with ZSCG methods using the fixed batch-size $B_k = 100$. Figure 2a shows the dependence of the optimal error ($f(x_k) - f^*$) on the number of calls of the gradient-free oracle T . And Figure 2b examines the dependence of the optimal error ($f(x_k) - f^*$) on the number of iterations N . We see that for a fixed (small) batch size, both algorithms have convergence, which is a positive result for practical experiments to use. We also see that ZO-SCGS and ZSCG methods require the same number of calls to the gradient-free oracle, since we have fixed the batch size in contrast to Figure 1. We can also observe that Algorithm 1 significantly outperforms the method both in the number of to gradient-free oracle calls and in iterations.

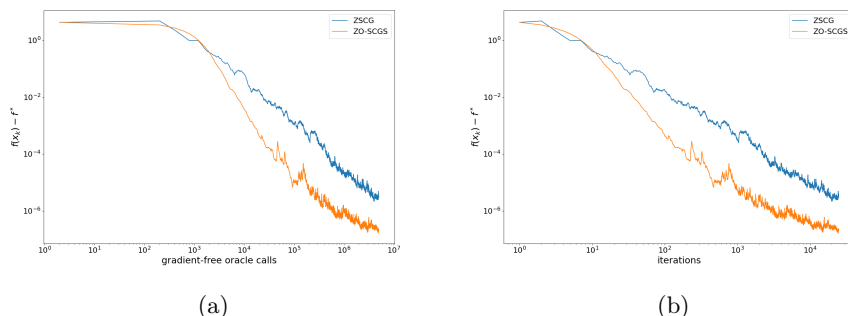


Fig. 2: Comparison of convergence result of algorithms with fixed batch size.

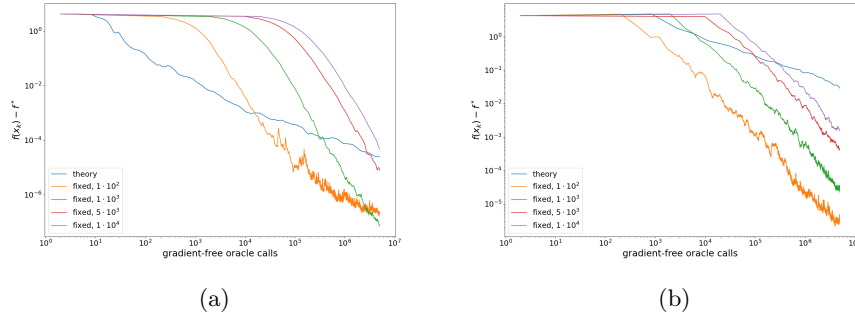


Fig. 3: Effect of the batch size parameter B_k on convergence results.

Figure 3 shows the effect of the batch size parameter B_k on convergence. Where ‘theory’ means that the batch size corresponds to theoretical estimates, ‘fixed b ’ means that the batch size corresponds to the value of b . Figure 3a explores the dependence of Zero-Order Stochastic Conditional Gradient Sliding (ZO-SCGS) on batch size B_k , and Figure 3b explores the dependence of the Zeroth-Order Stochastic Conditional Gradient Method (ZSCG) on batch size B_k . We see that theoretical estimates of the batch size slow down the convergence rate of both methods. And we can also observe a tendency that the smaller the batch size, the faster the algorithms converge. However, it is worth observing the golden mean, because with a very small batch size the positive convergence effect will not be observed, as well as with a very large batch size.

7 Conclusion

We presented, as far as we know, the first gradient-free algorithm of the conditional gradient type, which is robust for solving non-smooth convex stochastic optimization problems (black-box problems in a non-smooth setting). Using a smoothing scheme with l_2 randomization and basing on an accelerated batched first-order algorithm, we showed that Zero-Order Stochastic Conditional Gradient Sliding (ZO-SCGS) is the optimal algorithm for three criteria: total number of iterations, oracle complexity, and maximum adversarial noise. Our theoretical results show that Algorithm 1 is a robust method not only for non-smooth black box problems, but also for black box problems with a smooth setting. We verified our theoretical results on a practical experiment in a smooth setup by comparing our algorithm with the state of the art algorithm. We have shown that using a fixed (small enough) batch size achieves better accuracy than with batch size derived from theoretical estimates.

References

1. Akhavan, A., Pontil, M., Tsybakov, A. (2020). Exploiting higher order smoothness in derivative-free optimization and continuous bandits. *Advances in Neural Information Processing Systems*, 33, 9017-9027.
2. Akhavan, A., Chzhen, E., Pontil, M., Tsybakov, A. B. (2022). A gradient estimator via L1-randomization for online zero-order optimization with two point feedback. arXiv preprint arXiv:2205.13910.
3. Anikin A. et al. (2022) Efficient numerical methods to solve sparse linear equations with application to pagerank. *Optimization Methods and Software*, V. 37, no. 3, P. 907-935. <https://doi.org/10.1080/10556788.2020.1858297>
4. Bach, F., Perchet, V. (2016, June). Highly-smooth zero-th order online optimization. In *Conference on Learning Theory* (pp. 257-283). PMLR.
5. Balasubramanian, K., Ghadimi, S. (2022). Zeroth-order nonconvex stochastic optimization: Handling constraints, high dimensionality, and saddle points. *Foundations of Computational Mathematics*, 1-42.
6. Braun, G., Carderera, A., Combettes, C. W., Hassani, H., Karbasi, A., Mokhtari, A., Pokutta, S. (2022). Conditional gradient methods. arXiv preprint arXiv:2211.14103.
7. Bubeck S. (2015) Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, V. 8, no. 3-4, P. 231-357. <https://doi.org/10.1561/9781601988614>
8. Bubeck, S., Jiang, Q., Lee, Y. T., Li, Y., Sidford, A. (2019). Complexity of highly parallel non-smooth convex optimization. *Advances in neural information processing systems*, 32.
9. Chen, L., Zhang, M., Karbasi, A. (2019, April). Projection-free bandit convex optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics* (pp. 2047-2056). PMLR.
10. Combettes, C. W., Spiegel, C., Pokutta, S. (2020). Projection-free adaptive gradients for large-scale optimization. arXiv preprint arXiv:2009.14114.
11. Cox B., Juditsky A., Nemirovski A. (2017) Decomposition techniques for bilinear saddle point problems and variational inequalities with affine monotone operators. *Journal of Optimization Theory and Applications*, V. 172, P. 402-435. <https://doi.org/10.1007/s10957-016-0949-3>
12. Dadras, A., Prakhya, K., Yurtsever, A. Federated Frank-Wolfe Algorithm. In *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*.
13. Duchi, J., Hazan, E., Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).
14. Dvinskikh, D., Tominin, V., Tominin, I., Gasnikov, A. (2022, June). Noisy Zeroth-Order Optimization for Non-smooth Saddle Point Problems. In *Mathematical Optimization Theory and Operations Research: 21st International Conference, MOTOR 2022, Petrozavodsk, Russia, July 2–6, 2022, Proceedings* (pp. 18-33). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-09607-5_2
15. Dvurechensky, P., Gorbunov, E., Gasnikov, A. (2021). An accelerated directional derivative method for smooth stochastic convex optimization. *European Journal of Operational Research*, 290(2), 601-621. <https://doi.org/10.1016/j.ejor.2020.08.027>
16. Frank, M., Wolfe, P. (1956). An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2), 95-110. <https://doi.org/10.1002/nav.3800030109>

17. Freund, R. M., Grigas, P., Mazumder, R. (2017). An extended Frank–Wolfe method with “in-face” directions, and its application to low-rank matrix completion. *SIAM Journal on optimization*, 27(1), 319-346. <https://doi.org/10.1137/15m104726x>
18. Garber, D., Kretzu, B. (2020, June). Improved regret bounds for projection-free bandit convex optimization. In *International Conference on Artificial Intelligence and Statistics* (pp. 2196-2206). PMLR.
19. Garber, D., Wolf, N. (2021, July). Frank-Wolfe with a nearest extreme point oracle. In *Conference on Learning Theory* (pp. 2103-2132). PMLR.
20. Garber, D. (2022). Linear convergence of Frank–Wolfe for rank-one matrix recovery without strong convexity. *Mathematical Programming*, 1-35. <https://doi.org/10.1007/s10107-022-01821-8>
21. Gasnikov, A., Gasnikova, E. (2020). Traffic assignment models. Numerical aspects. arXiv preprint arXiv:2003.12160.
22. Gasnikov, A., Novitskii, A., Novitskii, V., Abdukhakimov, F., Kamzolov, D., Beznosikov, A., ..., Gu, B. (2022). The power of first-order smooth optimization for black-box non-smooth problems. arXiv preprint arXiv:2201.12289.
23. Gasnikov, A., Dvinskikh, D., Dvurechensky, P., Gorbunov, E., Beznosikov, A., Lobanov, A. (2022). Randomized gradient-free methods in convex optimization. arXiv preprint arXiv:2211.13566.
24. Goldfarb, D., Iyengar, G., Zhou, C. (2017, April). Linear convergence of stochastic frank wolfe variants. In *Artificial Intelligence and Statistics* (pp. 1066-1074). PMLR.
25. Gorbunov, E., Dvinskikh, D., Gasnikov, A. (2019). Optimal decentralized distributed algorithms for stochastic convex optimization. arXiv preprint arXiv:1911.07363.
26. Gorbunov, E., Danilova, M., Gasnikov, A. (2020). Stochastic optimization with heavy-tailed noise via accelerated gradient clipping. *Advances in Neural Information Processing Systems*, 33, 15042-15053.
27. Guélat, J., Marcotte, P. (1986). Some comments on Wolfe’s ‘away step’. *Mathematical Programming*, 35(1), 110-119. <https://doi.org/10.1007/bf01589445>
28. Hazan, E. (2016). Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4), 157-325. <https://doi.org/10.1561/24000000013>
29. Hazan, E., Luo, H. (2016, June). Variance-reduced and projection-free stochastic optimization. In *International Conference on Machine Learning* (pp. 1263-1271). PMLR.
30. Jaggi, M. (2011). Sparse convex optimization methods for machine learning (No. THESIS LIB). ETH Zürich.
31. Kiefer, J., Wolfowitz, J. (1952). Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 462-466. <https://doi.org/10.1214/aoms/1177729392>
32. Lacoste-Julien, S., Jaggi, M. (2015). On the global linear convergence of Frank-Wolfe optimization variants. *Advances in neural information processing systems*, 28.
33. Lan, G., Zhou, Y. (2016). Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2), 1379-1409. <https://doi.org/10.1137/140992382>
34. Levitin, E. S., Polyak, B. T. (1966). Constrained minimization methods. *USSR Computational mathematics and mathematical physics*, 6(5), 1-50.
35. LeBlanc, L. J., Helgason, R. V., Boyce, D. E. (1985). Improved efficiency of the Frank-Wolfe algorithm for convex network programs. *Transportation Science*, 19(4), 445-462. <https://doi.org/10.1287/trsc.19.4.445>

36. Lobanov, A., Alashqar, B., Dvinskikh, D., Gasnikov, A. (2022). Gradient-Free Federated Learning Methods with l_1 and l_2 -Randomization for Non-Smooth Convex Stochastic Optimization Problems. arXiv preprint arXiv:2211.10783.
37. McMahan, H. B., Streeter, M. (2010). Adaptive bound optimization for online convex optimization. arXiv preprint arXiv:1002.4908.
38. Mitradjieva, M., Lindberg, P. O. (2013). The stiff is moving—Conjugate direction Frank-Wolfe methods with applications to traffic assignment. *Transportation Science*, 47(2), 280-293. <https://doi.org/10.1287/trsc.1120.0409>
39. Mokhtari, A., Hassani, H., Karbasi, A. (2020). Stochastic conditional gradient methods: From convex minimization to submodular maximization. *The Journal of Machine Learning Research*, 21(1), 4232-4280.
40. Négiar, G., Dresdner, G., Tsai, A., El Ghaoui, L., Locatello, F., Freund, R., Pedregosa, F. (2020, November). Stochastic Frank-Wolfe for constrained finite-sum minimization. In *International Conference on Machine Learning* (pp. 7253-7262). PMLR.
41. Risteski, A., Li, Y. (2016). Algorithms and matching lower bounds for approximately-convex optimization. *Advances in Neural Information Processing Systems*, 29.
42. Rosenbrock, H. (1960). An automatic method for finding the greatest or least value of a function. *The computer journal*, 3(3), 175-184. <https://doi.org/10.1093/comjnl/3.3.175>
43. Saha, A., Tewari, A. (2011, June). Improved regret guarantees for online smooth convex optimization with bandit feedback. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 636-642). JMLR Workshop and Conference Proceedings.
44. Stich, S. U., Karimireddy, S. P. (2020). The error-feedback framework: Better rates for sgd with delayed gradients and compressed updates. *The Journal of Machine Learning Research*, 21(1), 9613-9648.
45. Scheinberg, K. (2022). Finite Difference Gradient Approximation: To Randomize or Not?. *INFORMS Journal on Computing*, 34(5), 2384-2388. <https://doi.org/10.1287/ijoc.2022.1218>
46. Vasin, A., Gasnikov, A., Spokoiny, V. (2021). Stopping rules for accelerated gradient methods with additive noise in gradient (Vol. 2812, No. 2812). Berlin: Weierstraß-Institut für Angewandte Analysis und Stochastik.
47. Vaswani, S., Bach, F., Schmidt, M. (2019, April). Fast and faster convergence of sgd for over-parameterized models and an accelerated perceptron. In *The 22nd international conference on artificial intelligence and statistics* (pp. 1195-1204). PMLR.
48. Yousefian, F., Nedić, A., Shanbhag, U. V. (2012). On stochastic gradient and sub-gradient methods with adaptive steplength sequences. *Automatica*, 48(1), 56-67. <https://doi.org/10.1016/j.automatica.2011.09.043>
49. Yurtsever, A., Sra, S., Cevher, V. (2019, May). Conditional gradient methods via stochastic path-integrated differential estimator. In *International Conference on Machine Learning* (pp. 7282-7291). PMLR.
50. Zhang, M., Shen, Z., Mokhtari, A., Hassani, H., Karbasi, A. (2020, June). One sample stochastic frank-wolfe. In *International Conference on Artificial Intelligence and Statistics* (pp. 4012-4023). PMLR.

A Proof Theorem 1

Before giving the proof of Theorem 1, we prove the following statement from the last part of Subsection 4.1: to solve a non-smooth problem with ε -accuracy, it is necessary to solve a smooth problem with $(\varepsilon/2)$ -accuracy, where ε -suboptimality is the accuracy of the solution in terms of expectation.

Connection between Problems (1) and (2). Using Lemma 1 we obtain

$$\begin{aligned} f(x^{N+1}) - f(x_*) &\stackrel{\textcircled{1}}{\leq} f_\gamma(x^{N+1}) - f(x_*) \stackrel{\textcircled{2}}{\leq} f_\gamma(x^{N+1}) - f_\gamma(x_*) + \gamma M_2 \\ &\leq f_\gamma(x^{N+1}) - f_\gamma(x_*(\gamma)) + \gamma M_2 \leq \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon, \end{aligned}$$

where $\textcircled{1}$ means the first inequality of the Lemma 1 and $\textcircled{2}$ means the second inequality of the Lemma 1. Thus, if we solve a smooth problem (2) with $\varepsilon/2$ accuracy ($f_\gamma(x^{N+1}) - f_\gamma(x_*(\gamma)) \leq \varepsilon/2$), we solve a non-smooth problem (1) with ε -accuracy ($f(x^{N+1}) - f(x_*) \leq \varepsilon$) when we choose the gamma parameter $\gamma = \varepsilon/(2M_2)$.

First, let's define the procedure CG (see Algorithm 2) to which Algorithm 1 refers. The Conditional Gradient procedure has the following input parameters: g_0, u_0, η, β which correspond to parameters: $g_k, y_{k-1}, \eta_k, \beta_k$, where k is number of iteration for Algorithm 1. The CG procedure outputs the following value u .

Algorithm 2 Conditional Gradient procedure ($u \leftarrow \text{CG}(g_0, u_0, \eta, \beta)$)

```

1: for  $t = 0, \dots, T$  do
2:    $v_t \leftarrow \operatorname{argmin}_{v \in Q} \langle g_t, v \rangle$ 
3:   if  $\langle g_t, u_t - v_t \rangle \leq \beta$  then
4:     return  $u_t$ 
5:   end if
6:    $\alpha_t \leftarrow \min \left\{ \frac{\langle g_t, u_t - v_t \rangle}{\eta \|u_t - v_t\|^2}, 1 \right\}$ 
7:    $u_{t+1} \leftarrow u_t + \alpha_t (v_t - u_t)$ 
8:    $g_{t+1} \leftarrow g_0 + \eta (u_{t+1} - u_0)$ 
9: end for

```

We now obtain the convergence rate of Algorithm 1: Zero-Order Stochastic Conditional Gradient Sliding (ZO-SCGS) to solve the non-smooth problem (1). **Convergence rate for Algorithm (1).** Let us write out the convergence rate for the Stochastic Conditional Gradient Sliding algorithm from [33]:

$$\mathbb{E}[f_\gamma(x_k)] - f_\gamma(x_*(\gamma)) \leq \frac{7.5LD^2}{(k+1)(k+2)}, \quad (5)$$

where L is a constant of Lipschitz gradient. Then, in order to obtain the convergence rate for Algorithm 1, we use the Smoothing Scheme (see Subsection 4.1 for more details). Substituting $L = L_{f_\gamma} = \frac{2\sqrt{d}MM_2}{\varepsilon}$ (from Lemma 3) and

$\sigma^2 \leq 2\sqrt{2} \min\{q, \ln d\} d^{2-\frac{2}{p}} M_2^2$ (from Remark 1) in (5) we obtain the convergence rate for Zero-Order Stochastic Conditional Gradient Sliding (ZO-SCGS):

$$\mathbb{E}[f(x_k)] - f^* \leq \frac{15\sqrt{d}MM_2D^2}{\varepsilon(k+1)(k+2)}. \quad (6)$$

We turn to estimates of number of iterations and call of the gradient-free oracle. **Iterative and oracular complexities.** We first find an estimate of the number of iterations N . To do this, we assume that the Algorithm 1 achieves ε -accuracy after N iterations, then from (6) we obtain:

$$\begin{aligned} \frac{15\sqrt{d}MM_2D^2}{\varepsilon(N+1)(N+2)} \leq \varepsilon &\Rightarrow N^2 \gtrsim \frac{15\sqrt{d}MM_2D^2}{\varepsilon^2} \Rightarrow \\ &\Rightarrow N = \mathcal{O}\left(\frac{d^{1/4}\sqrt{MM_2D}}{\varepsilon}\right). \end{aligned} \quad (7)$$

Next, we find the number of calls of the gradient-free oracle (see Definition 1):

$$\begin{aligned} T = \sum_{k=1}^N B_k &= \sum_{k=1}^N \frac{\min\{q, \ln d\} d^{1-\frac{2}{p}} (k+3)^3 \varepsilon^2}{(MD)^2} \\ &\stackrel{\textcircled{3}}{\simeq} \frac{\min\{q, \ln d\} d^{1-\frac{2}{p}} N^4 \varepsilon^2}{4(MD)^2} \\ &\stackrel{(7)}{=} \mathcal{O}\left(\frac{\min\{q, \ln d\} d^{1-\frac{2}{p}} d M_2^2 M^2 D^4 \varepsilon^2}{(MD)^2 \varepsilon^4}\right) \\ &= \mathcal{O}\left(\frac{\min\{q, \ln d\} d^{2-\frac{2}{p}} M_2^2 D^2}{\varepsilon^2}\right) \\ &= \begin{cases} \mathcal{O}\left(\frac{dM_2^2D^2}{\varepsilon^2}\right), & p=2 \quad (q=2); \\ \mathcal{O}\left(\frac{(\ln d)M_2^2D^2}{\varepsilon^2}\right), & p=1 \quad (q=\infty), \end{cases} \end{aligned}$$

where in $\textcircled{3}$ we use auxiliary fact that $\sum_{k=1}^N k^3 \simeq \frac{1}{4}N^4$.