Towards Gradient Free and Projection Free Stochastic Optimization

Anit Kumar Sahu
Carnegie Mellon University

Manzil Zaheer
Google AI

Soummya Kar
Carnegie Mellon University

Abstract

This paper focuses on the problem of *constrained stochastic* optimization. A zeroth order Frank-Wolfe algorithm is proposed, which in addition to the projection-free nature of the vanilla Frank-Wolfe algorithm makes it gradient free. Under convexity and smoothness assumption, we show that the proposed algorithm converges to the optimal objective function at a rate $O\left(1/T^{1/3}\right)$, where T denotes the iteration count. In particular, the primal sub-optimality gap is shown to have a dimension dependence of $O\left(d^{1/3}\right)$, which is the best known dimension dependence among all zeroth order optimization algorithms with one directional derivative per iteration. For non-convex functions, we obtain the *Frank-Wolfe* gap to be $O\left(d^{1/3}T^{-1/4}\right)$. Experiments on black-box optimization setups demonstrate the efficacy of the proposed algorithm.

1 Introduction

In this paper, we aim to solve the following stochastic optimization problem:

$$\min_{\mathbf{x}\in\mathcal{C}} f(\mathbf{x}) = \min_{x\in\mathcal{C}} \mathbb{E}_{\mathbf{y}\sim\mathcal{P}}\left[F(\mathbf{x};\mathbf{y})\right], \qquad (1)$$

where $\mathcal{C} \in \mathbb{R}^d$ is a closed convex set. This problem of stochastic constrained optimization has been a focus of immense interest in the context of convex functions (Bubeck et al., 2015) and non-convex functions especially in the context of deep learning (Goodfellow et al., 2016). Solutions to the problem (1) can be broadly classified into two classes: algorithms which require a projection at each step, for example, projected gradient descent (Bubeck et al., 2015) and projection free methods such as the Frank-Wolfe algorithm (Jaggi, 2013). Furthermore, algorithms designed

to solve the above optimization problem access various kinds of oracles, i.e., first order oracle (gradient queries) and zeroth order oracle (function queries). In this paper, we focus on a stochastic version of projection free method, namely Frank-Wolfe algorithm, with access to a zeroth order oracle.

Derivative free optimization or zeroth order optimization is motivated by settings where the analytical form of the function is not available or when the gradient evaluation is computationally prohibitive. Developments in zeroth order optimization has been fueled by various applications ranging from problems in medical science, material science and chemistry (Gray et al., 2004; Marsden et al., 2008; Gray et al., 2004; Deming et al., 1978; Marsden et al., 2007). In the context of machine learning, zeroth order methods have been applied to attacks on deep neural networks using black box models (Chen et al., 2017), scalable policy optimization for reinforcement learning (Choromanski et al., 2018) and optimization with bandit feedback (Bubeck et al., 2012).

For the problem in (1), it is well known that the primal sub-optimality gap of first order schemes are dimension independent. However, algorithms which involve a projection operator might be expensive in practice depending on the structure of $\mathcal{C}$. Noting the potentially expensive projection operators, projection free methods such as Frank-Wolfe (Jaggi, 2013) have had a resurgence. Frank-Wolfe avoids the projection step, and only requires access to a linear minimization oracle, which can be implemented efficiently and needs to be solved to a certain degree of exactness. Stochastic versions of Frank-Wolfe have been studied in both the convex (Hazan and Kale, 2012; Hazan and Luo, 2016; Mokhtari et al., 2018; Lan and Zhou, 2016) and non-convex (Reddi et al., 2016) setting with access to stochastic first order oracles (SFO). However, convergence of stochastic Frank-Wolfe with access to only stochastic zeroth order oracle (SZO) remains unexplored.

In this paper, we study a setting of the stochastic Frank-Wolfe where a small batch-size (independent of dimension or the number of iterations) is sampled at each epoch while having access to a zeroth order oracle. Unlike, the first order oracle based stochas-

Reference	Setting	Memory	Primal Rate	Oracle
Jaggi (2013)	Det. Convex	-	$O(1/t)$	SFO
Hazan and Kale (2012)	Stoch. Convex	$O(t)$	$O(1/t^{1/2})$	SFO
Mokhtari et al. (2018)	Stoch. Convex	$O(1)$	$O(1/t^{1/3})$	SFO
Lacoste-Julien (2016)	Det. Non-convex	-	$O(1/t^{1/2})$	SFO
Reddi et al. (2016)	Stoch. Non-convex	$O(\sqrt{t})$	$O(1/t^{1/4})$	SFO
RDSA [Theorem 3.5(1)]	Stoch. Convex	1	$O(d^{1/3}/t^{1/3})$	SZO
I-RDSA [Theorem 3.5(2)]	Stoch. Convex	m	$O((d/m)^{1/3}/t^{1/3})$	SZO
KWSA [Theorem 3.5(3)]	Stoch. Convex	d	$O(1/t^{1/3})$	SZO
I-RDSA [Theorem 3.6]	Stoch. Non-convex	m	$O((d/m)^{1/3}/t^{1/4})$	SZO

Table 1: Convergence of Frank-Wolfe: Det. refers to deterministic while stoch. refers to stochastic. Memory indicates the number of samples at which the gradients needs to be tracked in the first order case. In the zeroth order case, it indicates the number of directional derivatives being evaluated at one sample. The rates correspond to the rate of decay of $\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^*)]$ in the convex setting and the Frank-Wolfe duality gap in context of the non-convex setting.

tic Frank-Wolfe, the zeroth order counterpart is only able to generate biased gradient estimates. We focus on three different zeroth order gradient approximation schemes, namely, the classical Kiefer Wolfowitz stochastic approximation (KWSA) (Kiefer and Wolfowitz, 1952), random directions stochastic approximation (RDSA) (Nesterov and Spokoiny, 2011; Duchi et al., 2015), and an improvized RDSA (I-RDSA). KWSA samples directional derivatives along the canonical basis directions at each iteration, while RDSA samples one directional derivative at each iteration, and I-RDSA samples $m < d$ directional derivatives at each iteration. Naïve usage of the biased gradient estimates in the linear minimization step, in addition to the stochasticity of the function evaluations, can lead to potentially diverging iterate sequences.

To circumvent the potential divergence issue due to non-decaying gradient noise and bias, we use a gradient averaging technique used in Yang et al. (2016); Ruszczyński (2008); Mokhtari et al. (2018) to get a surrogate gradient estimate which reduces the noise and the associated bias. The gradient averaging technique intuitively reduces the linear minimization step to that of an inexact minimization if the exact gradient was available. For each of the zeroth order optimization schemes, i.e., KWSA, RDSA, and I-RDSA, we derive primal sub-optimality bounds and Frank-Wolfe duality gap bounds and quantify the dependence in terms of the dimension and the number of epochs. We show that the primal sub-optimality gap to be of the order $O(d^{1/3}/T^{1/3})$ for RDSA, which improves to $O((d/m)^{1/3}/T^{1/3})$ for I-RDSA, and $O(1/T^{1/3})$ for KWSA at the cost of additional directional derivatives. The dimension dependence in zeroth order optimization is unavoidable due to the inherent bias-variance trade-off but nonetheless, the dependence on the number of iterations matches that of its first order counter-

part in Mokhtari et al. (2018). Recently in (Balasubramanian and Ghadimi, 2018), a zeroth order Frank Wolfe algorithm was proposed where the number of gradient directions sampled at each epoch scales linearly with both respect to the number of iterations and dimension of the problem. For the convex case, the number of gradient directions further scales as squared number of iterations. In contrast, we focus on the case where the number of gradient directions sampled at each epoch are independent of the dimension and the number of iterations. Moreover, in (Balasubramanian and Ghadimi, 2018) it is not clear how the primal and dual gap scales with respect to dimension when dimension and iteration independent gradient directions are sampled at each iteration. Furthermore, we also derive rates for non-convex functions and show the Frank-Wolfe duality gap to be $O(d^{1/3}/T^{1/4})$, where the dependence on the number of iterations matches that of its first order counterpart in Reddi et al. (2016). To complement the theoretical results, we also demonstrate the efficacy of our algorithm through empirical evaluations on datasets. In particular, we perform experiments on a dataset concerning constrained black box non-convex optimization, where generic first order methods are rendered unusable and show that our proposed algorithm converges to a first order stationary point.

1.1 Related Work

Algorithms for convex optimization with access to a SZO have been studied in Wang et al. (2018); Duchi et al. (2015); Liu et al. (2018); Sahu et al. (2018), where in Liu et al. (2018) to address constrained optimization a projection step was considered. In the context of projection free methods, Frank and Wolfe (1956) studied the Frank-Wolfe algorithm for smooth convex func-

tions with line search which was extended to encompass inexact linear minimization step in Jaggi (2013). Subsequently with additional assumptions, the rates for classical Frank-Wolfe was improved in Lacoste-Julien and Jaggi (2015); Garber and Hazan (2015). Stochastic versions of Frank-Wolfe for convex optimization with number of calls to SFO at each iteration dependent on the number of iterations with additional smoothness assumptions have been studied in Hazan and Kale (2012); Hazan and Luo (2016) so as to obtain faster rates, while Mokhtari et al. (2018) studied the version with a mini-batch size of 1. In the context of non-convex optimization, a deterministic Frank-Wolfe algorithm was studied in Lacoste-Julien (2016), while Reddi et al. (2016) addressed the stochastic version of Frank-Wolfe and further improved the rates by using variance reduction techniques. Table 1 gives a summary of the rates of various algorithms. For the sake of comparison, we do not compare our rates with those of variance reduced versions of stochastic Frank-Wolfe in Reddi et al. (2016); Hazan and Luo (2016), as our proposed algorithm does not employ variance reduction techniques which tend to incorporate multiple restarts and extra memory in order to achieve better rates. However, note that our algorithm can be extended so as to incorporate variance reduction techniques.

2 Frank-Wolfe: First to Zeroth Order

In this paper, the objective is to solve the following optimization problem:

$$\min_{\mathbf{x}\in\mathcal{C}} f(\mathbf{x}) = \min_{x\in\mathcal{C}} \mathbb{E}_{\mathbf{y}\sim\mathcal{P}}\left[F(\mathbf{x};\mathbf{y})\right], \qquad (2)$$

where $\mathcal{C}\in\mathbb{R}^d$ is a closed convex set, the loss functions and the expected loss functions, $F(\cdot;\mathbf{y})$ and $f(\cdot)$ respectively are possibly non-convex. However, in the context of the optimization problem posed in (2), we assume that we have access to a stochastic zeroth order oracle (SZO). On querying a SZO at the iterate $\mathbf{x}_t$, yields an unbiased estimate of the loss function $f(\cdot)$ in the form of $F(\mathbf{x}_t;\mathbf{y}_t)$. Before proceeding to the algorithm and the subsequent results, we revisit preliminaries concerning the Frank-Wolfe algorithm and zeroth order optimization.

2.1 Background: Frank-Wolfe Algorithm

The celebrated Frank-Wolfe algorithm is based around approximating the objective by a first-order Taylor approximation. In the case, when exact first order information is available, i.e., one has access to an incremental first order oracle (IFO), a deterministic Frank-Wolfe method involves the following steps:

$$\begin{aligned} \mathbf{v}_t &= \operatorname*{argmin}_{\mathbf{v}\in\mathcal{C}}\langle\nabla f(\mathbf{x}_t),\mathbf{v}\rangle \\ \mathbf{x}_{t+1} &= (1-\gamma_{t+1})\mathbf{x}_t + \gamma_{t+1}\mathbf{v}_t, \end{aligned} \qquad (3)$$

where $\gamma_t = \frac{2}{t+2}$. A linear minimization oracle (LMO) is queried at every epoch. Note that, the exact minimization in (3) is a linear program[1] and can be performed efficiently without much computational overload. It is worth noting that the exact minimization in (3) can be replaced by an inexact minimization of the following form, where a $\mathbf{v}\in\mathcal{C}$ is chosen to satisfy,

$$\langle\nabla f(\mathbf{x}_t),\mathbf{v}\rangle \le \operatorname*{argmin}_{\mathbf{v}\in\mathcal{C}}\langle\nabla f(\mathbf{x}_t),\mathbf{v}\rangle + \gamma_t C_1,$$

and the algorithm can be shown to retain the same convergence rate (see, for example (Jaggi, 2013)).

2.2 Background: Zeroth Order Optimization

The crux of zeroth order optimization consists of gradient approximation schemes from appropriately sampled values of the objective function. We briefly describe the few well known zeroth order gradient approximation schemes. The Kiefer-Wolfowitz stochastic approximation (KWSA, see (Kiefer and Wolfowitz, 1952)) scheme approximates the gradient by sampling the objective function along the canonical basis vectors. Formally, gradient estimate can be expressed as:

$$\mathbf{g}(\mathbf{x}_t;\mathbf{y}) = \sum_{i=1}^{d}\frac{F(\mathbf{x}_t+c_t\mathbf{e}_i;\mathbf{y}) - F(\mathbf{x}_t;\mathbf{y})}{c_t}\mathbf{e}_i, \qquad (4)$$

where c_t is a carefully chosen time-decaying sequence. KWSA requires d samples at each step to evaluate the gradient. However, in order to avoid sampling the objective function d times, random directions based gradient estimators have been proposed recently (see, for example Duchi et al. (2015); Nesterov and Spokoiny (2011)). The random directions gradient estimator (RDSA) involves estimating the directional derivative along a randomly sampled direction from an appropriate probability distribution. Formally, the random directions gradient estimator is given by,

$$\mathbf{g}(\mathbf{x}_t;\mathbf{y},\mathbf{z}_t) = \frac{F(\mathbf{x}_t+c_t\mathbf{z}_t;\mathbf{y}) - F(\mathbf{x}_t;\mathbf{y})}{c_t}\mathbf{z}_t, \qquad (5)$$

where $\mathbf{z}_t\in\mathbb{R}^d$ is a random vector sampled from a probability distribution such that $\mathbb{E}\left[\mathbf{z}_t\mathbf{z}_t^\top\right] = \mathbf{I}_d$ and c_t is a carefully chosen time-decaying sequence. With $c_t\to 0$, both the gradient estimators in (4) and (5) turn out to be unbiased estimators of the gradient $\nabla f(\mathbf{x}_t)$.

3 Zeroth Order Stochastic Frank-Wolfe: Algorithm & Analysis

In this section, we start by stating assumptions which are required for our analysis.

---

[1]Technically speaking, when $\mathcal{C}$ is given by linear constraints.

Assumption A1. In problem (2), the set $\mathcal{C}$ is bounded with finite diameter R.

Assumption A2. F is convex and Lipschitz continuous with $\sqrt{\mathbb{E}\left[\|\nabla_x F(\mathbf{x};\cdot)\|^2\right]} \leq L_1$ for all $\mathbf{x} \in \mathcal{C}$.

Assumption A3. The expected function $f(\cdot)$ is convex. Moreover, its gradient ∇f is L-Lipschitz continuous over the set $\mathcal{C}$, i.e., for all $x, y \in \mathcal{C}$

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|. \tag{6}$$

Assumption A4. The $\mathbf{z}_t$'s are drawn from a distribution μ such that $M(\mu) = \mathbb{E}\left[\|\mathbf{z}_t\|^6\right]$ is finite, and for any vector $\mathbf{g} \in \mathbb{R}^d$, there exists a function $s(d) : \mathbb{N} \mapsto \mathbb{R}_+$ such that,

$$\mathbb{E}\left[\|\langle\mathbf{g}, \mathbf{z}_t\rangle\mathbf{z}_t\|^2\right] \leq s(d)\|\mathbf{g}\|^2.$$

Assumption A5. The unbiased gradient estimates, $\nabla F(\mathbf{x}; \mathbf{y})$ of $\nabla f(\mathbf{x})$, i.e., $\mathbb{E}_{\mathbf{y}\sim\mathcal{P}}[\nabla F(\mathbf{x}; \mathbf{y})] = \nabla f(\mathbf{x})$ satisfy

$$\mathbb{E}\left[\|\nabla F(\mathbf{x}, \mathbf{y}) - \nabla f(\mathbf{x})\|^2\right] \leq \sigma^2 \tag{7}$$

We note that Assumptions A1-A3 and A5 are standard in the context of stochastic optimization. Assumption A4 provides for the requisite moment conditions for the sampling distribution of the directions utilized for finding directional derivatives so as to be able to derive concentration bounds. In particular, if μ is taken to be uniform on the surface of the $\mathbb{R}^d$ Euclidean ball with radius $\sqrt{d}$, then we have that $M(\mu) = d^3$ and $s(d) = d$. Moreover, if μ is taken to be $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$, then $M(\mu) = d(d+2)(d+4) \approx d^3$ and $s(d) = d$. For the rest of the paper, we take μ to be either uniform on the surface of the $\mathbb{R}^d$ Euclidean ball with radius $\sqrt{d}$ or $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. Before getting into the stochastic case, we demonstrate how a typical zeroth order Frank-Wolfe framework corresponds to an inexact classical Frank-Wolfe optimization in the deterministic setting.

3.1 Deterministic Zeroth Order Frank-Wolfe

The deterministic version of the optimization in (2) can be re-stated as follows:

$$\min_{\mathbf{x}\in\mathcal{C}} F(\mathbf{x}). \tag{8}$$

In order to elucidate the equivalence of a typical zeroth order Frank-Wolfe framework corresponds to an inexact classical Frank-Wolfe optimization, we restrict our attention to the Kiefer-Wolfowitz stochastic approximation (KWSA) for gradient estimation. In particular, the KWSA gradient estimator in (4) can be expressed as follows:

$$\mathbf{g}(\mathbf{x}_t) = \sum_{i=1}^{d} \frac{F(\mathbf{x}_t + c_t\mathbf{e}_i) - F(\mathbf{x}_t)}{c_t}\mathbf{e}_i$$

Algorithm 1 Deterministic Zeroth Order Frank Wolfe

Require: Input, Loss Function $F(x)$, L (Lipschitz constant for the gradients), Convex Set $\mathcal{C}$, Sequences $\gamma_t = \frac{2}{t+1}, c_t = \frac{L\gamma_t}{d}$.

Output: : $\mathbf{x}_T$ or $\frac{1}{T}\sum_{t=1}^{T}\mathbf{x}_t$.

1: Initialize $\mathbf{x}_0 \in \mathcal{C}$
2: **for** $t = 0, 1, \ldots, T - 1$ **do**
3:     Compute $\mathbf{g}(\mathbf{x}_t) = \sum_{i=1}^{d} \frac{F(\mathbf{x}_t + c_t\mathbf{e}_i) - F(\mathbf{x}_t)}{c_t}\mathbf{e}_i$,
4:     Compute $\mathbf{v}_t = \mathrm{argmin}_{\mathbf{s}\in\mathcal{C}}\langle\mathbf{s}, \mathbf{g}(\mathbf{x}_t)\rangle$,
5:     Compute $\mathbf{x}_{t+1} = (1 - \gamma_t)\mathbf{x}_t + \gamma_t\mathbf{v}_t$.
6: **end for**

$$= \nabla F(\mathbf{x}_t) + \sum_{i=1}^{d} \frac{c_t}{2}\langle\mathbf{e}_i, \nabla^2 F(\mathbf{x}_t + \lambda_t c_t\mathbf{e}_i)\mathbf{e}_i\rangle\mathbf{e}_i, \quad (9)$$

where $\lambda \in [0, 1]$. The linear optimization step with the current gradient approximation reduces to:

$$\langle\mathbf{v}, \mathbf{g}(\mathbf{x}_t)\rangle = \langle\mathbf{v}, \nabla F(\mathbf{x}_t)\rangle$$

$$+ \frac{c_t}{2}\sum_{i=1}^{d}\langle\mathbf{e}_i, \nabla^2 F(\mathbf{x}_t + \lambda_t c_t\mathbf{e}_i)\mathbf{e}_i\rangle\langle\mathbf{v}, \mathbf{e}_i\rangle$$

$$\Rightarrow \min_{\mathbf{v}\in\mathcal{C}}\langle\mathbf{v}, \mathbf{g}(\mathbf{x}_t)\rangle \leq \min_{\mathbf{s}\in\mathcal{C}}\langle\mathbf{s}, \nabla F(\mathbf{x}_t)\rangle + \frac{c_t LRd}{2}. \quad (10)$$

In particular, if c_t is chosen to be $c_t = \frac{\gamma_t}{d}$ and $\gamma_t = \frac{2}{t+1}$, we obtain the following bound characterizing the primal gap:

Theorem 3.1. *Given the zeroth order Frank-Wolfe algorithm in Algorithm 1, we obtain the following bound:*

$$F(\mathbf{x}_t) - F(\mathbf{x}^*) = \frac{Q_{ns}}{t+2}, \tag{11}$$

where $Q_{ns} = \max\{2(F(\mathbf{x}_0) - F(\mathbf{x}^*)), 4LR^2\}$.

Theorem 3.1 asserts that with appropriate scaling of c_t, i.e., the smoothing parameter for the zeroth order gradient estimator, the iteration dependence of the primal gap matches that of the classical Frank-Wolfe scheme. In particular, for a primal gap of ϵ, the number of iterations needed for the zeroth order scheme in algorithm 1 is $O\left(\frac{1}{\epsilon}\right)$, while the number of calls to the linear minimization oracle and zeroth order oracle are given by $O\left(\frac{1}{\epsilon}\right)$ and $O\left(\frac{d}{\epsilon}\right)$ respectively. The proof of Theorem 3.1 is provided in the appendix A.

In summary, Theorem 3.1 shows that the deterministic zeroth order Frank-Wolfe algorithm reduces to the inexact classical Frank-Wolfe algorithm with the corresponding primal being dimension independent. However, the dimension independence comes at the cost of querying the zeroth order oracle d times at each iteration. In the sequel, we will focus on the random directions gradient estimator in (5) for the stochastic zeroth order Frank-Wolfe algorithm.

3.2 Zeroth Order Stochastic Frank-Wolfe

In this section, we formally introduce our proposed zeroth order stochastic Frank-Wolfe algorithm. A naive replacement of $\nabla f(\mathbf{x}_k)$ by its stochastic counterpart, i.e., $\nabla F(\mathbf{x}_k; \mathbf{y}_k)$ would make the algorithm potentially divergent due to non-vanishing variance of gradient approximations. Moreover, the naive replacement would lead to the linear minimization constraint to hold only in expectation and thereby potentially also making the algorithm divergent. We use a well known averaging trick to counter this problem which is as follows:

$$\mathbf{d}_t = (1 - \rho_t) \mathbf{d}_{t-1} + \rho_t \mathbf{g} (\mathbf{x}_t, \mathbf{y}_t), \qquad (12)$$

where $g(\mathbf{x}_t, \mathbf{y}_t)$ is a gradient approximation, $\mathbf{d}_0 = \mathbf{0}$ and ρ_t is a time-decaying sequence. Technically speaking, such a scheme allows for $\mathbb{E}\left[\|\mathbf{d}_t - \nabla f(\mathbf{x}_t)\|^2\right]$ to go to zero asymptotically. With the above averaging scheme, we replace the linear minimization and the subsequent steps as follows:

$$\mathbf{d}_t = (1 - \rho_t) \mathbf{d}_{t-1} + \rho_t \mathbf{g} (\mathbf{x}_t, \mathbf{y}_t)$$
$$\mathbf{v}_t = \operatorname*{argmin}_{\mathbf{v} \in \mathcal{C}} \langle \mathbf{d}_t, \mathbf{v} \rangle$$
$$\mathbf{x}_{t+1} = (1 - \gamma_{t+1}) \mathbf{x}_t + \gamma_{t+1} \mathbf{v}_t. \qquad (13)$$

We resort to three different gradient approximation schemes for approximating $g(\mathbf{x}_t, \mathbf{y}_t)$. In particular, in addition to the the KWSA scheme and the random directions scheme, as outlined in (4) and (5), we employ an improvised random directions gradient estimator (I-RDSA) by sampling m directions at each time followed by averaging, i.e., $\{\mathbf{z}_{i,t}\}_{i=1}^m$ for which we have,

$$\mathbf{g}_m(\mathbf{x}_t; \mathbf{y}_t, \mathbf{z}_{i,t})$$
$$= \frac{1}{m} \sum_{i=1}^m \left(\frac{F(\mathbf{x}_t + c_t \mathbf{z}_{i,t}; \mathbf{y}) - F(\mathbf{x}_t; \mathbf{y})}{c_t} \mathbf{z}_{i,t} \right). \quad (14)$$

It is to be noted that the above gradient approximation scheme uses more exactly one data point while utilizing m directional derivatives. In order to quantify the benefits of using such a scheme, we present the statistics concerning the gradient approximation of RDSA and I-RDSA. We have from (Duchi et al., 2015) for RDSA,

$$\mathbb{E}_{\mathbf{z}_t \sim \mu, \mathbf{y}_t \sim \mathcal{P}} [\mathbf{g}(\mathbf{x}; \mathbf{y}_t, \mathbf{z}_t)] = \nabla f(\mathbf{x}) + c_t L \mathbf{v}(\mathbf{x}, c_t)$$
$$\mathbb{E}_{\mathbf{z}_t \sim \mu, \mathbf{y}_t \sim \mathcal{P}} \left[\|\mathbf{g}(\mathbf{x}; \mathbf{y}_t, \mathbf{z}_t)\|^2\right] \le 2s(d) \mathbb{E}\left[\|\nabla F(\mathbf{x}; \mathbf{y}_t)\|^2\right]$$
$$+ \frac{c_t^2}{2} L^2 M(\mu), \qquad (15)$$

Using (15), similar statistics for the improvised RDSA gradient estimator can be evaluated as follows:

$$\mathbb{E}_{\mathbf{z}_t \sim \mu, \mathbf{y}_t \sim \mathcal{P}} [\mathbf{g}_m(\mathbf{x}; \mathbf{y}_t, \mathbf{z}_t)] = \nabla f(\mathbf{x}) + \frac{c_t}{m} L \mathbf{v}(\mathbf{x}, c_t)$$

Algorithm 2 Stochastic Gradient Free Frank Wolfe
Require: Input, Loss Function $F(x)$, Convex Set $\mathcal{C}$, number of directions m, sequences $\gamma_t = \frac{2}{t+8}$,
$$(\rho_t, c_t)_{RDSA} = \left(\frac{4}{d^{1/3}(t+8)^{2/3}}, \frac{2}{d^{3/2}(t+8)^{1/3}} \right)$$
$$(\rho_t, c_t)_{I-RDSA} = \left(\frac{4}{\left(1 + \frac{d}{m}\right)^{1/3}(t+8)^{2/3}}, \frac{2\sqrt{m}}{d^{3/2}(t+8)^{1/3}} \right)$$
$$(\rho_t, c_t)_{KWSA} = \left(\frac{4}{(t+8)^{2/3}}, \frac{2}{d^{1/2}(t+8)^{1/3}} \right).$$
Output: $\mathbf{x}_T$ or $\frac{1}{T} \sum_{t=0}^{T-1} \mathbf{x}_t$.
1: Initialize $\mathbf{x}_0 \in \mathcal{C}$
2: **for** $t = 0, 2, \ldots, T - 1$ **do**
3:     Compute
       KWSA:
       $\mathbf{g}(\mathbf{x}_t; \mathbf{y}) = \sum_{i=1}^d \frac{F(\mathbf{x}_t + c_t \mathbf{e}_i; \mathbf{y}) - F(\mathbf{x}_t; \mathbf{y})}{c_t} \mathbf{e}_i$
       RDSA: Sample $\mathbf{z}_t \sim \mathcal{N}(0, \mathbf{I}_d)$,
       $\mathbf{g}(\mathbf{x}_t; \mathbf{y}, \mathbf{z}_t) = \frac{F(\mathbf{x}_t + c_t \mathbf{z}_t; \mathbf{y}) - F(\mathbf{x}_t; \mathbf{y})}{c_t} \mathbf{z}_t$
       I-RDSA: Sample $\{\mathbf{z}_{i,t}\}_{i=1}^m \sim \mathcal{N}(0, \mathbf{I}_d)$,
       $\mathbf{g}(\mathbf{x}_t; \mathbf{y}, \mathbf{z}_t) = \frac{1}{m} \sum_{i=1}^m \frac{F(\mathbf{x}_t + c_t \mathbf{z}_{i,t}; \mathbf{y}) - F(\mathbf{x}_t; \mathbf{y})}{c_t} \mathbf{z}_{i,t}$
4:     Compute $\mathbf{d}_t = (1 - \rho_t) \mathbf{d}_{t-1} + \rho_t \mathbf{g}(\mathbf{x}_t, \mathbf{y}_t)$
5:     Compute $\mathbf{v}_t = \operatorname{argmin}_{\mathbf{s} \in \mathcal{C}} \langle \mathbf{s}, \mathbf{d}_t \rangle$,
6:     Compute $\mathbf{x}_{t+1} = (1 - \gamma_t) \mathbf{x}_t + \gamma_t \mathbf{v}_t$.
7: **end for**

$$\mathbb{E}_{\mathbf{z}_t \sim \mu, \mathbf{y}_t \sim \mathcal{P}} \left[\|\mathbf{g}_m(\mathbf{x}; \mathbf{y}_t, \mathbf{z}_t)\|^2\right] \le \left(\frac{1+m}{2m} \right) c_t^2 L^2 M(\mu)$$
$$+ 2 \left(1 + \frac{s(d)}{m} \right) \mathbb{E}\left[\|\nabla F(\mathbf{x}; \mathbf{y}_t)\|^2\right], \qquad (16)$$

where $\|\mathbf{v}(\mathbf{x}, c_t)\| \le \frac{1}{2} \mathbb{E}\left[\|\mathbf{z}\|^3\right]$. A proof for (16) can be found in (Liu et al., 2018). As we will see later the I-RDSA scheme improves the dimension dependence of the primal gap, but it comes at the cost of m calls to the SZO. We are ready to state the zeroth order stochastic Frank-Wolfe algorithm which is presented in algorithm 2. Before the main results, we first study the evolution of the gradient estimates in (12) and the associated mean square error. The following Lemma studies the error of the process $\{\mathbf{d}_t\}$ as defined in (12).

Lemma 3.2. *Let Assumptions A1-A5 hold. Given the recursion in (12), we have that $\|\nabla f(\mathbf{x}_t) - \mathbf{d}_t\|^2$ satisfies*

1) for the RDSA gradient approximation scheme

$$\mathbb{E}\left[\|\nabla f(\mathbf{x}_t) - \mathbf{d}_t\|^2\right] \le 2\rho_t^2 \sigma^2 + 4\rho_t^2 L_1^2 + 8\rho_t^2 s(d) L_1^2$$
$$+ 2\rho_t^2 c_t^2 L^2 M(\mu) + \frac{2L^2 R^2 \gamma_t^2}{\rho_t} + \frac{\rho_t}{2} c_t^2 L^2 M(\mu)$$
$$+ \left(1 - \frac{\rho_t}{2} \right) \mathbb{E}\left[\|\nabla f(\mathbf{x}_{t-1}) - \mathbf{d}_{t-1}\|^2\right], \qquad (17)$$

2) for the I-RDSA gradient approximation scheme

$$\mathbb{E}\left[\|\nabla f(\mathbf{x}_t) - \mathbf{d}_t\|^2\right] \le 2\rho_t^2 \left(\sigma^2 + 2L_1^2\right)$$
$$+ \frac{\rho_t}{2m^2} c_t^2 L^2 M(\mu) + 8\rho_t^2 \left(1 + \frac{s(d)}{m} \right) L_1^2$$

$$+ \left(\frac{1+m}{2m} \right) \rho_t^2 c_t^2 L^2 M(\mu) + \frac{2L^2 R^2 \gamma_t^2}{\rho_t}$$
$$+ \left(1 - \frac{\rho_t}{2} \right) \mathbb{E} \left[\| \nabla f(\mathbf{x}_{t-1}) - \mathbf{d}_{t-1} \|^2 \right] \quad (18)$$

3) for the KWSA gradient approximation scheme

$$\mathbb{E} \left[\| \nabla f(\mathbf{x}_t) - \mathbf{d}_t \|^2 \right] \leq 2\rho_t^2 \sigma^2 + 2\rho_t c_t^2 dL^2$$
$$+ \frac{2L^2 R^2 \gamma_t^2}{\rho_t} + \left(1 - \frac{\rho_t}{2} \right) \mathbb{E} \left[\| \nabla f(\mathbf{x}_{t-1}) - \mathbf{d}_{t-1} \|^2 \right] . \tag{19}$$

We use the following Lemma so as to study the dynamics of the primal gap.

Lemma 3.3. *Consider the zeroth order Frank Wolfe Algorithm in 1. Let Assumptions A1-A5 hold. Then, the primal gap $F(\mathbf{x}_{t+1}) - F(\mathbf{x}^*)$ satisfies*

$$F(\mathbf{x}_{t+1}) - F(\mathbf{x}^*) \leq (1 - \gamma_{t+1})(F(\mathbf{x}_t) - F(\mathbf{x}^*))$$
$$+ \gamma_{t+1} R \| \nabla F(\mathbf{x}_t) - \mathbf{d}_t \| + \frac{LR^2 \gamma_{t+1}^2}{2} . \tag{20}$$

Proof. The proof is relegated to the Appendix A. □

With the above recursions in place, we can now characterize the finite time rates of the mean square errors for the different error approximation schemes. In particular, using Lemma 3.2, we first state the main result concerning the setting, where the objective is convex.

3.2.1   Main Results: Convex Case

In this section, we state the main results. We first state the main results concerning the primal gap of the proposed algorithm.

Primal Gap: We state the main results involving the different gradient approximation schemes for the primal gap, which provide a characterization of $\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^*)]$.

Theorem 3.4. *Let Assumptions A1-A5 hold. Let the sequence γ_t be given by $\gamma_t = \frac{2}{t+8}$.*

1) Then, we have the following primal sub-optimality gap for the algorithm in 2, with the RDSA gradient approximation scheme:

$$\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^*)] = O \left(\frac{d^{1/3}}{(t+9)^{1/3}} \right) . \tag{21}$$

2) In case of the I-RDSA the gradient approximation scheme, the primal sub-optimality gap is given by,

$$\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^*)] = O \left(\frac{(d/m)^{1/3}}{(t+9)^{1/3}} \right) . \tag{22}$$

3) Finally, for the KWSA gradient approximation scheme, the primal sub-optimality gap is given by,

$$\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^*)] = O \left(\frac{1}{(t+9)^{1/3}} \right) . \tag{23}$$

Theorem 3.4 quantifies the dimension dependence of the primal gap to be $d^{1/3}$. At the same time the dependence on iterations, i.e., $O(T^{-1/3})$ matches that of the stochastic Frank-Wolfe which has access to first order information as in (Mokhtari et al., 2018). The improvement of the rates for I-RDSA and KWSA are at the cost of extra directional derivatives at each iteration. The number of queries to the SZO so as to obtain a primal gap of ϵ, i.e., $\mathbb{E}[f(\mathbf{x}_t) - f(\mathbf{x}^*)] \leq \epsilon$ is given by $O\left(\frac{d}{\epsilon^3}\right)$, where the dimension dependence is consistent with zeroth order schemes and cannot be improved on as illustrated in (Duchi et al., 2015). The proofs for parts (1), (2) and (3) for theorem 3.4 can be found in the appendix in the sections B, C and D respectively.

Dual Gap: We state the main results involving the different gradient approximation schemes for the dual gap, which provide a characterization of $\mathcal{G}(\mathbf{x}) = \max_{\mathbf{v} \in \mathcal{C}} \langle \nabla F(\mathbf{x}), \mathbf{x} - \mathbf{v} \rangle$.

Theorem 3.5. *Let Assumptions A1-A5 hold. Let the sequence γ_t be given by $\gamma_t = \frac{2}{t+8}$.*

1) Then, we have the following dual gap for the algorithm in 2, with the RDSA gradient approximation scheme:

$$\mathbb{E} \left[\min_{t=0,\cdots,T-1} \mathcal{G}(\mathbf{x}_t) \right] \leq \frac{7(F(\mathbf{x}_0) - F(\mathbf{x}^*))}{2T}$$
$$+ \frac{LR^2 \ln(T+7)}{T} + \frac{Q' + R\sqrt{2Q}}{2T}(T+7)^{2/3}, \quad (24)$$

where $Q = 32d^{-1/3}\sigma^2 + 64d^{-1/3}L_1^2 + 128d^{2/3}L_1^2 + 2L^2R^2d^{2/3} + 416d^{2/3}L^2$ and $Q' = \max\{2(f(\mathbf{x}_0) - f(\mathbf{x}^*)), 2R\sqrt{2Q} + LR^2/2\}$.

2) In case of the I-RDSA the gradient approximation scheme, the dual gap is given by,

$$\mathbb{E} \left[\min_{t=0,\cdots,T-1} \mathcal{G}(\mathbf{x}_t) \right] \leq \frac{7(F(\mathbf{x}_0) - F(\mathbf{x}^*))}{2T}$$
$$+ \frac{LR^2 \ln(T+7)}{T} + \frac{Q'_{ir} + R\sqrt{2Q_{ir}}}{2T}(T+7)^{2/3}, \tag{25}$$

where $Q_{ir} = 32(1 + d/m)^{-1/3}\sigma^2 + 128(1 + d/m)^{2/3}L_1^2 + 64(1 + d/m)^{-1/3}L_1^2 + 2L^2R^2(1 + d/m)^{2/3} + 416(1 + d/m)^{2/3}L^2$ and $Q'_{ir} = \max\{2(f(\mathbf{x}_0) - f(\mathbf{x}^*)), 2R\sqrt{2Q_{ir}} + LR^2/2$.

3) Finally, for the KWSA gradient approximation scheme, the dual gap is given by,

$$\mathbb{E} \left[\min_{t=0,\cdots,T-1} \mathcal{G}(\mathbf{x}_t) \right] \leq \frac{7(F(\mathbf{x}_0) - F(\mathbf{x}^*))}{2T}$$

$$+ \frac{LR^2 \ln(T+7)}{T} + \frac{Q'_{kw} + R\sqrt{2Q_{kw}}}{2T}(T+7)^{2/3},$$
$$(26)$$

where $Q_{kw} = \max\left\{4\|\nabla f(\mathbf{x}_0) - \mathbf{d}_0\|^2, \ 32\sigma^2 + 32L^2 + 2L^2R^2\right\}$ and $Q'_{kw} = \max\{2(f(\mathbf{x}_0) - f(\mathbf{x}^*)), 2R\sqrt{Q_{kw}} + LR^2/2\}.$

Theorem 3.5 quantifies the dimension dependence of the Frank-Wolfe duality gap to be $d^{1/3}$. At the same time the dependence on iterations, i.e., $O(T^{-1/3})$ matches that of the primal gap and hence follows that the number of queries to the SZO so as to obtain a Frank-Wolfe duality gap of ϵ, i.e., $\mathbb{E}\left[\min_{t=0,\cdots,T-1} \mathcal{G}(\mathbf{x}_t)\right] \leq \epsilon$ is given by $O\left(\frac{d}{\epsilon^3}\right)$. In particular, theorem 3.5 asserts that the initial conditions are forgotten as $O(1/T)$. The proofs for parts (1), (2) and (3) for theorem 3.5 can be found in the appendix in the sections B, C and D respectively.

3.2.2 Zeroth-Order Frank-Wolfe Non-Convex

We employ the following algorithm for the non-convex stochastic Frank-Wolfe:

Algorithm 3 Stochastic Gradient Free Frank-Wolfe

Require: Input, Loss Function $F(x)$, Convex Set $\mathcal{C}$, number of directions m. Sequences $\gamma = \frac{1}{T^{3/4}}$,
$$(\rho_t, c_t) = \left(\frac{4}{\left(1+\frac{d}{m}\right)^{1/3}(t+8)^{2/3}}, \frac{2\sqrt{m}}{(d^{3/2}(t+8)^{1/3})}\right)$$
Output: $\mathbf{x}_T$.
1: Initialize $\mathbf{x}_0 \in \mathcal{C}$
2: **for** $t = 0, 1, \ldots, T-1$ **do**
3:    Compute
     Sample $\{\mathbf{z}_{i,t}\}_{i=1}^m \sim \mathcal{N}(0, \mathbf{I}_d)$, $\mathbf{g}(\mathbf{x}_t; \mathbf{y}, \mathbf{z}_t) = \frac{1}{m}\sum_{i=1}^m \frac{F(\mathbf{x}_t + c_t \mathbf{z}_{i,t}; \mathbf{y}) - F(\mathbf{x}_t; \mathbf{y})}{c_t}\mathbf{z}_{i,t}$
4:    Compute $\mathbf{d}_t = (1-\rho_t)\mathbf{d}_{t-1} + \rho_t g(\mathbf{x}_t, \mathbf{y})$
5:    Compute $\mathbf{v}_t = \operatorname{argmin}_{\mathbf{s}\in\mathcal{C}}\langle \mathbf{s}, \mathbf{d}_t\rangle\rangle$,
6:    Compute $\mathbf{x}_{t+1} = (1-\gamma)\mathbf{x}_t + \gamma\mathbf{v}_t$.
7: **end for**

We use the following assumption concerning the smoothness of the non-convex loss function.

Assumption A6. The gradients ∇f are L-Lipschitz continuous over the set $\mathcal{C}$, i.e., for all $x, y \in \mathcal{C}$

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|.$$

Theorem 3.6. *Let Assumptions A4-A6 hold. Then, we have the following dual gap for iterations $t = 0, 1, \cdots, T-1$ for the algorithm as described in (13)*

$$\mathbb{E}\left[\min_{t=0,\cdots,T-1} \mathcal{G}(\mathbf{x}_t)\right] \leq \frac{Q'}{T^{1/4}} = O\left(\frac{(d/m)^{1/3}}{T^{1/4}}\right),$$
$$(27)$$

where $Q' = \max\{9^{1/3}(f(\mathbf{x}_0) - f(\mathbf{x}^*)), Q_{nc}R(d/m)^{1/3}\}.$

Theorem 3.6 quantifies the dimension dependence of the Frank-Wolfe duality gap for non-convex functions to be $d^{1/3}$. At the same time the dependence on iterations, i.e., $O(T^{-1/4})$ matches that of the rate of SFW in (Reddi et al., 2016) and hence follows that the number of queries to the SZO so as to obtain a Frank-Wolfe duality gap of ϵ, i.e., $\mathbb{E}\left[\min_{t=0,\cdots,T-1} \mathcal{G}(\mathbf{x}_t)\right] \leq \epsilon$ is given by $O\left(\frac{d^{4/3}}{\epsilon^4}\right)$. The proof is relegated to the appendix in section E.

4 Experiments

We now present empirical results for zeroth order Frank-Wolfe optimization with an aim to highlight three aspects of our method: (i) it is accurate even in stochastic case (Section 4.1) (ii) it scales to relatively high dimensions (Section 4.2) (iii) it reaches stationary point in non-convex setting (Section 4.3).

Methods and Evaluation We look at the optimality gap $|f(x_{\text{optimizer}}) - f(x^*)|$ as the evaluation metric, where $x_{\text{optimizer}}$ denotes the solution obtained from the employed optimizer and x^* corresponds to true solution. Most existing zero order optimization techniques like Nelder-Mead simplex (Nelder and Mead, 1965) or bound optimization by quadratic approximation (BOBYQA; Powell 2009) can only handle bound constraints, but not arbitrary convex constraints as our method can. Thus, for all experiments, we could compare proposed zeroth order stochastic Frank-Wolf (0-FW) only with COBYLA, a constrained optimizer by linear approximation, which is popular in engineering fields (Powell, 1994). For experiments where SFO is available, we additionally compare with stochastic proximal gradient descent (PGD) and first order stochastic Frank-Wolfe method (1-FW).

4.1 Stochastic Lasso Regression

To study performance of various stochastic optimization, we solve a simple lasso regression on the dataset covtype ($n = 581012$, $d = 54$) from libsvm website[2]. We use the variant with feature values in $[0, 1]$ and solve the following problem:

$$\min_{\|\mathbf{w}\|_1 \leq 1} \frac{1}{2}\|\mathbf{y} - \mathbf{X}^\top \mathbf{w}\|_2^2$$

where $\mathbf{X} \in \mathbb{R}^{n\times d}$ represents the feature vectors and $\mathbf{y} \in \mathbb{R}^n$ are the corresponding targets.

For the 0-FW, we used I-RDSA with $m = 6$. This problem represents a stochastic setting and from Figure 1a we note that the performance of 0-FW matches that of 1-FW in terms of the number of oracle calls

---
[2]Available at `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`

(a) Covtype Dataset

(b) Cox Regression Dataset
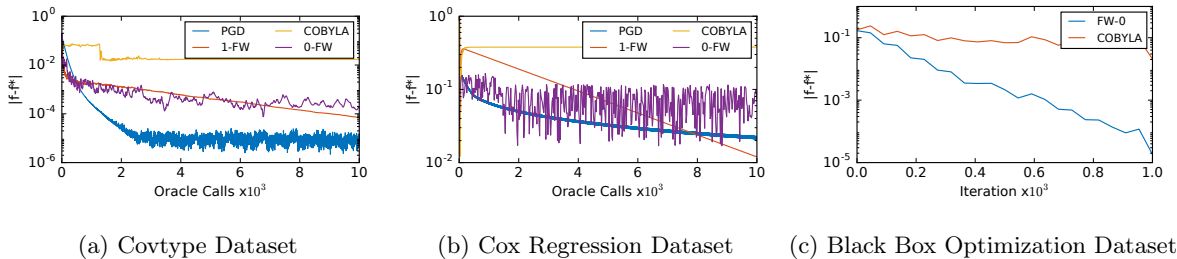
(c) Black Box Optimization Dataset

Figure 1: Comparison of proposed zeroth order Frank-Wolfe (0-FW) with first order Frank-Wolfe (1-FW), proximal gradient descent (PGD), and another zero order constrained optimaztion by linear approximation (COBYLA) on various problems.

to their respective oracles in spite of the dimension involved being $d = 54$.

4.2 High Dimensional Cox Regression

To demonstrate efficacy of zeroth order Frank-Wolfe optimization in a moderately high dimensional case, we look at gene expression data. In particular, we perform patient survival analysis by solving Cox regression (also known as proportional hazards regression) to relate different gene expression profiles with survival time (Sohn et al., 2009). We use the Kidney renal clear cell carcinoma dataset[3], which contains gene expression data for 606 patients (534 with tumor and 72 without tumor) along with survival time information. We preprocess the dataset by eliminating the rarely expressed genes, i.e. we only keep genes expressed in 50% of the patients. This leads to a feature vector x_i of size 9376 for each patient i. Also, for each patient i, we have the censoring indicator variable y_i that takes the value 0 if patient is alive or 1 if death is observed with t_i denoting the time of death. In this setup, we can obtain a sparse solution to cox regression by solving the following problem (Park and Hastie, 2007; Sohn et al., 2009):

$$\min_{\|\mathbf{w}\|_1 \le 10} \frac{1}{n} \sum_{i=1}^{n} y_i \left\{ -\mathbf{x}_i^\top \mathbf{w} + \log \left(\sum_{j \in \mathcal{R}_i} \exp(\mathbf{x}_j^\top \mathbf{w}) \right) \right\}$$

where $\mathcal{R}_i$ is the set of subjects at risk at time t_i, i.e. $\mathcal{R}_i = \{j : t_j \ge t_i\}$.

This problem represents a high-dimensional setting with $d = 9376$. For this setup, we take $m = 938$ for the I-RDSA scheme of our proposed algorithm. Due to the unavoidable dimension dependence of zeroth order schemes, Figure 1b shows the gap between 1-FW and 0-FW to be around $2\times$ and thereby reinforcing the result in Theorem 3.4 (2)

4.3 Black-Box Optimization

Finally, we show efficacy of zeroth order Frank-Wolfe optimization in a non-convex setting for a black-

box optimization. Many engineering problems can be posed as optimizing forward models from physics, which are often complicated, do not posses analytical expression, and cannot be differentiated. We take the example of analyzing electron back-scatter diffraction (EBSD) patterns in order to determine crystal orientation of the sample material. Such analysis is useful in determining strength, malleability, ductility, etc. of the material along various directions. Brute-force search has been the primary optimization technique in use (Ram et al., 2017). For this problem, we use the forward model of EBSD provided by EMSoft[4]. There are $d = 6$ parameters to optimize over the L_∞-ball of radius 1.

This problem represents a non-convex black box optimization setting for which we used $m = 1$ for the I-RDSA, i.e. RDSA. Figure 1c shows that our proposed algorithm converges to a first order stationary point there by showing the effectiveness of our proposed algorithm for black-box optimization.

5 Conclusion

In this paper, we proposed a stochastic zeroth order Frank-Wolfe algorithm. The proposed algorithm does not depend on hard to estimates like Lipschitz constants and thus is easy to deploy in practice. For the proposed algorithm, we quantified the rates of convergence of the proposed algorithm in terms of the primal gap and the Frank-Wolfe duality gap, which we showed to match its first order counterpart in terms of iterations. In particular, we showed that the dimension dependence, when one directional derivative is sampled at each iteration to be $O(d^{1/3})$. We demonstrated the efficacy of our proposed algorithm through experiments on multiple datasets. Natural future directions include extending the proposed algorithm to non-smooth functions and incorporating variance reduction techniques to get better rates.

---

[3]Available at `http://gdac.broadinstitute.org`

[4]Software is available at `https://github.com/EMsoft-org/EMsoft`

References

Krishnakumar Balasubramanian and Saeed Ghadimi. Zeroth-order (non)-convex stochastic optimization via conditional gradient and gradient updates. In *Advances in Neural Information Processing Systems*, pages 3459–3468, 2018.

Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26. ACM, 2017.

Krzysztof Choromanski, Mark Rowland, Vikas Sindhwani, Richard E Turner, and Adrian Weller. Structured evolution with compact architectures for scalable policy optimization. *arXiv preprint arXiv:1804.02395*, 2018.

Stanley N Deming, Lloyd R Parker Jr, and M Bonner Denton. A review of simplex optimization in analytical chemistry. *CRC Critical Reviews in Analytical Chemistry*, 7(3):187–202, 1978.

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

Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

Dan Garber and Elad Hazan. Faster rates for the frank-wolfe method over strongly-convex sets. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*, pages 541–549. JMLR. org, 2015.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

Genetha Anne Gray, Tamara G Kolda, Ken Sale, and Malin M Young. Optimizing an empirical scoring function for transmembrane protein structure determination. *INFORMS Journal on Computing*, 16(4):406–418, 2004.

Elad Hazan and Satyen Kale. Projection-free online learning. In *International Conference on Machine Learning*, pages 1843–1850, 2012.

Elad Hazan and Haipeng Luo. Variance-reduced and projection-free stochastic optimization. In *International Conference on Machine Learning*, pages 1263–1271, 2016.

Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *ICML (1)*, pages 427–435, 2013.

Jack Kiefer and Jacob Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, pages 462–466, 1952.

Simon Lacoste-Julien. Convergence rate of frank-wolfe for non-convex objectives. *arXiv preprint arXiv:1607.00345*, 2016.

Simon Lacoste-Julien and Martin Jaggi. On the global linear convergence of frank-wolfe optimization variants. In *Advances in Neural Information Processing Systems*, pages 496–504, 2015.

Guanghui Lan and Yi Zhou. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2):1379–1409, 2016.

Sijia Liu, Jie Chen, Pin-Yu Chen, and Alfred Hero. Zeroth-order online alternating direction method of multipliers: Convergence analysis and applications. In *International Conference on Artificial Intelligence and Statistics*, pages 288–297, 2018.

Alison L Marsden, Meng Wang, JE Dennis, and Parviz Moin. Trailing-edge noise reduction using derivative-free optimization and large-eddy simulation. *Journal of Fluid Mechanics*, 572:13–36, 2007.

Alison L Marsden, Jeffrey A Feinstein, and Charles A Taylor. A computational framework for derivative-free optimization of cardiovascular geometries. *Computer methods in applied mechanics and engineering*, 197(21-24):1890–1905, 2008.

Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. Conditional gradient method for stochastic submodular maximization: Closing the gap. In *International Conference on Artificial Intelligence and Statistics*, pages 1886–1895, 2018.

John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7 (4):308–313, 1965.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 2011.

Mee Young Park and Trevor Hastie. L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):659–677, 2007.

Michael JD Powell. A direct search optimization method that models the objective and constraint

functions by linear interpolation. In *Advances in optimization and numerical analysis*, pages 51–67. Springer, 1994.

Michael JD Powell. The bobyqa algorithm for bound constrained optimization without derivatives. *Cambridge NA Report NA2009/06, University of Cambridge, Cambridge*, pages 26–46, 2009.

Farangis Ram, Stuart Wright, Saransh Singh, and Marc De Graef. Error analysis of the crystal orientations obtained by the dictionary approach to ebsd indexing. *Ultramicroscopy*, 181:17–26, 2017.

Sashank J Reddi, Suvrit Sra, Barnabás Póczos, and Alex Smola. Stochastic frank-wolfe methods for nonconvex optimization. In *Communication, Control, and Computing (Allerton), 2016 54th Annual Allerton Conference on*, pages 1244–1251. IEEE, 2016.

Andrzej Ruszczyński. A merit function approach to the subgradient method with averaging. *Optimisation Methods and Software*, 23(1):161–172, 2008.

Anit Kumar Sahu, Dusan Jakovetic, Dragana Bajovic, and Soummya Kar. Distributed zeroth order optimization over random networks: A Kiefer-Wolfowitz stochastic approximation approach. In *57th IEEE Conference on Decision and Control (CDC)*, Miami, 2018.

Insuk Sohn, Jinseog Kim, Sin-Ho Jung, and Changyi Park. Gradient lasso for cox proportional hazards model. *Bioinformatics*, 25(14):1775–1781, 2009.

Yining Wang, Simon Du, Sivaraman Balakrishnan, and Aarti Singh. Stochastic zeroth-order optimization in high dimensions. In *International Conference on Artificial Intelligence and Statistics*, pages 1356–1365, 2018.

Yang Yang, Gesualdo Scutari, Daniel P Palomar, and Marius Pesavento. A parallel decomposition method for nonconvex stochastic multi-agent optimization problems. *IEEE Transactions on Signal Processing*, 64(11):2949–2964, 2016.