Hessian-Aware Zeroth-Order Optimization for Black-Box Adversarial Attack

Haishan Ye [*] Zhichao Huang [*] Cong Fang [†] Chris Junchi Li [‡] Tong Zhang [‡]

HKUST and Peking University

January 1, 2019

Abstract

Zeroth-order optimization or derivative-free optimization is an important research topic in machine learning. In recent, it has become a key tool in black-box adversarial attack to neural network based image classifiers. However, existing zeroth-order optimization algorithms rarely extract Hessian information of the model function. In this paper, we utilize the second-order information of the objective function and propose a novel *Hessian-aware zeroth-order algorithm* called `ZO-HessAware`. Our theoretical result shows that `ZO-HessAware` has an improved zeroth-order convergence rate and query complexity under structured Hessian approximation, where we propose a few approximation methods of such. Our empirical studies on the black-box adversarial attack problem validate that our algorithm can achieve improved success rates with a lower query complexity.

Contents

[*] email: yhs12354123@gmail.com; zhuangbx@connect.ust.hk
[†] email: fangcong@pku.edu.cn
[‡] email: junchi.li.duke@gmail.com; tongzhang@tongzhang-ml.org

1 Introduction

In this paper, we consider the following convex optimization problem:

$$\min_{x\in\mathbb{R}^d} f(x) \tag{1.1}$$

where f is differentiable and strongly convex. Optimization method that solves the above problem with function value access only is known as *zeroth-order optimization* or *black-box optimization* (Nesterov & Spokoiny, 2017; Ghadimi & Lan, 2013).

Zeroth-order optimization has attracted attention from the machine learning community (Bergstra et al., 2011; Ilyas et al., 2018) and it is especially useful for solving problem (1.1) where the evaluations of gradients $\nabla f(x)$ are difficult or even infeasible. One prominent example of the zeroth-order optimization is the *black-box adversarial attack* on deep neural networks (Chen et al., 2017; Hu & Tan, 2017; Papernot et al., 2017; Ilyas et al., 2018). In the black-box adversarial attack, only the inputs and outputs of the neural network are available to the system and backpropagation on the target neural network is prohibited. Another application example is the hyper-parameter tuning which searches for the optimal parameters of deep neural networks or other learning models (Snoek et al., 2012; Bergstra et al., 2011).

In the past years, theoretical works on zeroth-order optimization arise as alternatives of the corresponding first-order methods, and they estimate gradients using function value difference (Nesterov & Spokoiny, 2017; Ghadimi & Lan, 2013; Duchi et al., 2015). However, these works on zeroth-order optimization have been concentrating on extracting gradient information of the objective function and *failed* to utilize the second-order Hessian information and, to some extent, have *not* fully exploited the information of models and enjoyed less competitive rates. In this paper, we aim to take advantages of the model's second-order information and propose a novel method called *Hessian-aware* zeroth-order optimization. Aligning with earlier works Nesterov & Spokoiny (2017); Ghadimi & Lan (2013) we present in this paper our gradient estimation as follows:

$$g_\mu(x) = \frac{1}{b}\sum_{i=1}^{b} \frac{f(x+\mu\tilde{H}^{-1/2}u_i) - f(x)}{\mu} \cdot \tilde{H}^{1/2}u_i, \quad \text{with } \mu > 0 \tag{1.2}$$

where b is the batch size of points for gradient estimation and $\tilde{H}$ is an approximate Hessian at the evaluation point. With Eqn. (1.2) at hands, the core update rule of our Hessian-aware zero-order algorithm, namely `ZO-HessAware`, is

$$x_{t+1} = x_t - \eta\tilde{H}^{-1}g_\mu(x_t) = x_t - \eta\tilde{H}^{-1} \cdot \frac{1}{b}\sum_{i=1}^{b} \frac{f(x+\mu\tilde{H}^{-1/2}u_i) - f(x)}{\mu} \cdot \tilde{H}^{1/2}u_i,$$

where η is the step size. If one lets $\tilde{g}_\mu(x)$ be defined as

$$\tilde{g}_\mu(x) \triangleq \frac{1}{b}\sum_{i=1}^{b} \frac{f(x+\mu\tilde{u}_i) - f(x)}{\mu} \cdot \tilde{u}_i, \quad \text{with } \tilde{u}_i \sim N(0, \tilde{H}^{-1}), \tag{1.3}$$

then using the linear transformation property of the multivariate Gaussian distribution, the update rule further reduces to

$$x_{t+1} = x_t - \eta\tilde{g}_\mu(x_t). \tag{1.4}$$

In comparison, early zeroth-order literatures (for instance Nesterov & Spokoiny (2017)) conduct gradient estimation via

$$\hat{g}_\mu(x) \triangleq \frac{1}{b}\sum_{i=1}^{b}\frac{f(x+\mu u_i)-f(x)}{\mu}u_i \quad \text{with } u_i \sim N(0, I_d). \tag{1.5}$$

Comparing (1.3) and (1.5), one observes that the directions of updates $\tilde{g}_\mu$ and $\hat{g}_\mu$ share the same form but admit different covariances. Because $\tilde{g}_\mu(x)$ contains Hessian information and shares the same form with estimated gradient $\hat{g}_\mu(x)$, $\tilde{g}_\mu(x)$ can be regarded as a *natural gradient* and ZO-HessAware can be regarded as *natural gradient descent* method. Perhaps surprising, in the context of zeroth-order optimization the difference between our zeroth-order update and earlier works boil down to the difference of search direction covariances. Our special choice of covariance matrix as the approximate inversed Hessian allows us to incorporate Hessian information into the update rule in Eqn. (1.4), which further achieves an improved theoretical convergence rate in zeroth-order optimization.

Let $\tilde{H}$ be an approximate Hessian satisfy $\rho\tilde{H} \preceq \nabla^2 f(x) \preceq (2-\rho)\tilde{H}$ with $0 < \rho \le 1$, and the algorithm is initialized at a point sufficiently close to the optimal solution, then in order to obtain an ϵ-accuracy, our ZO-Hess algorithm with a proper step size achieves an iteration complexity of $N(\epsilon) = O\left(\frac{d}{b\rho}\log\left(\frac{1}{\epsilon}\right)\right)$. Note if the objective function has the strong convexity parameter τ, ρ can be chosen as τ/λ_{k+1} where λ_{k+1} is the $(k+1)$-th largest eigenvalue of the Hessian, and ZO-HessAware enjoys an iteration complexity of

$$N(\epsilon) = O\left(\frac{d\lambda_{k+1}}{b\tau}\log\left(\frac{1}{\epsilon}\right)\right).$$

Furthermore, let ZO-HessAware be implemented with power-method based Hessian approximation (named ZOHA-PW), it then achieves following query complexity

$$Q(\epsilon) = \tilde{O}\left(\frac{d\lambda_{k+1}}{\tau}\log\left(\frac{1}{\epsilon}\right)\right).$$

Seeing that $\lambda_{k+1} \le L$ always holds, our proposed ZO-HessAware algorithm enjoys a sharper theoretical convergence rate and query complexity than the comparable version using no Hessian information (recall that Nesterov & Spokoiny (2017) indicates an iteration complexity of $O\left(\frac{dL}{b\tau}\log\left(\frac{1}{\epsilon}\right)\right)$ and a query complexity of $O\left(\frac{dL}{\tau}\log\left(\frac{1}{\epsilon}\right)\right)$ where L is the smoothness parameter of the objective function).

Though ZOHA-PW obtains a nice theoretical result of query complexity, it takes at least $O(d)$ queries due to the power method procedure. This is very expensive especially when the dimension d is very large. Hence, we also propose several heuristic but practical methods to construct approximate Hessians with much lower query complexity.

(i) First, we use Gaussian sampling method to approximate Hessian. This method only samples a small batch of points from the Gaussian distribution to estimate the Hessian with batch size being much smaller than the data dimension d.

(ii) Furthermore, we propose diagonal Hessian approximation which is a popular method in training deep neural networks. This approximation approach does not need extra query to function value and can keep principal information of the Hessian which has been proved in training deep neural networks (Kingma & Ba, 2015; Duchi et al., 2011; Zeiler, 2012).

To numerically justify the effectiveness of our zeroth-order algorithm `ZO-HessAware`, we apply our algorithm with Hessian approximation approaches to the task of black-box adversarial attack in the neural network based image classifier (Ilyas et al., 2018; Chen et al., 2017). The adversarial attack aims to find an example x of the given image x_0 with small noise but misclassified by the neural network.

(i) We compare our algorithms with two state-of-the-art algorithms, `PGD-NES` and `ZOO` (Ilyas et al., 2018; Chen et al., 2017). The comparison shows that our Hessian-aware zeroth-order algorithms take much less queries to the function value while obtaining a better success rate of attack. Especially when the attack task is hard, our `ZO-HessAware` type algorithms achieve much better success rates than the state-of-the-art algorithms. Our experiment results also reveal such a potential that Hessian information is a key tool to promote the success rate of the black-box adversarial attack when the attack task is very hard.

(ii) To promote the attack success rate and reduce the query complexity, we propose a novel strategy called `Descent-Checking`. `Descent-Checking` empirically can bring a higher success rate and a lower query complexity. This benefit is more evident when the attack task is hard.

1.1 Main Contribution

We summarize our main contribution as follows.

(i) We exploit the Hessian information of the model function and propose a novel Hessian-aware zeroth-order algorithm called `ZO-HessAware`. It is our creation that we integrate Hessian information into gradient estimation while keeping the algorithmic form similar to zeroth-order based gradient descent method. Theoretically we show `ZO-HessAware` has a faster convergence rate and lower query complexity with power-method based Hessian approximation than existing work without Hessian information.

(ii) Several novel structured Hessian approximation methods are proposed including Gauss sampling method as well as the diagonalization method. The Hessian estimation via Gauss sampling is our creation to the best of our knowledge. It only takes a few extra queries to the function value. In the construction of diagonal approximate Hessian, we use natural gradient which contains Hessian information other than a ordinary gradient which is used in training deep neural networks.

(iii) We propose a descent-checking trick for the black-box adversarial attack. This trick can significantly improve the success rate and reduce the number of queries.

(iv) We empirically prove the power of Hessian information in zeroth-order optimization especially in the black-box adversarial attack. Experiment results show that our `ZO-HessAware` type algorithm can achieve better success rates and need fewer queries than state-of-the-art algorithms especially when the problem is hard.

1.2 Related Work

Zeroth-order optimization minimizes functions only through the function value oracles. It is an important research topic in optimization (Nesterov & Spokoiny, 2017; Matyas, 1965; Ghadimi & Lan, 2013). Nesterov & Spokoiny (2017) utilized random Gaussian vectors as the search directions and gave the convergence properties of the zeroth-order algorithms when the objective function is convex. Ghadimi & Lan (2013) proposed new zeroth-order algorithms which has better convergence rate when the problem is non-smooth. Zeroth-order method with variance reduction was proposed to solve non-convex problem recently (Fang et al., 2018; Liu et al., 2018). Zeroth-order algorithm is also a crucial research topic in the on-line learning. Lots of results were obtained in the recent years (Shamir, 2017; Bach & Perchet, 2016; Duchi et al., 2015). In these works, one can only access to the function value and uses this feed-back to approximate the gradient or sub-gradient.

Recently, zeroth-order algorithm is becoming the main tool for the black-box adversarial attack (Chen et al., 2017; Ilyas et al., 2018). Chen et al. (2017) extended the CW (Carlini & Wagner, 2017) attack which is a powerful white-box method to the black-box attack and proposed `ZOO`. Algorithm `ZOO` can be viewed as a kind of zeroth-order stochastic coordinate descent (Chen et al., 2017). It chooses a coordinate randomly, then uses zeroth-order oracles to estimate the gradient of current coordinate. However, `ZOO` suffers from a poor query complexity because it needs $O(d)$ queries to estimate the gradients of all the coordinates theoretically. To reduce the query complexity, Ilyas et al. (2018) resorted to the natural evolutionary strategies (Wierstra et al., 2014) to estimate gradients. And Ilyas et al. (2018) used the so-called 'antithetic sampling' technique (Salimans et al., 2017) to get better performance.

Furthermore, covariance matrix adaptation evolution strategy (`CMA-ES`) is another important zeroth-order method which is closely related to our algorithm (Hansen & Ostermeier, 2001). `CMA-ES` uses a learned covariance to generate search direction and this covariance matrix is much like the inversion of our approximate Hessian. The main difference between these two algorithms is the way to use zeroth-order oracles. Eqn. (1.3) shows that `ZO-HessAWare` queries to function value to approximate a natural gradient. In contrast, `CMA-ES` generates u_i's and picks up such $\tilde{u}_i$'s as the search directions that $f(x+\tilde{u}_i)$ is mall.

Organization. The rest of this paper is organized as follows. In Section 2, we present notation and preliminaries. In Section 3, we depict Algorithm `ZO-HessAware` in detail and analyze its local and global convergence rate, and query complexity with power-method based Hessian approximation, respectively. In Section 4, we propose two different strategies to construct a good approximate Hessian. In Section 5, we compare our `ZO-HessAware` type algorithms with two state-of-the-art algorithms in the adversarial attack problem. Finally, we conclude our work in Section 6. All the detailed proofs are deferred to the appendix in their order of appearance.

2 Notation and Preliminaries

We first introduce notation that will be used in this paper. Then, we give some assumptions about the objective function that will be used.

2.1 Notation

Given a positive semi-definite matrix $A \in \mathbb{R}^{d\times d}$ of rank-ℓ and a positive integer $k \le \ell$, its eigenvalue decomposition is given as

$$A = U\Lambda U^T = U_k \Lambda_k U_k^T + U_{\setminus k}\Lambda_{\setminus k}U_{\setminus k}^T, \tag{2.1}$$

where U_k and $U_{\setminus k}$ contain the eigenvectors of A, and $\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_d)$ with $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_\ell > 0$ are the nonzero eigenvalues of A. We also use $\lambda_{\max}$ and $\lambda_{\min}$ to denote the largest and smallest eigenvalue of a positive semi-definite matrix, respectively.

Using matrix A, we can define A-norm as $\|x\|_A = \sqrt{x^T A x}$. Furthermore, if B is a positive semi-definite matrix, we say $B \preceq A$ when $A - B$ is positive semi-definite.

2.2 Properties of Smoothness and Convexity

In this paper, we consider functions with L-smoothness and τ-strongly convexity. It indicates the following properties.

L-smoothness If function $f(x)$ is L-smooth, then we have

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|, \; x, y \in \mathbb{R}^d, \tag{2.2}$$

and

$$|f(y) - f(x) - \langle \nabla f(x), y - x\rangle| \le \frac{L}{2}\|x - y\|^2, \; x, y \in \mathbb{R}^d \tag{2.3}$$

τ-strong convexity If function $f(x)$ is τ-strongly convex, then we have

$$f(y) \ge f(x) + \langle \nabla f(x), y - x\rangle + \frac{\tau}{2}\|x - y\|^2, \; x, y \in \mathbb{R}^d,$$

and

We also assume that the Hessian of $f(x)$ is γ-Lipschitz continuous, that is,

$$\left\|\nabla^2 f(y) - \nabla^2 f(x)\right\| \le \gamma\|y - x\|, \; x, y \in \mathbb{R}^d, \tag{2.4}$$

and

$$\left| f(y) - f(x) - \langle \nabla f(x), y - x\rangle - \frac{1}{2}\left\langle \nabla^2 f(x)(y - x), y - x\right\rangle \right| \le \frac{\gamma}{6}\|x - y\|^3, \; x, y \in \mathbb{R}^d \tag{2.5}$$

2.3 Gaussian Smoothing

Let $f(x)$ be a function which is differentiable along any direction in $\mathbb{R}^d$. The *Gaussian smoothing* of $f(x)$ is defined as

$$f_\mu(x) \triangleq \frac{1}{M}\int_{\mathbb{R}^d} f(x+\mu u)\ \exp\left(-\frac{\|u\|^2}{2}\right) du, \text{ where } u \sim N(0, I_d), \tag{2.6}$$

and

$$M = \int_{\mathbb{R}^d} \exp\left(-\frac{\|u\|^2}{2}\right) du = (2\pi)^{d/2}.$$

And μ is the parameter to control the smoothness. $f_\mu(x)$ preserves several important properties of $f(x)$. For example, if $f(x)$ is convex, then $f_\mu(x)$ is also convex. If $f(x)$ is L-smooth, then $f_\mu(x)$ is also L-smooth.

3 Hessian-Aware Zeroth-Order Method

In this section, we will exploit the Hessian information of the model function which was commonly ignored in the past works on zeroth-order optimization and propose `ZO-HessAware` algorithm.

Our algorithm first constructs an approximate Hessian $\tilde{H}$ for the current point x satisfies

$$\rho\tilde{H} \preceq \nabla^2 f(x) \preceq (2-\rho)\cdot\tilde{H}, \text{ and, } \ \zeta\cdot I_d \preceq \tilde{H}, \tag{3.1}$$

with $0 < \rho \le 1$. Parameter ρ measures how well $\tilde{H}$ approximates $\nabla^2 f(x)$. If $\rho = 1$, then $\tilde{H}$ is the exact Hessian. On the other hand, if ρ is small, then $\tilde{H}$ approximates $\nabla^2 f(x)$ poorly. One can use different methods to construct such $\tilde{H}$. In Section 4, we will provide several approaches to compute a good approximate Hessian with a small number of queries to the function value. Note that, we do not need to construct an approximate Hessian for each iteration. Empirically, we can only update it every p iterations where p is a parameter controls the frequency of the Hessian approximation.

Then we begin to estimate the gradient by derivative-free oracles. Different from the existing zeroth-order works (Nesterov & Spokoiny, 2017; Ghadimi & Lan, 2013; Duchi et al., 2015), the Hessian information is used in our gradient estimation represented as follows:

$$g_\mu(x) = \frac{1}{b}\sum_{i=1}^{b}\frac{f(x+\mu K u_i) - f(x)}{\mu}K^{-1}u_i, \text{ with } K = \tilde{H}^{-1/2}, \ u_i \sim N(0, I) \tag{3.2}$$

where b is the batch size. On the point x, we sample $b+1$ points to obtain a good gradient estimation. This strategy is widely used in real applications such as adversarial attack.

Finally, analogue to Newton-style algorithms, we update x_{t+1} using the approximate Hessian and estimated gradient as $x_{t+1} = x_t - \eta\tilde{H}^{-1}g_\mu(x)$. Combining with Eqn. (3.2), we represent the

Algorithm 1 Algorithm `ZO-HessAware`

```
Input: $x^{(0)}$ is an initial point sufficient close to $x^*$. And $b$ is the batch size and $p$ is an integer. Parameter $\eta$ is the step size.
for $t = 0, \dots, T$ do
    if $t \bmod p == 0$ then
        Compute an approximate Hessian $\tilde{H}_t$ satisfies Eqn. (3.1).
    end if
    Generate $b$ samples with $u_i \sim N(0, I_d)$ and construct $\tilde{g}_\mu(x_t) = \frac{1}{b}\sum_{i=1}^{b} \frac{f(x+\tilde{H}_t^{-1/2}u_i)-f(x)}{\mu}\tilde{H}_t^{-1/2}u_i$;
    Update $x_{t+1} = x_t - \eta\tilde{g}_\mu(x_t)$.
end for
```

algorithmic procedure of `ZO-HessAware` as follows

$$\begin{cases} \tilde{g}_\mu(x_t) = & \frac{1}{b}\sum_{i=1}^{b} \frac{f(x_t+\mu\tilde{H}_t^{-1/2}u_i)-f(x_t)}{\mu}\tilde{H}_t^{-1/2}u_i, \text{ with } u_i \sim N(0, I) \\ x_{t+1} = & x_t - \eta\tilde{g}_\mu(x_t). \end{cases}$$

We depict the detailed algorithmic procedure of `ZO-HessAware`in Algorithm 1.

In the rest of this section, we will first give some important properties of the estimated gradient computed as Eqn. (3.2). Then we analyze the local and global convergence property of Algorithm 1, respectively. Finally, the query complexity will be analyzed with power-method based Hessian approximation.

3.1 Properties of Estimated Gradient

Now, we list some important properties of $g_\mu(x)$ defined in Eqn. (3.2) that will be used in our analysis of convergence rate of `ZO-HessAware` in the following lemmas. These lemmas are also of independent interest in zeroth-order algorithm.

Lemma 1. *Let $f(x)$ be L-smooth, then $g_\mu(x)$ defined in Eqn. (3.2) satisfies that*

$$\|\mathbb{E}_u[g_\mu(x)] - \nabla f(x)\|_{K^2}^2 \le \frac{\mu^2}{4}L^2\|K\|^4(d+3)^3.$$

Lemma 2. *Let $f(x)$ be L-smooth, then $\|g_\mu(x)\|_{K^2}^2$ can be bounded as*

$$\mathbb{E}_u\|g_\mu(x)\|_{K^2}^2 \le \frac{\mu^2}{2b}L^2\|K\|^4(d+6)^3 + \frac{2(d+2)}{b}\cdot\|\nabla f(x)\|_{K^2}^2.$$

Now we give the bound of $\mathbb{E}_u\left\|K^2g_\mu(x)\right\|_{K^{-2}}^3$ in the following lemma.

Lemma 3. *Let $f(x)$ be L-smooth, then $g_\mu(x)$ has such a property that*

$$\mathbb{E}_u\left\|K^2g_\mu(x)\right\|_{K^{-2}}^3 \le \frac{1}{b^2}\left(2\mu^3L^3\|K\|^6\cdot(d+9)^{9/2} + 12(d+5)^{3/2}\|\nabla f(x)\|_{K^2}^3\right).$$

3.2 Local Convergence

Now we begin to analyze the local convergence property of Algorithm 1. To achieve a fast convergence rate, the initial point should be close enough to the optimal point. At the same time, the Hessian should be well-approximated.

Theorem 1. *Let $f(x)$ be τ-strongly convex and L-smooth. And $\nabla^2 f(x)$ is γ-Lipschitz continuous. Let the approximate Hessian $\tilde{H}_t$ satisfy Eqn. (3.1). Setting the step size $\eta = \frac{b}{4(d+2)}$, then Algorithm 1 has the following convergence properties:*

$$\mathbb{E}\left[f(x_{t+1}) - f(x^\star)\right] \le \left(1 - \frac{b\rho}{16(d+2)}\right)\left(f(x_t) - f(x^\star)\right) + \Delta_\mu, \tag{3.3}$$

if x_t satisfies that

$$\|x_t - x^\star\| \le \frac{\rho}{\gamma} \cdot \min\left(\frac{3\tau\zeta^{1/2}}{64L^{1/2}}, \frac{d^{3/2}\zeta^2}{17L(d+2)}\right). \tag{3.4}$$

And Δ_μ is defined as

$$\Delta_\mu = b \cdot \left(\frac{\mu^2 L^2}{32\zeta^2}(d+5)^2 + \frac{\mu^2 L^2}{64\zeta}(d+38) + \frac{\gamma\mu^3 L^3}{768\zeta^{9/2}} \cdot (d+110)^{3/2}\right).$$

Remark 1. *Note that, the local convergence properties rely on the condition (3.4). However, this condition may be violated for next iteration if the descent direction is not good. This problem can be remedied by checking the value of $f(x_{t+1})$. We will discard the current x_{t+1} if $f(x_{t+1})$ is larger than $f(x_t)$.*

To achieve an ϵ-accuracy solution, both the first and second terms in the right hand of inequality (3.3) must be smaller than $\epsilon/2$. Therefore, `ZO-HessAware` needs

$$N(\epsilon) = O\left(\frac{d}{b\rho}\log\left(\frac{1}{\epsilon}\right)\right) = O\left(\frac{d\lambda_{k+1}}{b\tau}\log\left(\frac{1}{\epsilon}\right)\right)$$

iterations. In contrast, without the second-order information, first order methods with zeroth-order oracles need $O\left(db^{-1}\kappa\log\left(\frac{1}{\epsilon}\right)\right)$ iterations where $\kappa = L/\tau$ is the condition number. Since it holds that $\lambda_{k+1} \le L$, `ZO-HessAware` has a faster convergence rate than conventional zeroth-order methods without Hessian information. Especially when the Hessian can be well approximated by a rank-k matrix, that is $\lambda_{k+1} \ll L$, our algorithm will show great advantages.

3.3 Global Convergence

We will analyze the global convergence property of Algorithm 1 in this section. To guarantee a global convergence, we have to set a smaller step size compared with the one set in Theorem 1. Then, we have the following theorem.

Algorithm 2 Power-method Based Hessian Approximation.

```
Input: Orthonormal matrix $V_0 \in \mathbb{R}^{d\times k}$ where $k$ is the target rank;
for $t = 0, \dots, T-1$ do
    Approximate the $\nabla^2 f(x)V_t$ by $Y_t = H_\mu V_t$ implemented as Eqn. (3.5);
    QR factorization: $Y_t = V_{t+1}R_{t+1}$, where $V_{t+1}$ consists of orthonormal columns.
end for
Compute $Y = H_\mu V_T$ and compute the SVD decomposition $Y = \hat{U}\Lambda\hat{V}^\top$.
Return: $\tilde{H} = V\Lambda V^\top + 5\lambda_{k+1}I_d$ with $V = V_T\hat{V}$
```

Theorem 2. *Let function $f(x)$ satisfy the properties described in Theorem 1. For each iteration, the approximate Hessian $\tilde{H}_t$ satisfies Eqn. (3.1). By choosing the step size $\eta = \frac{\zeta}{4(d+2)L}$, Algorithm 1 has the following convergence property*

$$\mathbb{E}_u\left[f(x_{t+1}) - f(x^\star)\right] \le \left(1 - \frac{b\zeta}{16(d+2)\kappa L}\right)\cdot\left(f(x_t) - f(x^\star)\right) + \Delta_\mu,$$

where Δ_μ is defined as

$$\Delta_\mu = \frac{b\mu^2 L}{64\zeta(d+2)}\left(2(d+3)^3 + \frac{(d+6)^3}{d+2}\right).$$

In our analysis of global convergence, we set a fixed step size. We can also use the line search method to get a better convergence property at the cost of extra query to the function value.

3.4 Query Complexity Analysis

In this section, we will analyze the query complexity of `ZOHA-PW` which implements `ZO-HessAware` with power-method based Hessian approximation. The power method only needs to access Hessian-Vector product which can be approximated by

$$[\nabla^2 f(x)v]_i \approx [H_\mu v]_i \triangleq \frac{f(x+\mu_1\cdot(v+e_i)) - f(x+\mu_1\cdot(v-e_i)) + f(x-\mu_1\cdot e_i) - f(x+\mu_1\cdot e_i)}{2\mu_1^2}, \qquad (3.5)$$

where $[H_\mu v]_i$ means the i-th entry of vector $H_\mu v$. Note that, H_μ does not need to be explicitly represented. And it can be regarded as the Hessian $\nabla^2 f(x)$ with some small perturbations.

Given the above results to approximating Hessian-vector product, we conduct power method to obtain the k-largest eigenvalue and their corresponding eigenvectors. The detailed algorithmic procedure is depicted in Algorithm 2. Then the approximate Hessian $\tilde{H}$ computed based on power method has the following properties.

Theorem 3. *Let the objective function satisfy Eqn. (2.4). Let μ_1 and the iteration number T satisfy that*

$$\mu_1 \le \min\left\{\frac{C_1}{4k\gamma}\lambda_{k+1}, \frac{C_2}{4}\cos(U_k, V_0)\right\} \text{ and } T = 2C_3\log\left(2\tan(U_k, V_0)\right)$$

where U_k is the matrix consists of eigenvectors corresponding to the first k largest eigenvalues of

$\nabla^2 f(x)$. C_1, C_2, and C_3 are absolute constants. Then $\tilde{H}$ returned from Algorithm 2 has the following property

$$\frac{\lambda_{\min}}{\lambda_{\min} + 10\lambda_{k+1}} \tilde{H} \preceq \nabla^2 f(x) \preceq \tilde{H}, \text{ and, } \; 5\lambda_{k+1} I_d \preceq \tilde{H}.$$

Now we give the query complexity analysis of `ZOHA-PW`. First, we choose the batch size $b = O(dk)$ and the parameter $p = 1$. Then, for each iteration of `ZOHA-PW`, it takes $O(dk)$ queries to estimate gradient, and $\tilde{O}(dk)$ queries to construct the approximate Hessian. Combining the convergence rate depicted in Theorem 1, we have the following query complexity of `ZOHA-PW`.

Theorem 4. *Set the $b = O(dk)$ and $p = 1$ in Algorithm 1. Then the query complexity of `ZOHA-PW` is*

$$Q(\epsilon) = \tilde{O}\left(\frac{d\lambda_{k+1}}{\tau} \cdot \log\left(\frac{1}{\epsilon}\right)\right).$$

The approximate Hessian $\tilde{H}$ constructed using power method can capture the principal rank-k information of $\nabla^2 f(x)$. We have the empirical fact that the Hessian of model function can be written as a rank-k matrix and a perturbation matrix of small norm and its main information lies in the rank-k matrix (Yuan et al., 2007; Bakker et al., 2018; Sainath et al., 2013), that is $\lambda_{k+1} \ll L$. Hence the query complexity of `ZOHA-PW` is much smaller than the one of zeroth-order methods without Hessian information which takes $O\left(\frac{dL}{\tau} \cdot \log\left(\frac{1}{\epsilon}\right)\right)$ indicated in the work of Nesterov & Spokoiny (2017).

Furthermore, we can use V_T of the last iteration of Algorithm 1 as the input V_0 of Algorithm 2. Because x_t is close to the optimal point $x^\star$, the value of $\tan(U_k, V_0)$ can be regarded as a constant, that is, we can obtain an approximate Hessian in $O(dk)$ query complexity. Hence, the query complexity of `ZOHA-PW` can be further improved to $O\left(\frac{d\lambda_{k+1}}{\tau} \cdot \log\left(\frac{1}{\epsilon}\right)\right)$.

4 Structured Hessian Approximation

In this section, we will provide two kinds of *heuristic* methods to construct approximate Hessian. These methods takes much fewer queries to function value compared with power-method based Hessian approximation which takes at least $O(dk)$ queries. The first method is based on Gauss sampling and the second one is based on diagonalization.

4.1 Gaussian-Sampling Based Hessian Approximation

In this section, we propose a novel method to approximate the Hessian of $f(x)$ with a much lower query complexity. This method is based on Gaussian Sampling, and we name `ZO-HessAware` implemented with such Hessian Approximation as `ZOHA-Gauss`.

Our new method is going to estimate the Hessian of $f_\mu(x)$ defined in Eqn. (2.6). Since $\nabla^2 f_\mu(x)$ is close to $\nabla^2 f(x)$ if μ is small, a good approximation of $\nabla^2 f_\mu(x)$ will approximate $\nabla^2 f(x)$ well. In fact, we can bound the error between $\nabla^2 f(x)$ and $\nabla^2 f_\mu(x)$ as follows.

Lemma 4. *Let $f_\mu(x)$ be defined in Eqn. (2.6). The objective function $f(x)$ satisfies Eqn. (2.4). Then, we have*

$$\left\|\nabla^2 f_\mu(x) - \nabla^2 f(x)\right\| \le \gamma\mu(d+1)^{1/2}.$$

Using Gaussian sampling, we can approximate the Hessian of $f_\mu(x)$ as follows:

$$\tilde{H} = b^{-1}\sum_{i=1}^{b}\frac{f(x+\mu u_i)+f(x-\mu u_i)-2f(x)}{2\mu^2}u_i u_i^\top + \lambda I_d, \text{ with } u_i \sim N(0, I_d) \tag{4.1}$$

where λ is a properly chosen regularizer to keep $\tilde{H}$ invertible. In the construction of $\tilde{H}$ of Eqn. (4.1), we only take a small batch of points, that is b is small even much smaller the dimension d. Hence, the construction of such $\tilde{H}$ has a low query complexity.

The approximate Hessian $\tilde{H}$ constructed as Eqn. (4.1) has the following property.

Lemma 5. *Let $\tilde{H}$ be an approximate Hessian defined in Eqn. (4.1). Function $f_\mu(x)$ is the smoothed function defined in Eqn. (2.6). Then $\tilde{H}$ has the following property*

$$\nabla^2 f_\mu(x) \preceq \mathbb{E}_u[\tilde{H}] = \nabla^2 f_\mu(x) + \left(\lambda - \frac{f(x)-f_\mu(x)}{\mu^2}\right)\cdot I_d$$

Combining Lemma 4 and 5, we can obtain the result that if μ is small, and the batch size b in Eqn. 4.1 is large, then $\tilde{H}$ constructed as Eqn. (4.1) can approximate $\nabla^2 f(x)$ very well. However, we can not give the exact approximation precision of such $\tilde{H}$ measured by ρ in Eqn. (3.1) when the bash size b is much smaller than d. Thus, we will not give the theoretical convergence rate and query complexity of `ZOHA-Gauss`.

4.2 Diagonalization Based Hessian Approximation

We propose to use a diagonal matrix to approximate the Hessian. This method has been used in the optimization of deep neural networks (Kingma & Ba, 2015; Zeiler, 2012; Tieleman & Hinton, 2012) and online learning (Duchi et al., 2011).

First, we compute an approximate Hessian in the manner of `ADAM` (Kingma & Ba, 2015) as follows:

$$\begin{aligned}
\tilde{g}_\mu(x_{t-1}) =& \frac{1}{b}\sum_{i=1}^{b}\frac{f(x_{t-1}+\mu\tilde{u}_i)-f(x_{t-1})}{\mu}\tilde{u}_i, \text{ with } \tilde{u}_i \sim N(0, \tilde{H}_{t-1}^{-1})\\
D_t =& \nu D_{t-1} + (1-\nu)\tilde{g}_\mu^2(x_{t-1})\\
\tilde{H}_t =& \mathrm{diag}\left(\frac{D_t}{1-\nu^t}\right)
\end{aligned} \tag{4.2}$$

with $0 \le \nu \le 1$. And $\tilde{g}_\mu^2(x)$ means the entry-wise square of $\tilde{g}_\mu(x)$.

Second, we can also use the method of `ADAGRAD` (Duchi et al., 2011) to construct the approximate

Hessian as

$$\begin{aligned}\tilde{g}_\mu(x_{t-1}) =& \frac{1}{b}\sum_{i=1}^{b}\frac{f(x_{t-1}+\mu\tilde{u}_i)-f(x_{t-1})}{\mu}\tilde{u}_i, \text{ with } \tilde{u}_i \sim N(0, \tilde{H}_{t-1}^{-1})\\ D_t =& D_{t-1} + \tilde{g}_\mu^2(x_{t-1})\\ \tilde{H}_t =& \text{diag}\left(\frac{D_t}{n}\right).\end{aligned}$$

Other methods of constructing diagonal Hessian approximation such as `ADADELTA` (Zeiler, 2012) used in training deep neural networks can also be use to in our diagonal Hessian approximation.

These kinds of Hessian approximation are heuristic. We can not give an exact convergence rate of `ZO-HessAware` with diagonal Hessian approximation by Theorem 1. However, diagonal Hessian approximations have shown their power in training deep neural networks. Furthermore, diagonal approximate Hessian has an important advantage that it does not need extra queries to the function value and need less computational and storage cost.

Though the construction procedure of the diagonal Hessian approximation is the same with the one of `ADAM` and `ADAGRAD`, there some difference between these diagonal Hessians. First, `ADAM` and `ADAGRAD` use $\tilde{H}^{1/2}$ as the approximate Hessian in training neural network which is different from our approximate Hessian. Second, in the construction of our diagonal Hessian, we use the 'natural gradient' defined in Eqn. (1.3) which contains the Hessian information other than the ordinary gradient. And the information of the current diagonal Hessian will be used in the estimation of next 'natural gradient'. In contrast, the diagonal Hessian will not affect the computation of gradients.

5 Experiments

In this section, we apply our Hessian-aware zeroth-order algorithm to the black-box adversarial attacks. This is an important research topic in security of deep learning because neural networks are widely used in image classification. However, current neural network-based classifiers are susceptible to adversarial examples.

Our adversarial attack experiments include both targeted attack and un-targeted attack. The targeted attack task aims to find an adversarial example x of a given image x_0 with a targeted class label ℓ toward misclassification. In this case, we are going to minimize the following problem proposed in the work of Carlini & Wagner (2017):

$$f(x, \ell) = \max\{\max_{i\neq\ell}[Z(x)]_i - [Z(x)]_\ell, -\omega\}, \text{ with } \|x - x_0\|_\infty \leq \varepsilon, \tag{5.1}$$

where $Z(x)$ is the logit layer representation (logits) in the DNN for x such that $[Z(x)]_i$ represents the predicted probability that x belongs to class i. Parameter $\omega \geq 0$ is a tuning parameter for attack transferability and we set it $\omega = 1$ in our experiments. The constrain means that the adversarial image should be close to the given image.

The un-targeted adversarial attack task aims to find an example x of the given image x_0 with label ℓ but misclassified by the neural network. In this case, we will minimize the following function (Carlini & Wagner, 2017):

$$f(x) = \max([Z(x)]_\ell - \max_{i \neq \ell}[Z(x)]_i, -\omega), \text{ with } \|x - x_0\|_\infty \leq \varepsilon. \tag{5.2}$$

5.1 Algorithm Implementation

In the experiments, we will implement `ZO-HessAware` (Algorithm 1) with two different kinds of Hessian approximation. The first one is based on the Gaussian sampling described in Section 4.1, and we call it `ZOHA-Gauss`. The second implementation is using the diagonal Hessian approximation described in Section 4.2 with the update procedure as `ADAM` defined in Eqn. (4.2). And we name it as `ZOHA-Diag`. We do not implement `ZO-HessAware` with other kinds of diagonal Hessian approximation because these methods have the similar performance.

Furthermore, because the adversarial problem is of constrain, we will modify the update step (7) of Algorithm 1 as follows:

$$x_{t+1} = \Pi[x_t - \eta \tilde{g}_\mu(x_t)], \tag{5.3}$$

where $\Pi[\cdot]$ is a projection operator to make x_{t+1} satisfy $\|x_{t+1} - x_0\|_\infty \leq \varepsilon$. Note that, this projection is exact for `ZOHA-Diag`. But as to `ZOHA-Gauss`, we should compute x_{t+1} by optimizing the following sub-problem:

$$x_{t+1} = \operatorname*{argmin}_{y \in [x_0 - \epsilon, x_0 + \epsilon]} \left\| y - \left(x_t - \eta \tilde{H}_t^{-1} g_\mu(x_t) \right) \right\|_{\tilde{H}}^2. \tag{5.4}$$

However, the projection as Eqn. (5.3) performs well and is of simple implementation even it is just an approximation to the true one computed by Eqn. (5.4).

Because the objective function of the neural network model may be non-convex, we implement the approximate Hessian in `ZOHA-Gauss` as follows:

$$\tilde{H} = b^{-1} \sum_{i=1}^{b} \frac{|f(x + \mu u_i) + f(x - \mu u_i) - 2f(x)|}{2\mu^2} u_i u_i^\top + \lambda I_d, \text{ with } u_i \sim N(0, I_d).$$

Such modification ensures that such $\tilde{H}$ is positive definite. Furthermore, we observe that $\tilde{H}$ can be written as $\tilde{H} = CC^\top + \lambda I$. Then we can compute $\tilde{H}^{-1/2}$ as follows. First, we compute the SVD decomposition of C as $C = U_C \Lambda_C U_C^\top$ with $U_C \in \mathbb{R}^{d \times b}$ and $\Lambda_C \in \mathbb{R}^{b \times b}$. And we get $\tilde{H}^{-1/2} = U_C \left((\Lambda_C^2 + \lambda I)^{-1/2} - \lambda^{-1/2} I \right) U_C^\top + \lambda^{-1/2} I$. In practice, the value of λ can be set as a fraction of $\|CC^\top\|$ or tuned by several tries.

Algorithm 3 Algorithm `ZO-HessAware` with descent checking (`ZOHA-DC`)

```
**Input:** $x^{(0)}$ is an initial point sufficient close to $x^*$. And $b$ is the batch size and $p$ is an integer. Parameter $\eta$ is the step size. $\beta$ is the threshold of sample size in descent checking. $\delta_b$ is the parameter of the sample size increment.
for $t = 0, \dots, T$ do
  if $t$ mod $p == 0$ then
    Compute an approximate Hessian $\tilde{H}_t$ satisfies Eqn. (3.1).
  end if
  Generate $b$ samples with $u_i \sim N(0, I_d)$ and construct $\tilde{g}_\mu(x_t) = \frac{1}{b}\sum_{i=1}^{b} \frac{f(x+\tilde{H}_t^{-1/2}u_i)-f(x)}{\mu}\tilde{H}_t^{-1/2}u_i$;
  Compute $y_{t+1} = x_t - \eta\tilde{g}_\mu(x_t)$ and set $N_b = b$.
  while $f(y_{t+1}) > f(x_t)$ and $N_b < \beta$ do
    Generate another $\delta_b$ samples with $u_i \sim N(0, I_d)$ and set $N_b = N_b + \delta_b$;
    Construct $\tilde{g}_\mu(x_t) = \frac{1}{N_b}\sum_{i=1}^{N_b} \frac{f(x+\tilde{H}_t^{-1/2}u_i)-f(x)}{\mu}\tilde{H}_t^{-1/2}u_i$ ;
    Compute $y_{t+1} = x_t - \eta\tilde{g}_\mu(x_t)$.
  end while
  Update $x_{t+1} = x_t - \eta\tilde{g}_\mu(x_t)$.
end for
```

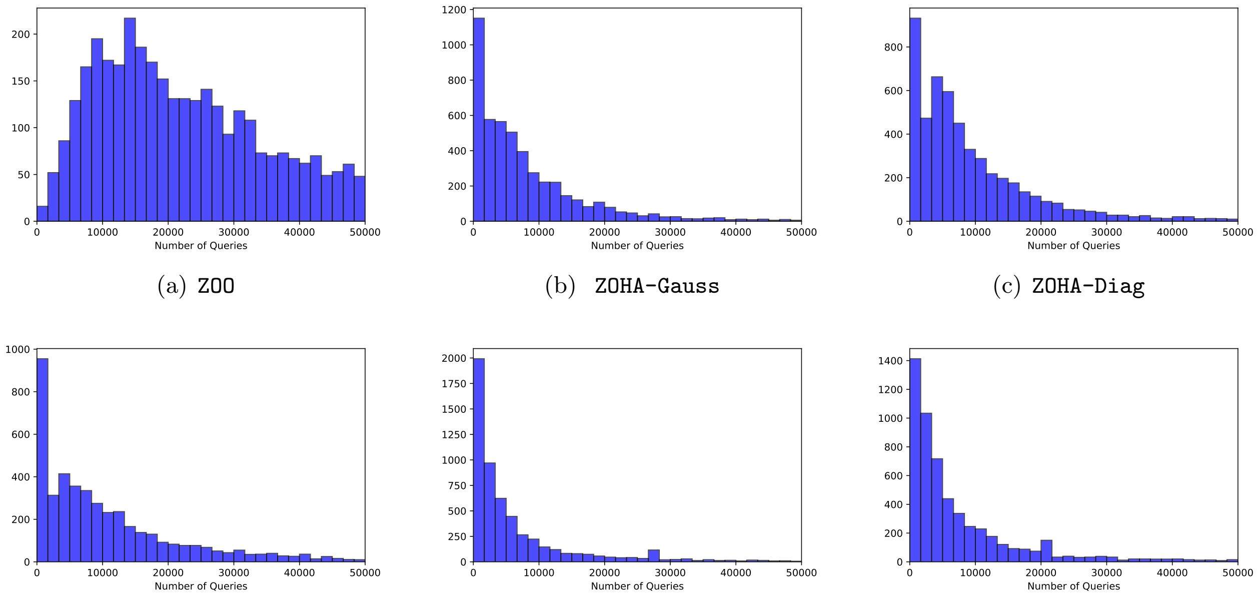
5.1.1 Descent Checking

To improve the success rate and the query efficiency, we introduce an important technique called `Descent-Checking` which has been discussed in Remark 1 but with a slight different implementation. `Descent-Checking` has the following algorithmic procedure. After obtaining the x_{t+1}, we will query to the value $f(x_{t+1})$ and check if $f(x_{t+1}) \le f(x_t)$. If it holds, we will go to the next iteration. Otherwise, we will discard current x_{t+1} and take extra δ_b samples combining with existing samples to estimate a new gradient until a new x_{t+1} satisfies $f(x_{t+1}) \le f(x_t)$ or the total sample size exceeds a threshold. If the total sample size exceeds the threshold, we will accept this 'bad' x_{t+1} and go to the next iteration. Because we will often set b be of several tens, `Descent-Checking` strategy will not bring many extra queries. As a result of `Descent-Checking`, we can filter some bad search direction effectively which will lead to a higher attack success rate. We depict the detailed algorithmic procedure of `ZO-HessAware` with `Descent-Checking` in Algorithm 3. Accordingly, we name `ZOHA-Gauss` and `ZOHA-Diag` with `Descent-Checking` strategy as `ZOHA-Gauss-DC` and `ZOHA-Diag-DC`, respectively.

5.2 Evalution on MNIST

We evaluate the effectiveness of our attacks against an convolution neural network (CNN) on the MNIST dataset. The network for MNIST is composed of two 5×5 convolutional layers with output 16 and 64 channels following two fully connected layers with 128 and 10 units. We use 2×2 max-pooling after each convolutional layer and use ReLU after every layer expect the layer. The network is trained for 100 epochs with learning rate starting at 0.1 and decay 0.5 every 20 epochs. The accuracy of the model is 98.95%.

We test the attack algorithms on 10000 images from the test set. The limit of ℓ_∞ perturbation is $\varepsilon = 0.2$. We run all the attack until getting the an adversarial examples unless the number of



(a) ZOO (b) ZOHA-Gauss (c) ZOHA-Diag

(d) PGD-NES (e) ZOHA-Gauss-DC (f) ZOHA-Diag-DC

Figure 1: The distribution of the number of queries on *targeted* black-box attacks on CNN model and MNIST

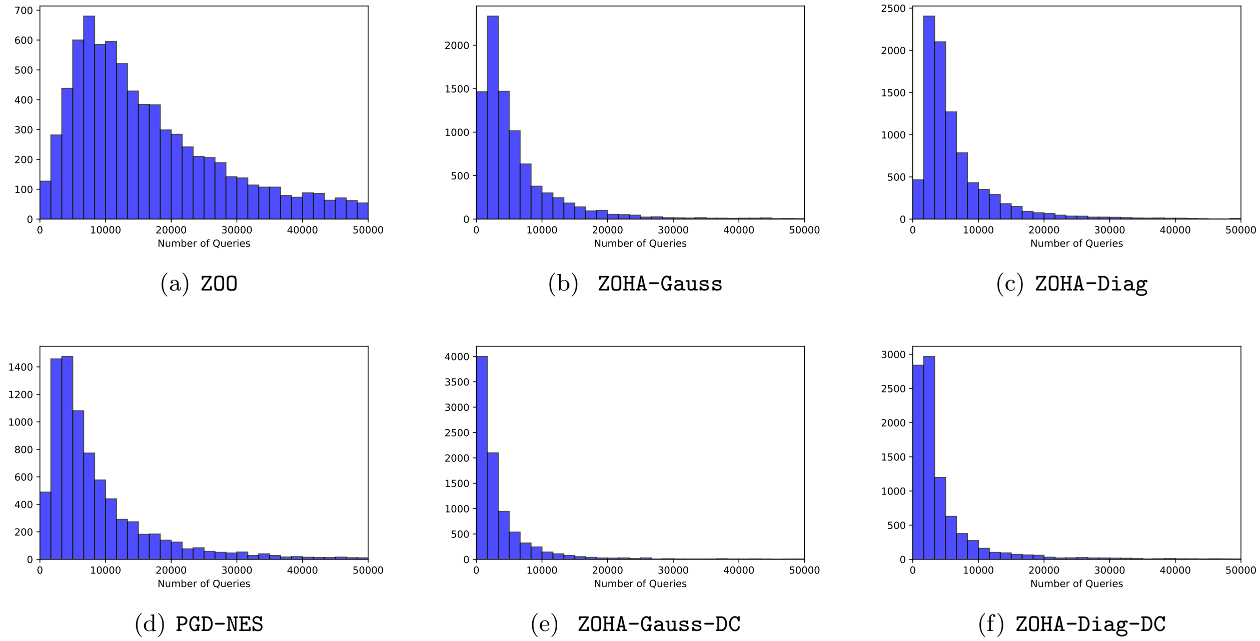
queries is more than $50,000$.

We report the experiment results in Table 1 and Figure 1 and 2. The visualization of adversarial attack is present in Appendix E. We can observe that ZO-HessAware with different implementations obtain much better success rates than two state-of-the-art algorithms. This validates the effectiveness of second-order information of model function in zeroth-order optimization. Especially, our ZOHA-DC type algorithms obtain the best success rates both target and un-target adversarial attacks which are much higher than ZOO and PGD-NES while taking less queries to function values. Furthermore, on the un-target attack, ZOHA-DC type algorithms only take less than half of queries of PGD-NES. This greatly shows the query efficiency of our Hessian-aware zeroth-order algorithm.

The comparison of the distribution of the query number in Figure 1 and 2 shows that Descent Checking technique can reduce the query number effectively. For example, comparing ZOHA-Gauss with ZOHA-Gauss-DC in Figure 2, we can observe that the percentage of the query number between 0 and 2000 of ZOHA-Gauss-DC is much higher than the one of ZOHA-Gauss.

5.3 Evaluation on ImageNet

In this experiment, we use a pre-trained ResNet50 that has 78.15% top-1 accuracy and 92.87% top-5 accuracy for evaluation. The limit of ℓ_∞ perturbation is $\varepsilon = 0.05$. We will choose $1,000$ images randomly from ImageNet test-set for evaluation and run the attack method until getting an adversarial example or the number of queries being more than $1,000,000$. Furthermore, if the



(a) `ZOO` (b) `ZOHA-Gauss` (c) `ZOHA-Diag`

(d) `PGD-NES` (e) `ZOHA-Gauss-DC` (f) `ZOHA-Diag-DC`

Figure 2: The distribution of the number of queries on *un-targeted* black-box attacks on CNN model and MNIST

attack is targeted, the target label will be randomly chosen from $1,000$ classes.

In the experiment on ImageNet, instead of Eqn. (3.2), we use the following method to estimate the gradient

$$g_\mu(x) = \frac{1}{b}\sum_{i=1}^{b} \frac{f(x+\mu\tilde{H}^{-1/2}u_i) - f(x-\mu\tilde{H}^{-1/2}u_i)}{2\mu}\tilde{H}^{1/2}u_i.$$

We can see that such g_μ have the same expectation with the one defined in Eqn. (3.2). However, it has a better performance in this experiment. Accordingly, the natural gradient $\tilde{g}_\mu(x)$ is modified similarly as

$$\tilde{g}_\mu(x) = \frac{1}{b}\sum_{i=1}^{b} \frac{f(x+\mu\tilde{u}_i) - f(x-\mu\tilde{u}_i)}{2\mu}\tilde{u}_i, \text{ with } \tilde{u}_i \sim N(0,\tilde{H}^{-1}).$$

We report the results in Table 2 and Figure 3, 4. The visualization of adversarial attack is present in Figure 6 and 7 of Appendix E. We can observe that our algorithms take much less queries than `ZOO` and `PGD-NES`. For the un-target attack, the median queries of `ZOHA-Diag-DC` is only about 5% of `ZOO` and about 38% of `PGD-NES` with the same success rate. For the target attack, compared with the un-target attack problem, all these algorithms take much more queries. But our algorithms still show great query efficiency. Especially, both `ZOHA-Diag` and `ZOHA-Diag-DC`

Table 1: Comparison of ℓ_∞ norm based black-box attacks on CNN model and MNIST with $\varepsilon = 0.2$

	Algorithm	success rate %	median queries	average queries
targeted	ZOO (Chen et al., 2017)	42.13	15,200	17,091
	PGD-NES (Ilyas et al., 2018)	44.19	7,300	10,496
	ZOHA-Gauss	50.03	3,712	6,649
	ZOHA-Gauss-DC	**56.14**	**2,941**	**6,246**
	ZOHA-Diag	52.13	6,400	9,128
	ZOHA-Diag-DC	55.56	3,936	7,239
un-targeted	ZOO (Chen et al., 2017)	77.18	13,300	16,390
	PGD-NES (Ilyas et al., 2018)	81.55	5,800	8,567
	ZOHA-Gauss	85.06	3,612	5,000
	ZOHA-Gauss-DC	88.80	**2,152**	**3,629**
	ZOHA-Diag	90.37	4,500	6,439
	ZOHA-Diag-DC	**91.90**	2,460	4,352

achieve 100% attack success rate which is higher than PGD-NES but only with about 50% queries of PGD-NES. Though ZOO also obtain a 100% success rate, it takes several times of queries as ZOHA-Diag and ZOHA-Diag-DC.

5.4 Discussion

From above two experiments, we can find some important insights. First, the comparison between attack success rates of two deep learning models indicates that a deeper or more complicate neural network potentially involves more vulnerability to the adversarial attack. On the ResNet50, all algorithms achieve success rates over 99%. In contrast, the attack success rate on the simple convolution network with several layers is much lower.

Second, the great gap of success rates between our Hessian-aware zeroth-order methods and two state-of-the-art algorithms on the MNIST may reveal such a potential that the Hessian information will bring great advantages on the hard adversarial attack problem.

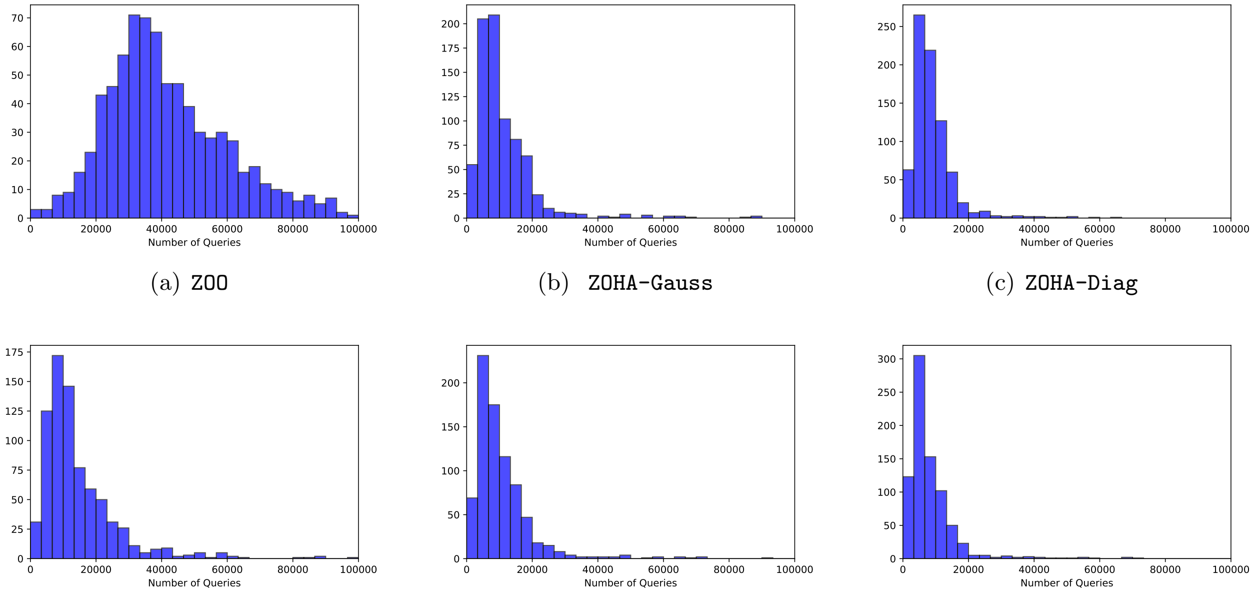
Third, we can observe that our algorithms with Descent-Checking have better performance than the ones without Descent-Checking. The experiment results on the MNIST show that Descent-Checking strategy can promote attack success rate effectively both for targeted and untargeted attack. At the same time, Descent-Checking is an effective way to reduce query complexity. This can be easily observed from the distribution of the number of queries in Figure 1-4.

6 Conclusion

In this paper, we propose a novel zeroth-order algorithmic framework called ZO-HessAware which exploits the second-order information of the model function. Due to this information, ZO-HessAware achieves a faster convergence rate and lower query complexity than existing works without the Hessian information. We also propose several methods to capture the principal information of the Hessian efficiently. Experiments on the black-box adversarial attack show that

Table 2: Comparison of ℓ_∞ norm based black-box attacks on ResNet50 model and ImageNet with $\varepsilon = 0.05$

	Algorithm	success rate %	median queries	average queries
targeted	ZOO (Chen et al., 2017)	**100**	39,100	45,822
	PGD-NES (Ilyas et al., 2018)	99.37	11,270	17,435
	ZOHA-Gauss	99.62	8,748	12,257
	ZOHA-Gauss-DC	100	8,588	11,770
	ZOHA-Diag	**100**	7,400	9,123
	ZOHA-Diag-DC	**100**	**6,273**	**8,574**
un-targeted	ZOO (Chen et al., 2017)	100	12,700	14,199
	PGD-NES (Ilyas et al., 2018)	100	1,500	2,283
	ZOHA-Gauss	100	1,212	2,259
	ZOHA-Gauss-DC	100	1,124	1,959
	ZOHA-Diag	100	800	1,149
	ZOHA-Diag-DC	100	**561**	**945**



(a) ZOO

(b) ZOHA-Gauss

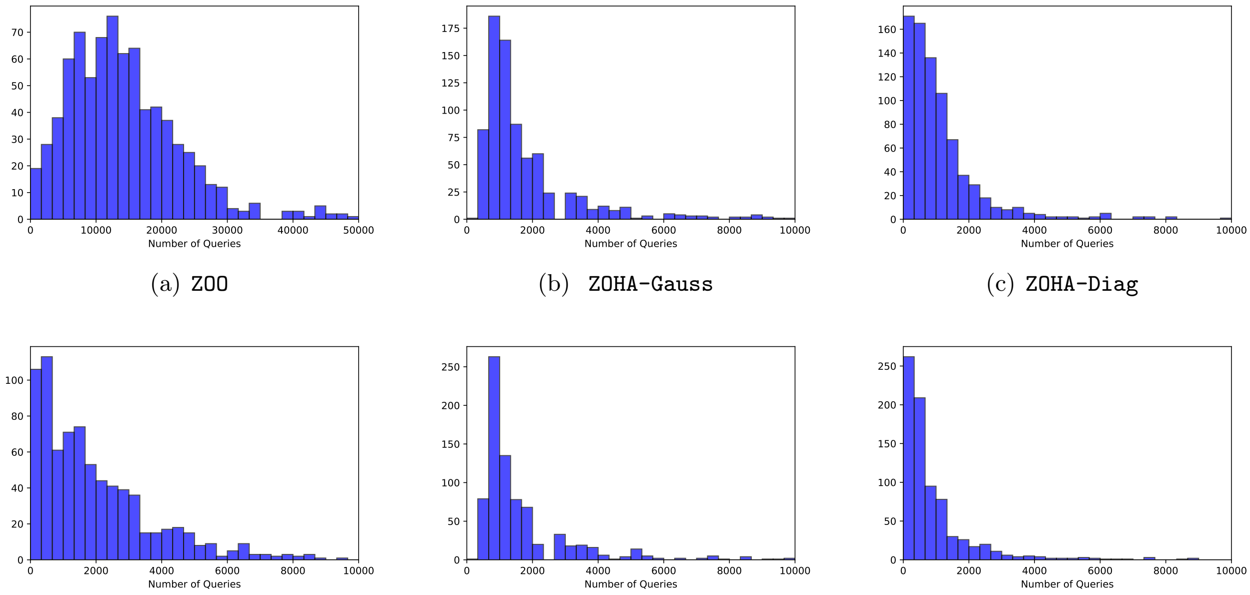
(c) ZOHA-Diag

(d) PGD-NES

(e) ZOHA-Gauss-DC

(f) ZOHA-Diag-DC

Figure 3: The distribution of the number of queries on *targeted* black-box attacks on ResNet50 model and ImageNet



(a) `ZOO` (b) `ZOHA-Gauss` (c) `ZOHA-Diag`

(d) `PGD-NES` (e) `ZOHA-Gauss-DC` (f) `ZOHA-Diag-DC`

Figure 4: The distribution of the number of queries on *un-targeted* black-box attacks on ResNet50 model and ImageNet

our `ZO-HessAware` algorithms improve the attack success rate and reduce the query complexity effectively. This validates the effectiveness of Hessian information in zeroth-order optimization and our theoretical analysis empirically. We also propose a novel technique called `Descent Checking` which can promote attack success rate and reduce query complexity empirically.

References

Bach, F. & Perchet, V. (2016). Highly-smooth zero-th order online optimization. In *Conference on Learning Theory* (pp. 257–283).

Bakker, C., Henry, M. J., & Hodas, N. O. (2018). Understanding and exploiting the low-rank structure of deep networks.

Balcan, M.-F., Du, S. S., Wang, Y., & Yu, A. W. (2016). An improved gap-dependency analysis of the noisy power method. In *Conference on Learning Theory* (pp. 284–309).

Bergstra, J. S., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems* (pp. 2546–2554).

Carlini, N. & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)* (pp. 39–57).: IEEE.

Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., & Hsieh, C.-J. (2017). 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* (pp. 15–26).: ACM.

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul), 2121–2159.

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

Fang, C., Li, C. J., Lin, Z., & Zhang, T. (2018). Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems* (pp. 686–696).

Ghadimi, S. & Lan, G. (2013). Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4), 2341–2368.

Hansen, N. & Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2), 159–195.

Hu, W. & Tan, Y. (2017). Generating adversarial malware examples for black-box attacks based on gan. *arXiv preprint arXiv:1702.05983*.

Ilyas, A., Engstrom, L., Athalye, A., & Lin, J. (2018). Black-box adversarial attacks with limited queries and information. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research* (pp. 2137–2146). Stockholmsmässan, Stockholm Sweden: PMLR.

Kingma, D. P. & Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR).*

Liu, S., Kailkhura, B., Chen, P.-Y., Ting, P., Chang, S., & Amini, L. (2018). Zeroth-order stochastic variance reduction for nonconvex optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 31* (pp. 3731–3741). Curran Associates, Inc.

Magnus, J. R. (1978). The moments of products of quadratic forms in normal variables. *Statistica Neerlandica*, 32(4), 201–210.

Matyas, J. (1965). Random optimization. *Automation and Remote control*, 26(2), 246–253.

Nesterov, Y. & Spokoiny, V. (2017). Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2), 527–566.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., & Swami, A. (2017). Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security* (pp. 506–519).: ACM.

Sainath, T. N., Kingsbury, B., Sindhwani, V., Arisoy, E., & Ramabhadran, B. (2013). Low-rank matrix factorization for deep neural network training with high-dimensional output targets. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on* (pp. 6655–6659).: IEEE.

Salimans, T., Ho, J., Chen, X., Sidor, S., & Sutskever, I. (2017). Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*.

Shamir, O. (2017). An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18(52), 1–11.

Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems* (pp. 2951–2959).

Tieleman, T. & Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2), 26–31.

Wierstra, D., Schaul, T., Glasmachers, T., Sun, Y., Peters, J., & Schmidhuber, J. (2014). Natural evolution strategies. *The Journal of Machine Learning Research*, 15(1), 949–980.

Yuan, M., Ekici, A., Lu, Z., & Monteiro, R. (2007). Dimension reduction and coefficient estimation in multivariate linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(3), 329–346.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

A Proof of Section 3.1

In this section, we will prove three properties of the estimated gradient defined in Eqn. (3.2). Before that, we first list some important lemmas related to the Gaussian distribution that will be used in our proof.

A.1 Important Lemmas

Lemma 6 ((Nesterov & Spokoiny, 2017)). *If $f(x)$ is L-smooth, then we have*

$$\|\nabla f_\mu(x) - \nabla f_\mu(x)\| \le L\,\|x - y\|\,.$$

Lemma 7 ((Nesterov & Spokoiny, 2017)). *If $f(x)$ is L-smooth, then*

$$\|\nabla f_\mu(x) - \nabla f(x)\| \le \frac{\mu}{2} L (d+3)^{3/2}.$$

If the Hessian is γ-Lipschitz continuous, then we can guarantee that

$$\left\|\nabla^2 f_\mu(x) - \nabla^2 f(x)\right\| \le \frac{\mu^2}{6}\gamma(d+4)^2.$$

Lemma 8 ((Nesterov & Spokoiny, 2017)). *Let $p \ge 2$, u be from $N(0, I_d)$, then we have the following bound*

$$d^{p/2} \le \mathbb{E}_u\left[\|u\|^p\right] \le (p+d)^{p/2}.$$

Then we give the results of moments of products quadratic forms in normal distribution.

Lemma 9 ((Magnus, 1978)). *Let A and B be two symmetric matrices, and u has the Gaussian distribution, that is, $u \sim N(0, I_d)$. Define $z = u^\top A u \cdot u^\top B u$. The expectation of z and z^2 are:*

$$\begin{aligned}
\mathbb{E}_u(z) =& (\mathrm{tr}\ A)(\mathrm{tr}\ B) + 2(\mathrm{tr}\ AB)\\
\mathbb{E}_u(z^2) =& (\mathrm{tr}\ A)^2(\mathrm{tr}\ B)^2 + 16\left[(\mathrm{tr}\ A)(\mathrm{tr}\ AB^2) + (\mathrm{tr}\ B)(\mathrm{tr}\ A^2B)\right]\\
&+ 2\left[(\mathrm{tr}\ A)^2(\mathrm{tr}\ B)^2 + 4(\mathrm{tr}\ A)(\mathrm{tr}\ B)(\mathrm{tr}\ AB) + (\mathrm{tr}\ B)^2(\mathrm{tr}\ A^2)\right]\\
&+ 4\left[(\mathrm{tr}\ A^2)(\mathrm{tr}\ B^2) + 2(\mathrm{tr}\ AB)^2\right] + 16\left[(\mathrm{tr}\ AB)^2 + 2(\mathrm{tr}\ A^2B^2)\right].
\end{aligned}$$

A.2 Proof of Lemma 1

Proof of Lemma 1. First, the gradient of $f_\mu(x)$ can be represented as (Nesterov & Spokoiny, 2017):

$$\begin{aligned}
\nabla f_\mu(x) =& \frac{1}{M}\int_{\mathbb{R}^d} \frac{f(x+\mu u) - f(x)}{\mu} u\ \exp\left(-\frac{\|u\|^2}{2}\right) du\\
=& \mathbb{E}_u\left[\frac{f(x+\mu u) - f(x)}{\mu} u\right] \qquad \text{(A.1)}
\end{aligned}$$

Let us denote

$$h(y) \triangleq f(x + Ky).$$

Then we have

$$f(x) = h(0).$$

By Eqn. (A.1), we have

$$\mathbb{E}_u \frac{f(x + \mu K u) - f(x)}{\mu} u = \mathbb{E}_u \frac{h(0 + \mu u) - h(0)}{\mu} u = \nabla h_\mu(0).$$

And we also have

$$\nabla h(0) = K \cdot \nabla f(x)$$

By Lemma 7, we can obtain that

$$\|\nabla h_\mu(0) - \nabla h(0)\| \leq \frac{\mu}{2} L(h_\mu)(d+3)^{3/2}. \tag{A.2}$$

where $L(h_\mu)$ is the smoothness parameter of $h_\mu(y)$. By Lemma 6, we know that $L(h_\mu)$ is no larger than the one of $h(y)$. And $L(h)$ has the following upper bound:

$$L(h) \leq L(f) \|K\|^2 = L \|K\|^2 .$$

Therefore, we have

$$\begin{aligned}\|\mathbb{E}_u[g_\mu(x)] - \nabla f(x)\|_{K^2}^2 =& \left\|K^{-1}(\nabla h_\mu(0) - \nabla h(0))\right\|_{K^2}^2 \\ =& \|\nabla h_\mu(0) - \nabla h(0)\|^2 \\ \leq& \frac{\mu^2}{4} L^2 \|K\|^4 (d+3)^3.\end{aligned}$$

□

A.3 Proof of Lemma 2

Proof of Lemma 2. By the definition of $g_\mu(x)$, we have

$$\begin{aligned}\mathbb{E}_u \|g_\mu(x)\|_B^2 =& \frac{1}{\mu^2 b^2} \left(\sum_{i=1}^{b} \mathbb{E}_{u_i} \left([f(x + \mu K u_i) - f(x)]^2 \left\|K^{-1} u_i\right\|_B^2 \right) \right) \\ =& \frac{1}{\mu^2 b} \mathbb{E}_u \left([f(x + \mu K u) - f(x)]^2 \left\|K^{-1} u\right\|_B^2 \right).\end{aligned}$$

The first equality is because u_i's are independent. And we have

$$\begin{aligned}[f(x+\mu Ku)-f(x)]^2 =&[f(x+\mu Ku)-f(x)-\mu\langle\nabla f(x),Ku\rangle+\mu\langle\nabla f(x),Ku\rangle]\\ \le&2\left[\frac{\mu^2}{2}L\|K\|^2\|u\|^2\right]^2+2\mu^2\langle\nabla f(x),Ku\rangle^2\end{aligned}$$

where the inequality follows from Eqn. (2.3) and Cauchy's inequality.

Since we set $K=B^{1/2}$, we can obtain that

$$\begin{aligned}\mathbb{E}_u\left[\langle K\nabla f(x),u\rangle^2\left\|K^{-1}u\right\|_B^2\right] =&\mathbb{E}_u\left[u^\top K\nabla f(x)\nabla f(x)^\top K^\top u\cdot u^\top K^{-1}BK^{-1}u\right]\\ =&(\mathrm{tr}\,K\nabla f(x)\nabla f(x)^\top K^\top)(\mathrm{tr}\,I)+2\mathrm{tr}(K\nabla f(x)\nabla f(x)^\top K^\top)\\ =&d\|\nabla f(x)\|_B^2+2\|\nabla f(x)\|_B^2\\ =&(d+2)\|\nabla f(x)\|_B^2,\end{aligned}$$

where the second equation is because of Lemma 9 with $A=K\nabla f(x)\nabla f(x)^\top K^\top$ and $B=K^{-1}BK^{-1}=I$.

Therefore, we have

$$\begin{aligned}\mathbb{E}_u\|g_\mu(x)\|_B^2\le&\frac{\mu^2}{2b}L^2\|K\|^4\mathbb{E}_u\left[\|u\|^4\cdot\left\|K^{-1}u\right\|_B^2\right]+\frac{1}{b}\cdot\mathbb{E}_u\left[\langle K\nabla f(x),u\rangle^2\left\|K^{-1}u\right\|_B^2\right]\\ \le&\frac{\mu^2}{2b}L^2\|B\|^2\mathbb{E}_u\left[\|u\|^6\right]+\frac{2(d+2)}{b}\cdot\|\nabla f(x)\|_B^2\\ \le&\frac{\mu^2}{2b}L^2\|B\|^2(d+6)^3+\frac{2(d+2)}{b}\cdot\|\nabla f(x)\|_B^2,\end{aligned}\tag{A.3}$$

where the last inequality is due to Lemma 8. □

A.4 Proof of Lemma 3

Proof of Lemma 3. First, we have

$$\begin{aligned}\mathbb{E}_u\left\|K^2g_\mu(x)\right\|_{K^{-2}}^3=&\frac{1}{\mu^3b^3}\sum_{i=1}^b\mathbb{E}_{u_i}\left([f(x+\mu Ku_i)-f(x)]^3\cdot\|Ku_i\|_{K^{-2}}^3\right)\\ =&\frac{1}{\mu^3b^2}\mathbb{E}_u\left([f(x+\mu Cu)-f(x)]^3\cdot\|u\|^3\right).\end{aligned}$$

Furthermore, we can obtain that

$$\begin{aligned}[f(x+\mu Ku)-f(x)]^3=&[f(x+\mu Ku)-f(x)-\mu\langle\nabla f(x),Ku\rangle+\mu\langle\nabla f(x),Ku\rangle]^3\\ \le&4\left[\frac{\mu^2}{2}L\|K\|^2\|u\|^2\right]^3+4\mu^3\langle C\nabla f(x),u\rangle^3\end{aligned}$$

And we also have

$$\begin{aligned}
&\mathbb{E}_u \left[\|u\|^3 \cdot \langle K\nabla f(x), u\rangle^3 \right] \\
=&\mathbb{E}_u \left[\left((u^\top u)^2 \cdot (u^\top K \nabla f(x) \nabla^\top f(x) K u)^2 \right)^{3/4} \right] \\
\leq& \left(\mathbb{E}_u \left[(u^\top u)^2 \cdot (u^\top K \nabla f(x) \nabla^\top f(x) K u)^2 \right] \right)^{3/4},
\end{aligned}$$

where the last inequality is because Jensen's inequality.

Let us denote $A = K\nabla f(x) \nabla^\top f(x) K$. It is easy to check that A is a rank one positive semi-definite matrix. And its trace satisfies that $\mathrm{tr}(A) = \|\nabla f(x)\|_{K^2}^2$. By Lemma 9 with $A = K\nabla f(x)\nabla^\top f(x) K$ and $B = I$, we have

$$\begin{aligned}
&\mathbb{E}_u \left[(u^\top u)^2 \cdot (u^\top K \nabla f(x) \nabla^\top f(x) K u)^2 \right] \\
=&(\mathrm{tr}\ I_d)^2 (\mathrm{tr}\ A)^2 + 16[(\mathrm{tr}\ I_d)(\mathrm{tr}\ I_d A^2) + (\mathrm{tr}\ A)(\mathrm{tr}\ I_d^2 A)] \\
&+ 4[(\mathrm{tr}\ I_d^2)(\mathrm{tr}\ A^2) + 2(\mathrm{tr}\ I_d A)^2] + 2[(\mathrm{tr}\ I_d)^2(\mathrm{tr}\ A^2) + 4(\mathrm{tr} I_d)(\mathrm{tr}\ A)(\mathrm{tr}\ I_d A) \\
&+ (\mathrm{tr}\ A)^2 (\mathrm{tr} I_d^2)] + 16[\mathrm{tr}(I_d A)^2 + 2(\mathrm{tr}\ I_d^2 A^2)] \\
=&d^2 \|\nabla f(x)\|_{K^2}^4 + 16 \left[d \|\nabla f(x)\|_{K^2}^4 + \|\nabla f(x)\|_{K^2}^4 \right] + 4 \left[d \|\nabla f(x)\|_{K^2}^4 + 2 \|\nabla f(x)\|_{K^2}^4 \right] \\
&+ 2 \left[d^2 \|\nabla f(x)\|_{K^2}^4 + 4d \|\nabla f(x)\|_{K^2}^4 + d \|\nabla f(x)\|_{k^2}^4 \right] + 16 \left[\|\nabla f(x)\|_{K^2}^4 + 2 \|\nabla f(x)\|_{K^2}^4 \right] \\
=&(3d^2 + 30d + 72) \cdot \|\nabla f(x)\|_{K^2}^4 .
\end{aligned}$$

Thus, we can obtain that

$$\begin{aligned}
\mathbb{E}_u \left[\|u\|^3 \cdot \langle K \nabla f(x), u \rangle^3 \right] \leq& \left(\mathbb{E}_u \left[(u^\top u)^2 \cdot (u^\top K \nabla f(x) \nabla^\top f(x) K u)^2 \right] \right)^{3/4} \\
\leq& \left((3d^2 + 30d + 72) \cdot \|\nabla f(x)\|_{K^2}^4 \right)^{3/4} \\
\leq& \left(3(d+5)^2 \cdot \|\nabla f(x)\|_{K^2}^4 \right)^{3/4} \\
\leq& 3(d+5)^{3/2} \|\nabla f(x)\|_{K^2}^3
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\mathbb{E}_u \left\| K^2 g_\mu(x) \right\|_{K^{-2}}^3 \leq& \frac{1}{\mu^3 b^2} \mathbb{E}_u \left(4 \left[\frac{\mu^2}{2} L \|K\|^2 \|u\|^2 \right]^3 \|u\|^3 + 4\mu^3 \langle C \nabla f(x), u \rangle^3 \|u\|^3 \right) \\
\leq& \frac{\mu^3 L^3 \|K\|^6}{2b^2} \mathbb{E}_u \left[\|u\|^9 \right] + \frac{12(d+5)^{3/2}}{b^2} \|\nabla f(x)\|_{K^2}^3 \\
\leq& \frac{1}{b^2} \left(2\mu^3 L^3 \|K\|^6 \cdot (d+9)^{9/2} + 12(d+5)^{3/2} \|\nabla f(x)\|_{K^2}^3 \right)
\end{aligned}$$

where the last inequality follows from Lemma 8. □

B Proof of Convergence Rate of Algorithm 1

In this section, we will prove the convergence rate of Algorithm 1. Before that, we first give an important lemma which depicts some properties related to the strong convexity.

Lemma 10. *Let f be continuously differentiable and strongly convex with parameter τ. And $x^\star$ is the minimizer of f. Then for any $x \in \mathbb{R}^d$, we have*

$$\|\nabla f(x)\|^2 \geq 2\tau(f(x) - f(x^\star),$$

and

$$\|x - x^\star\|^2 \leq \frac{2}{\tau}(f(x) - f(x^\star)).$$

Proof. First, by the strong convexity of f,

$$\begin{aligned} f(x^\star) \geq& f(x) + \langle \nabla f(x), x^\star - x\rangle + \frac{\tau}{2}\|x^\star - x\|^2 \\ \geq& f(x) + \min_v \left(\langle \nabla f(x), v\rangle + \frac{\tau}{2}\|v\|^2\right) \\ =& f(x) - \frac{1}{2\tau}\|\nabla f(x)\|^2. \end{aligned}$$

The last equality holds by plugging in the minimizer $v = -\nabla f(x)/\tau$.

Also by the strong convexity of f, we have

$$\begin{aligned} f(x) \geq& f(x^\star) + \langle \nabla f(x^\star), x - x^\star\rangle + \frac{\tau}{2}\|x^\star - x\|^2 \\ =& f(x^\star) + \frac{\tau}{2}\|x^\star - x\|^2. \end{aligned}$$

□

B.1 Proof of Theorem 1

Now, we give the proof of the local convergence properties depicted in Theorem 1.

Proof of Theorem 1. Taking a random step from x_t, we have

$$\begin{aligned}
&\mathbb{E}_u\left[f(x_{t+1})\right]\\
=&\mathbb{E}_u\left[f(x_t-\eta\tilde{H}^{-1}g_\mu(x_t))\right]\\
\overset{(2.5)}{\le}& f(x_t)-\eta\left\langle\nabla f(x_t),\tilde{H}^{-1}\mathbb{E}_u[g_\mu(x_t)]\right\rangle+\frac{\eta^2}{2}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|_H^2+\frac{\eta^3\gamma}{6}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|^3\\
\overset{(3.1)}{\le}& f(x_t)-\eta\left\|\nabla f(x_t)\right\|_{\tilde{H}^{-1}}-\eta\left\langle\nabla f(x_t),\tilde{H}^{-1}\left(\mathbb{E}_u[g_\mu(x_t)]-\nabla f(x_t)\right)\right\rangle\\
&+\frac{\eta^2}{2}\mathbb{E}_u\left\|g_\mu(x_t)\right\|_{\tilde{H}^{-1}}^2+\frac{\eta^3\gamma}{6}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|^3\\
\le& f(x_t)-\eta\left\|\nabla f(x_t)\right\|_{\tilde{H}^{-1}}^2+\frac{\eta}{2}\left(\left\|\nabla f(x_t)\right\|_{\tilde{H}^{-1}}^2+\left\|\mathbb{E}_u[g_\mu(x_t)]-\nabla f(x_t)\right\|_{\tilde{H}^{-1}}^2\right)\qquad\text{(B.1)}\\
&+\frac{\eta^2}{2}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|_{\tilde{H}}^2+\frac{\eta^3\gamma}{6}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|^3\\
=& f(x_t)-\frac{\eta}{2}\left\|\nabla f(x_t)\right\|_{\tilde{H}^{-1}}^2+\underbrace{\frac{\eta}{2}\left\|\mathbb{E}_u[g_\mu(x_t)]-\nabla f(x_t)\right\|_{\tilde{H}^{-1}}^2}_{T_1}+\underbrace{\frac{\eta^2}{2}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|_{\tilde{H}}^2}_{T_2}\\
&+\underbrace{\frac{\eta^3\gamma}{6}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|^3}_{T_3},
\end{aligned}$$

where inequality (B.1) follows from Cauchy's Inequality.

Now we begin to bound terms T_1, T_2, and T_3. First, by Lemma 1, we have

$$\left\|\mathbb{E}_u[g_\mu(x_t)]-\nabla f(x_t)\right\|_{\tilde{H}^{-1}}^2=\left\|\mathbb{E}_u[g_\mu(x_t)]-\nabla f(x_t)\right\|_{K^2}^2\le\frac{\mu^2L^2}{4}\left\|\tilde{H}^{-1}\right\|^2(d+3)^3,$$

that is,

$$T_1\le\frac{\eta\mu^2L^2}{8}\left\|\tilde{H}^{-1}\right\|^2(d+3)^3.$$

By Lemma 2 with $B=\tilde{H}^{-1}$, we have

$$\begin{aligned}
\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|_{\tilde{H}}^2=&\mathbb{E}_u\left\|g_\mu(x_t)\right\|_{\tilde{H}^{-1}}^2\\
\le&\frac{\mu^2}{2b}L^2\left\|\tilde{H}^{-1}\right\|(d+6)^3+\frac{2(d+2)}{b}\left\|\nabla f(x)\right\|_{\tilde{H}^{-1}}^2.
\end{aligned}$$

Thus, T_2 is upper bounded as

$$T_2\le\frac{\mu^2L^2\eta^2}{4b}\left\|\tilde{H}^{-1}\right\|(d+6)^3+\frac{\eta^2(d+2)}{b}\left\|\nabla f(x)\right\|_{\tilde{H}^{-1}}^2.$$

Using Lemma 3, we have

$$\begin{aligned}T_3 =&\frac{\eta^3}{6}\cdot\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|^3\\ \le&\frac{\eta^3}{6}\cdot\left\|\tilde{H}^{-1}\right\|^{3/2}\cdot\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|_{\tilde{H}}^3\\ \le&\frac{\gamma\mu^3L^3\eta^3\left\|\tilde{H}^{-1}\right\|^{9/2}}{12b^2}\cdot(d+9)^{9/2}+\frac{2\gamma\eta^3}{b^2}(d+5)^{3/2}\left\|\tilde{H}^{-1}\right\|^{3/2}\cdot\|\nabla f(x)\|_{\tilde{H}^{-1}}^3\end{aligned}$$

Thus, we have

$$\begin{aligned}T_1+T_2+T_3\le&\frac{\eta\mu^2L^2}{8}\left\|\tilde{H}^{-1}\right\|^2(d+3)^3+\frac{\mu^2L^2\eta^2}{4b}\left\|\tilde{H}^{-1}\right\|(d+6)^3+\frac{\eta^2(d+2)}{b}\|\nabla f(x)\|_{\tilde{H}^{-1}}^2\\ &+\frac{\gamma\mu^3L^3\eta^3\left\|\tilde{H}^{-1}\right\|^{9/2}}{12b^2}\cdot(d+9)^{9/2}+\frac{2\gamma\eta^3}{b^2}(d+5)^{3/2}\left\|\tilde{H}^{-1}\right\|^{3/2}\cdot\|\nabla f(x)\|_{\tilde{H}^{-1}}^3.\end{aligned}$$

By choosing $\eta=\frac{b}{4(d+2)}$, we obtain that

$$\begin{aligned}&\mathbb{E}_u\left[f(x_{t+1})\right]\\ \le&f(x_t)-\frac{\eta}{2}\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2+T_1+T_2+T_3\\ \le&f(x_t)-\frac{b}{16(d+2)}\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2+\frac{\mu^2L^2b}{32}\left\|\tilde{H}^{-1}\right\|^2(d+5)^2+\frac{b\mu^2L^2\left\|\tilde{H}^{-1}\right\|}{64}(d+38)\\ &+\frac{b\gamma\mu^3L^3\left\|\tilde{H}^{-1}\right\|^{9/2}}{768}\cdot(d+110)^{3/2}+\frac{b\gamma}{64}d^{-3/2}\left\|\tilde{H}^{-1}\right\|^{3/2}\cdot\|\nabla f(x)\|_{\tilde{H}^{-1}}^3\\ =&f(x_t)-\frac{b}{16(d+2)}\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2+\frac{b\gamma}{64}d^{-3/2}\left\|\tilde{H}^{-1}\right\|^{3/2}\cdot\|\nabla f(x)\|_{\tilde{H}^{-1}}^3\\ &+b\cdot\left(\frac{\mu^2L^2}{32}\left\|\tilde{H}^{-1}\right\|^2(d+5)^2+\frac{\mu^2L^2\left\|\tilde{H}^{-1}\right\|}{64}(d+38)+\frac{\gamma\mu^3L^3\left\|\tilde{H}^{-1}\right\|^{9/2}}{768}\cdot(d+110)^{3/2}\right)\\ =&f(x_t)-\frac{b}{16(d+2)}\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2+\frac{b\gamma}{64}d^{-3/2}\left\|\tilde{H}^{-1}\right\|^{3/2}\cdot\|\nabla f(x)\|_{\tilde{H}^{-1}}^3+\Delta_\mu,\end{aligned}$$

where we denote

$$\Delta_\mu=b\cdot\left(\frac{\mu^2L^2}{32}\left\|\tilde{H}^{-1}\right\|^2(d+5)^2+\frac{\mu^2L^2\left\|\tilde{H}^{-1}\right\|}{64}(d+38)+\frac{\gamma\mu^3L^3\left\|\tilde{H}^{-1}\right\|^{9/2}}{768}\cdot(d+110)^{3/2}\right).$$

Now, we begin to give the connections between $\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2$ and $f(x_t)-f(x^\star)$. First, by the

Taylor's expansion, we have

$$\begin{aligned}\nabla f(x_t) =&\nabla f(x^\star) + \nabla^2 f(x^\star)(x_t - x^\star) + \int_0^1 \left(\nabla^2 f(x^\star + s(x_t - x^\star)) - \nabla^2 f(x^\star)\right)(x_t - x^\star) ds\\ =&\nabla^2 f(x^\star)(x_t - x^\star) + \Delta_1.\end{aligned}$$

where the last equation is because $\nabla f(x^\star) = 0$ and we denote that

$$\Delta_1 \triangleq \int_0^1 \left(\nabla^2 f(x^\star + s(x_t - x^\star)) - \nabla^2 f(x^\star)\right)(x_t - x^\star) ds.$$

And $\|\Delta_1\|$ is upper bounded as

$$\|\Delta_1\| \le \rho \int_0^1 s \|x_t - x^\star\|^2 \; ds = \frac{\rho}{2} \|x_t - x^\star\|^2 . \tag{B.2}$$

Let us denote $H_\star = \nabla^2 f(x^\star)$. We have

$$\begin{aligned}&- \|\nabla f(x_t)\|^2_{\tilde{H}^{-1}}\\ =&- \|H_\star(x_t - x^\star) + \Delta_1\|^2_{\tilde{H}^{-1}}\\ =&- \|H(x_t - x^\star) + \Delta_1 + (H_\star - H)(x_t - x^\star)\|^2_{\tilde{H}^{-1}}\\ \le&- \left(\|H(x_t - x^\star)\|^2_{\tilde{H}^{-1}} + \|\Delta_1\|^2_{\tilde{H}^{-1}} + \|(H_\star - H)(x_t - x^\star)\|^2_{\tilde{H}^{-1}}\right)\\ &+ 2\Bigg(\|H(x_t - x^\star)\|_{\tilde{H}^{-1}} \cdot \|\Delta_1\|_{\tilde{H}^{-1}} + \|\Delta_1\|_{\tilde{H}^{-1}} \cdot \|(H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}}\\ &+ \|(H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}} \cdot \|H(x_t - x^\star)\|_{\tilde{H}^{-1}}\Bigg)\\ \le&- \rho \|x_t - x^\star\|^2_H + \Delta_2\end{aligned}$$

where the last inequality is because of the condition that $\rho\tilde{H} \preceq H$ and we denote that

$$\begin{aligned}\Delta_2 =&2 \|H(x_t - x^\star)\|_{\tilde{H}^{-1}} \cdot \|\Delta_1\|_{\tilde{H}^{-1}} + 2 \|\Delta_1\|_{\tilde{H}^{-1}} \cdot \|(H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}}\\ &+ 2 \|(H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}} \cdot \|H(x_t - x^\star)\|_{\tilde{H}^{-1}}\end{aligned}$$

Furthermore, we have

$$\begin{aligned}f(x_t) \overset{(2.5)}{\le}& f(x^\star) + \langle\nabla f(x^\star), x_t - x^\star\rangle + \frac{1}{2}\langle\nabla^2 f(x^\star)(x_t - x^\star), x_t - x^\star\rangle + \frac{\gamma}{6} \|x_t - x^\star\|^3\\ =&f(x^\star) + \frac{1}{2} \|x_t - x^\star\|^2_{H_\star} + \frac{\gamma}{6} \|x_t - x^\star\|^3 .\end{aligned}$$

Hence, we have

$$-\frac{1}{2} \|x_t - x^\star\|^2_{H_\star} \le - (f(x_t) - f(x^\star)) + \frac{\gamma}{6} \|x_t - x^\star\|^3 \tag{B.3}$$

Therefore, we obtain that

$$\begin{aligned}
&-\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2 \\
\le& -\rho\,\|x_t - x^\star\|_H^2 + \Delta_2 \\
\le& -\rho\left(\|x_t - x^\star\|_{H^\star}^2 + \langle x_t - x^\star, (H - H^\star)(x_t - x^\star)\rangle\right) + \Delta_2 \\
\overset{(B.3)}{\le}& -2\rho\left(f(x_t) - f(x^\star) - \frac{\gamma}{6}\|x_t - x^\star\|^3 - \frac{1}{2}\langle x_t - x^\star, (H - H^\star)(x_t - x^\star)\rangle\right) + \Delta_2 \\
\overset{(2.4)}{\le}& -2\rho(f(x_t) - f(x^\star)) + \frac{4\gamma\rho}{3}\|x_t - x^\star\|^3 + \Delta_2.
\end{aligned}$$

Now we begin to bound the value of Δ_2. First, by the condition $\nabla^2 f(x) \le (1 + (1-\rho))\tilde{H}$, we have

$$\begin{aligned}
\|H(x_t - x^\star)\|_{\tilde{H}^{-1}} \cdot \|\Delta_1\|_{\tilde{H}^{-1}} \le& \|x_t - x^\star\| \cdot \left\|H\tilde{H}^{-1}H\right\|^{1/2} \cdot \|\Delta_1\| \cdot \left\|\tilde{H}^{-1/2}\right\| \\
\le& \|x_t - x^\star\| \cdot \sqrt{2}\left\|H^{1/2}\right\| \cdot \frac{\gamma}{2}\|x_t - x^\star\|^2 \cdot \left\|\tilde{H}^{-1/2}\right\| \\
\le& \frac{\sqrt{2}\gamma L^{1/2}}{2}\|x_t - x^\star\|^3 \cdot \left\|\tilde{H}^{-1/2}\right\|
\end{aligned}$$

And, we also have

$$\begin{aligned}
\|\Delta_1\|_{\tilde{H}^{-1}} \cdot \|(H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}} \le& \|\Delta_1\| \cdot \left\|\tilde{H}^{-1/2}\right\| \cdot \gamma\,\|x_t - x^\star\| \cdot \|x_t - x^\star\| \left\|\tilde{H}^{-1/2}\right\| \\
\le& \frac{\gamma^2}{2}\|x_t - x^\star\|^4 \cdot \left\|\tilde{H}^{-1}\right\|
\end{aligned}$$

where the first inequality is because the condition that $\|H_t - H_\star\| \le \gamma\,\|x_t - x^\star\|$.

Finally, we have

$$\begin{aligned}
&\|(H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}} \cdot \|H(x_t - x^\star)\|_{\tilde{H}^{-1}} \\
\le& \gamma\,\|x_t - x^\star\|^2 \cdot \left\|\tilde{H}^{-1/2}\right\| \cdot \|x_t - x^\star\| \cdot \left\|H\tilde{H}^{-1}H\right\|^{1/2} \\
\le& \sqrt{2}\gamma L^{1/2}\,\|x_t - x^\star\|^3 \cdot \left\|\tilde{H}^{-1/2}\right\|
\end{aligned}$$

Combining the conditions that $\|x_t - x^\star\| \le \frac{\rho}{\gamma} \cdot \frac{\tau\zeta^{1/2}}{L^{1/2}}$ and $\lambda_{\min}(\tilde{H}) \ge \zeta$, we have

$$\begin{aligned}
\Delta_2 \le& 3\sqrt{2}\gamma L^{1/2}\,\|x_t - x^\star\|^3 \cdot \left\|\tilde{H}^{-1/2}\right\| + \gamma^2\,\|x_t - x^\star\|^4 \cdot \left\|\tilde{H}^{-1}\right\| \\
\le& 5\gamma L^{1/2}\zeta^{-1/2}\,\|x_t - x^\star\|^3 + \frac{\tau}{L} \cdot \rho\gamma L^{1/2}\zeta^{-1/2}\,\|x_t - x^\star\|^3 \\
\le& 6\gamma L^{1/2}\zeta^{-1/2}\,\|x_t - x^\star\|^3\,, \qquad \text{(B.4)}
\end{aligned}$$

where the last inequality is due to $\tau \le L$ and $\rho \le 1$.

Thus, we get that

$$
\begin{aligned}
-\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2 \le& -2\rho(f(x_t)-f(x^\star)) + \frac{4\gamma\rho}{3}\|x_t - x^\star\|^3 + 6\gamma L^{1/2}\zeta^{-1/2}\|x_t - x^\star\|^3 \\
\le& -2\rho(f(x_t)-f(x^\star)) + \frac{22}{3}\gamma L^{1/2}\zeta^{-1/2}\|x_t - x^\star\|^3,
\end{aligned}
\tag{B.5}
$$

where the last inequality follows from that $\rho \le 1$ and $\zeta \le L$.

Now, we begin to bound the value of term $\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^3$. First, we have

$$
\begin{aligned}
\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^3 =& \|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2 \cdot \|\nabla f(x_t)\|_{\tilde{H}^{-1}} \\
\le& \|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2 \cdot \left\|\tilde{H}^{-1/2}\right\| \|\nabla f(x_t)\| \\
\overset{(2.2)}{\le}& L\zeta^{-1/2}\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2 \cdot \|x_t - x^\star\|.
\end{aligned}
$$

We can bound the value of $\|\nabla f(x)\|_{\tilde{H}^{-1}}^2$ as follows:

$$
\begin{aligned}
&\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2 \\
=& \|H(x_t - x^\star) + \Delta_1 + (H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}}^2 \\
\le& \|H(x_t - x^\star)\|_{\tilde{H}^{-1}}^2 + \|\Delta_1\|_{\tilde{H}^{-1}}^2 + \|(H_\star - H)(x_t - x^\star)\|_{\tilde{H}^{-1}}^2 + \Delta_2 \\
\overset{(3.1)}{\le}& \|x_t - x^\star\|_H^2 + 2\gamma^2\left\|\tilde{H}^{-1}\right\| \cdot \|x_t - x^\star\|^4 + \Delta_2 \\
=& \|x_t - x^\star\|_{H^\star}^2 + \langle x_t - x^\star, (H - H^\star)(x_t - x^\star)\rangle + 2\gamma^2\left\|\tilde{H}^{-1}\right\| \cdot \|x_t - x^\star\|^4 + \Delta_2 \\
\overset{(2.4)}{\le}& \|x_t - x^\star\|_{H^\star}^2 + \gamma\|x_t - x^\star\|^3 + 2\gamma^2\left\|\tilde{H}^{-1}\right\| \cdot \|x_t - x^\star\|^4 + \Delta_2 \\
\le& \|x_t - x^\star\|_{H^\star}^2 + 7\gamma L^{1/2}\zeta^{-1/2}\|x_t - x^\star\|^3.
\end{aligned}
$$

The last inequality is because of Eqn. (B.4), $1 \le L^{1/2}\zeta^{-1/2}$ and the condition $\|x_t - x^\star\| \le \frac{\rho}{\gamma} \cdot \frac{\tau\zeta^{1/2}}{L^{1/2}}$.

We also have that

$$
\begin{aligned}
f(x_t) \ge& f(x^\star) + \langle \nabla f(x^\star), x_t - x^\star\rangle + \frac{1}{2}\langle \nabla^2 f(x^\star)(x_t - x^\star), x_t - x^\star\rangle - \frac{\gamma}{6}\|x_t - x^\star\|^3 \\
=& f(x^\star) + \frac{1}{2}\|x_t - x^\star\|_{H_\star}^2 - \frac{\gamma}{6}\|x_t - x^\star\|^3.
\end{aligned}
$$

Therefore, we have

$$
\begin{aligned}
\|\nabla f(x)\|_{\tilde{H}^{-1}}^3 \le& L\zeta^{-1/2}\|\nabla f(x_t)\|_{\tilde{H}^{-1}}^2 \cdot \|x_t - x^\star\| \\
\le& L\zeta^{-1/2} \cdot \|x_t - x^\star\| \left(2(f(x_t) - f(x^\star) + \frac{\gamma}{3}\|x_t - x^\star\|^3 + 7\gamma L^{1/2}\zeta^{-1/2}\|x_t - x^\star\|^3\right) \\
\le& 2L\zeta^{-1/2}\|x_t - x^\star\| \cdot (f(x_t) - f(x^\star)) + 15\gamma L^{3/2}\zeta^{-1}\|x_t - x^\star\|^4,
\end{aligned}
\tag{B.6}
$$

where the last inequality is due to the fact that $1 \le L^{1/2}\zeta^{-1/2}$.

Therefore, we can obtain that

$$\begin{aligned}
&\mathbb{E}_u[f(x_{t+1}) - f(x^\star)] \\
\leq & f(x_t) - f(x^\star) - \frac{b}{16(d+2)} \|\nabla f(x_t)\|^2_{\tilde{H}^{-1}} + \frac{b\gamma}{64} d^{-3/2} \left\|\tilde{H}^{-1}\right\|^{3/2} \cdot \|\nabla f(x)\|^3_{\tilde{H}^{-1}} + \Delta_\mu \\
\overset{(B.5)}{\leq} & f(x_t) - f(x^\star) - \frac{b\rho}{8(d+2)} (f(x_t) - f(x^\star)) + \frac{2b\gamma L^{1/2}\zeta^{-1/2}}{3(d+2)} \|x_t - x^\star\|^3 \\
& + \frac{b\gamma}{64} d^{-3/2} \left\|\tilde{H}^{-1}\right\|^{3/2} \cdot \|\nabla f(x)\|^3_{\tilde{H}^{-1}} + \Delta_\mu \\
\overset{(B.6)}{\leq} & \left(1 - \frac{b\rho}{8(d+2)}\right) (f(x_t) - f(x^\star)) + \frac{2b\gamma L^{1/2}\zeta^{-1/2}}{3(d+2)} \|x_t - x^\star\|^3 \\
& + \frac{b\gamma L\zeta^{-2}}{32 d^{3/2}} \|x_t - x^\star\| \cdot (f(x_t) - f(x^\star)) + \frac{b\gamma^2 L^{3/2}\zeta^{-5/2}}{4d^{3/2}} \|x_t - x^\star\|^4 + \Delta_\mu.
\end{aligned}$$

Since the objective function is τ-strongly convex, by Lemma 10, we have

$$\|x_t - x^\star\|^3 \leq \frac{2}{\tau} \cdot \|x_t - x^\star\| \cdot (f(x_t) - f(x^\star))\,.$$

And by the condition of $\|x_t - x^\star\| \, \frac{\rho}{\gamma} \cdot \frac{\tau\zeta^{1/2}}{L^{1/2}}$, we also have

$$\begin{aligned}
\frac{b\gamma^2 L^{3/2}\zeta^{-5/2}}{4d^{3/2}} \|x_t - x^\star\|^4 \leq & \frac{b\gamma^2 L^{3/2}\zeta^{-5/2}}{4d^{3/2}} \cdot \frac{2}{\tau} \|x_t - x^\star\|^2 \cdot (f(x_t) - f(x^\star)) \\
\leq & \frac{b\gamma L\zeta^{-2}}{2d^{3/2}} \|x_t - x^\star\| \cdot (f(x_t) - f(x^\star))
\end{aligned}$$

Thus, we have

$$\begin{aligned}
&\mathbb{E}_u\left[f(x_{x+1}) - f(x^\star)\right] \\
\leq & \left(1 - \frac{b\rho}{8(d+2)}\right) (f(x_t) - f(x^\star)) + \frac{4b\gamma L^{1/2}\zeta^{-1/2}}{3\tau(d+2)} \|x_t - x^\star\| \cdot (f(x_t) - f(x^\star)) \\
& + \left(\frac{b\gamma L\zeta^{-2}}{32d^{3/2}} + \frac{b\gamma L\zeta^{-2}}{2d^{3/2}}\right) \|x_t - x^\star\| \cdot (f(x_t) - f(x^\star)) + \Delta_\mu \\
= & \left(1 - \frac{b\rho}{8(d+2)}\right) (f(x_t) - f(x^\star)) + b\gamma \cdot \Delta_3 \cdot \|x_t - x^\star\| \cdot (f(x_t) - f(x^\star)) + \Delta_\mu,
\end{aligned}$$

where Δ_3 is defined as

$$\Delta_3 = \frac{4L^{1/2}\zeta^{-1/2}}{3\tau(d+2)} + \frac{17L\zeta^{-2}}{32d^{3/2}}.$$

To keep a fast convergence rate, we need

$$\begin{aligned}\|x_t - x^\star\| \le & \frac{1}{2\gamma\Delta_3}\frac{\rho}{8(d+2)} \\ \le & \frac{\rho}{\gamma}\cdot\min\left(\frac{3\tau\zeta^{1/2}}{64L^{1/2}}, \frac{d^{3/2}\zeta^2}{17L(d+2)}\right)\end{aligned}$$

Thus, when x_t satisfies the above condition, we can obtain that

$$\mathbb{E}_u\left[f(x_{x+1}) - f(x^\star)\right] \le \left(1 - \frac{b\rho}{16(d+2)}\right)\left(f(x_t) - f(x^\star)\right) + \Delta_\mu.$$

□

B.2 Proof of Theorem 2

Now, we give the global convergence property of Algorithm 1 in Theorem 2.

Proof of Theorem 2. Taking a random step from x_t, we have

$$\begin{aligned}
&\mathbb{E}_u\left[f(x_{t+1})\right] \\
=&\mathbb{E}_u\left[f(x_t - \eta\tilde{H}^{-1}g_\mu(x_t))\right] \\
\overset{(2.2)}{\le}&\mathbb{E}_u\left[f(x_t) - \eta\left\langle\nabla f(x_t), \tilde{H}^{-1}\mathbb{E}_u[g_\mu(x_t)]\right\rangle + \frac{\eta^2 L}{2}\mathbb{E}_u\left\|\tilde{H}^{-1}g_\mu(x_t)\right\|^2\right] \\
\le&f(x_t) - \eta\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} - \eta\left\langle\nabla f(x_t), \tilde{H}^{-1}\left(\mathbb{E}_u[g_\mu(x_t)] - \nabla f(x_t)\right)\right\rangle + \frac{\eta^2 L\left\|\tilde{H}^{-1}\right\|}{2}\mathbb{E}_u\left\|g_\mu(x_t)\right\|^2_{\tilde{H}^{-1}} \\
\le&f(x_t) - \eta\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} + \frac{\eta}{2}\left(\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} + \left\|\mathbb{E}_u[g_\mu(x_t)] - \nabla f(x_t)\right\|^2_{\tilde{H}^{-1}}\right) \qquad \text{(B.7)} \\
&+\frac{\eta^2 L\left\|\tilde{H}^{-1}\right\|}{2}\mathbb{E}_u\left\|g_\mu(x_t)\right\|^2_{\tilde{H}^{-1}} \\
\le&f(x_t) - \frac{\eta}{2}\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} + \frac{\eta\mu^2L^2}{8}\left\|\tilde{H}^{-1}\right\|^2(d+3)^3 + \frac{\eta^2 L\left\|\tilde{H}^{-1}\right\|}{2}\mathbb{E}_u\left\|g_\mu(x_t)\right\|^2_{\tilde{H}^{-1}} \qquad \text{(B.8)} \\
\le&f(x_t) - \frac{\eta}{2}\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} + \frac{\eta\mu^2L^2}{8}\left\|\tilde{H}^{-1}\right\|^2(d+3)^3 \\
&+\frac{\eta^2 L\left\|\tilde{H}^{-1}\right\|}{2b}\left(\frac{\mu^2}{2}L^2\left\|\tilde{H}^{-1}\right\|(d+6)^3 + 2(d+2)\left\|\nabla f(x)\right\|^2_{\tilde{H}^{-1}}\right) \qquad \text{(B.9)} \\
=&f(x_t) - \frac{\eta}{2}\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} + \eta^2 L(d+2)b^{-1}\left\|\tilde{H}^{-1}\right\|\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} \\
&+\frac{\eta\mu^2L^2}{8}\left\|\tilde{H}^{-1}\right\|^2(d+3)^3 + \frac{\eta^2L^3\mu^2\left\|\tilde{H}^{-2}\right\|}{4b}(d+6)^3
\end{aligned}$$

Inequality (B.7) is because of Cauchy's inequality and $2ab \le a^2 + b^2$. Inequality (B.8) follows from

Lemma 1. and inequality (B.9) is due to Lemma 2 with $B = \tilde{H}^{-1}$.

Since $\zeta I \preceq \tilde{H}$ holds for all iterations, let us set the step size η as

$$\eta = \frac{b\zeta}{4(d+2)L}.$$

Thus we obtain that

$$\mathbb{E}_u\left[f(x_{t+1})\right] \leq f(x_t) - \frac{b\zeta}{16(d+2)L}\left\|\nabla f(x_t)\right\|^2_{\tilde{H}^{-1}} + \Delta_\mu,$$

where Δ_μ denotes that

$$\Delta_\mu = \frac{b\mu^2 L}{32(d+2)\zeta}(d+3)^2 + \frac{b\mu^2 L}{64\zeta(d+2)^2}(d+6)^3 = \frac{b\mu^2 L}{32\zeta(d+2)}\left((d+3)^3 + \frac{(d+6)^3}{2(d+2)}\right).$$

By Lemma 10, we have

$$\begin{aligned}\mathbb{E}_u\left[f(x_{t+1}) - f(x^\star)\right] \leq & f(x_t) - f(x^\star) - \frac{\zeta}{16(d+2)L^2}\left\|\nabla f(x_t)\right\|^2 + \Delta_\mu \\ \leq & f(x_t) - f(x^\star) - \frac{\zeta\tau}{16(d+2)L^2}(f(x_t) - f(x^\star)) + \Delta_\mu \\ = & \left(1 - \frac{\zeta}{16(d+2)\kappa L}\right)\cdot(f(x_t) - f(x^\star)) + \Delta_\mu\end{aligned}$$

□

C Proof of Section 3.4

C.1 Proof of Theorem 3

Our proof of Theorem 3 relies on the existing results about noisy power method depicted in Algorithm 4. Because of the approximation, we can cast our zeroth-order power method as the noisy power method and the product $H_\mu V_t$ can be regarded as $H_\mu V_t = \nabla^2 f(x)V_t + G_t$ with G_t being the noise matrix. Balcan et al. (2016) showed that the noisy power method converges to the principal eigenvalues and eigenvectors of the matrix and have the follow properties.

Lemma 11 (Balcan et al. (2016))**.** *Given positive definite matrix $H \in \mathbb{R}^{d\times d}$ and $0 < \epsilon < 1$, suppose that noise matrix satisfies*

$$\|G_t\| \leq C_1\epsilon^2\lambda_{k+1} \quad \text{and} \quad \|V_k^T G_t\|_2 = C_2\epsilon^2\lambda_{k+1}\cos\left(V_k, X_t\right).$$

Then, after T iterations, V_T returned by Algorithm 4 satisfies

$$\left\|H - V_T V_T^T H\right\| \leq (1+\epsilon)\left\|H - H_k\right\|, \quad \text{if} \quad T = \frac{C_3}{\epsilon}\log\left(\frac{\tan(U_k, V_0)}{\epsilon}\right).$$

where $H_k = U_k \Lambda_k U_k^\top$ is the best rank-k approximation of H. C_1, C_2, and C_3 are absolute constants.

Proof of Theorem 3. First, we us denote $H = \nabla^2 f(x)$. By the definition of V_T and $\hat{V}$ in Algorithm 2, we have

$$\begin{aligned}
\left\|H - V\Lambda V^\top\right\| \le& \left\|H - V_T V_T^\top H V_T V_T^\top\right\| + \left\|V_T V_T^\top H V_T V_T^\top - V\Lambda V^\top\right\| \\
\le& \left\|H - V_T V_T^\top H\right\| + \left\|V_T V_T^\top H - V_T V_T^\top H V_T V_T^\top\right\| + \left\|V_T V_T^\top H V_T V_T^\top - V_T \hat{V}\Lambda\hat{V}^\top V_T^\top\right\| \\
\le& 2\left\|H - V_T V_T^\top H\right\| + \left\|V_T V_T^\top H V_T V_T^\top - V_T \hat{V}\Lambda\hat{V}^\top V_T^\top\right\| \\
\le& 3\lambda_{k+1} + \left\|V_T V_T^\top H V_T V_T^\top - V_T \hat{V}\Lambda\hat{V}^\top V_T^\top\right\|,
\end{aligned}$$

where the last inequality is because of Lemma 11 with $\epsilon = 1/2$.

Furthermore, by the step 6 of Algorithm 2, we have

$$\begin{aligned}
\left\|V_T V_T^\top H V_T V_T^\top - V_T \hat{V}\Lambda\hat{V}^\top V_T^\top\right\| =& \left\|V_T^\top H V_T - \hat{V}\Lambda\hat{V}^\top\right\| \\
=& \left\|V_T^\top H V_T - \hat{V} U^\top H_\mu V_T\right\| \\
\le& \left\|V_T^\top - \hat{V} U^\top\right\| \cdot \left\|H V_T - H_\mu V_T\right\| \\
\le& 2\left\|H V_T - H_\mu V_T\right\|,
\end{aligned}$$

where the last inequality is because of $\left\|V_T^\top - \hat{V}U^\top\right\| \le \|V_T\| + \left\|\hat{V}U^\top\right\| \le 2$.

Now, we begin to bound the value of $\|HV_T - H_\mu V_T\|$. Let v be a unit vector, we have

$$\begin{aligned}
\left\|\frac{\nabla f(x + \mu_1 \cdot v) - \nabla f(x)}{\mu_1} - \nabla^2 f(x) v\right\| =& \left\|\frac{\nabla f(x + \mu_1 \cdot v) - \nabla f(x) - \mu_1 \nabla^2 f(x) v}{\mu_1}\right\| \\
=& \left\|\frac{\mu_1 \cdot v\left(\nabla^2 f(\tilde{x}) - \nabla^2 f(x)\right)}{\mu_1}\right\| \\
\le& \mu_1 \cdot \gamma \|v\|^2 = \gamma \mu_1,
\end{aligned}$$

where the last inequality is because of Eqn. (2.4) and $\tilde{x} = x + \theta \cdot (\mu_1 v)$ with $0 \le \theta \le 1$.

Let $\tilde{\nabla} f(x)$ denote the approximate gradient computed as follows:

$$\tilde{\nabla}_j f(x) = \frac{f(x + \mu_1 \cdot e_j) - f(x - \mu_1 \cdot e_j)}{2\mu_1}.$$

We also use [] Then, we have

$$
\begin{aligned}
&\left\|\nabla f(x) - \tilde{\nabla} f(x)\right\| \\
=&\left\|\left[\tilde{\nabla}_1 f(x) - \nabla_1 f(x), \dots, \tilde{\nabla}_d f(x) - \nabla_d f(x)\right]^\top\right\| \\
=&\frac{1}{2\mu_1} \cdot \Big\| \big[\, (f(x+\mu_1 \cdot e_1) - f(x - \mu_1 \cdot e_1) - 2\mu_1 \nabla_j f(x)) \,, \dots, \\
&\qquad (f(x+\mu_1 \cdot e_d) - f(x - \mu_1 \cdot e_d) - 2\mu_1 \nabla_d f(x)) \,\big]^\top \Big\| \\
=&\left\|\frac{\mu_1^2 \cdot \mathrm{diag}\left(\nabla^2 f(\tilde{x}_1) - \nabla^2(\tilde{x}_2)\right)}{4\mu_1}\right\| \\
\overset{(2.4)}{\le}& \frac{\mu_1}{4}\gamma \left\|\tilde{x}_1 - \tilde{x}_2\right\| \\
\le&\frac{\gamma\mu_1^2}{2},
\end{aligned}
$$

where the last inequality is because of $\tilde{x}_1 = x + \theta_1 \cdot (\mu_1 v)$ with $0 \le \theta_1 \le 1$ and $\tilde{x}_2 = x - \theta_2 \cdot (\mu_1 v)$ with $0 \le \theta_2 \le 1$.

Combining the above results, we have

$$
\begin{aligned}
\left\|HV_T - H_\mu V_T\right\| =& \left\|H[v_1, v_2, \dots, v_k] - H_\mu[v_1, v_2, \dots, v_k]\right\| \\
\le& k \left\|Hv_1 - H_\mu v_1\right\| \\
=& k \left\|\frac{\tilde{\nabla} f(x + \mu_1 \cdot v) - \tilde{\nabla} f(x)}{\mu_1} - \nabla^2 f(x) v\right\| \\
\le& k \left\|\frac{\nabla f(x + \mu_1 \cdot v) - \nabla f(x)}{\mu_1} - \nabla^2 f(x) v\right\| \\
&+ k \left\|\frac{\tilde{\nabla} f(x + \mu_1 \cdot v) - \nabla f(x + \mu_1 \cdot v) + \nabla f(x) - \tilde{\nabla} f(x)}{\mu_1}\right\| \\
\le& k \left(\mu_1 \gamma \left\|v_1\right\|^2 + 2\frac{\gamma\mu_1^2}{2\mu_1}\right) \\
\le& 2k\mu_1\gamma.
\end{aligned}
$$

Therefore, we have

$$
\left\|H - V\Lambda V^\top\right\| \le 3\lambda_{k+1} + 2\left\|HV_T - H_\mu V_T\right\| \le 5\lambda_{k+1}
$$

where the last inequality is because that we set μ and μ_1 as follows

$$
k\gamma\mu_1 + 2\sqrt{d}L\frac{\mu}{\mu_1} \le 2\lambda_{k+1}
$$

Furthermore, we can obtain that

$$
\begin{aligned}
&\left\| H - VV^\top H_\mu VV^\top \right\| \leq 5\lambda_{k+1} \\
\Rightarrow & - 5\lambda_{k+1} \|y\|^2 \leq y^\top \left(H - VV^\top H_\mu VV^\top \right) y \leq 5\lambda_{k+1} \|y\|^2 \\
\Rightarrow & - 10\lambda_{k+1} \|y\|^2 \leq y^\top \left(H - \tilde{H} \right) y \leq 0 \\
\Rightarrow & y^\top \tilde{H} y - 6\lambda_{k+1} \|y\|^2 \overset{(a)}{\leq} y^\top H y \overset{(b)}{\leq} y^\top \tilde{H} y
\end{aligned}
$$

Let us consider the $\overset{(a)}{\leq}$ case and have

$$
\begin{aligned}
& y^\top \tilde{H} y - 10\lambda_{k+1} \|y\|^2 \leq y^\top H y \\
\Rightarrow & y^\top \tilde{H} y \leq y^\top H y + 10\lambda_{k+1} \|y\|^2 \leq \left(1 + \frac{10\lambda_{k+1}}{\lambda_{\min}} \right) y \leq y^\top H y \\
\Rightarrow & \left(1 - \frac{10\lambda_{k+1}}{\lambda_{\min} + 10\lambda_{k+1}} \right) \tilde{H} \preceq H
\end{aligned}
$$

Therefore, we have

$$
\left(1 - \frac{10\lambda_{k+1}}{\lambda_{\min} + 10\lambda_{k+1}} \right) \tilde{H} \preceq H \preceq \tilde{H}.
$$

□

C.2 Proof of Theorem 4

Proof of Theorem 4. By Theorem 3, it takes $O\left(dk \cdot \log(\tan(U_k, V_0))\right)$ queries to construct the approximate Hessian. We also need $O(dk)$ queries to estimate gradient. By Theorem 1 and 3, we need $O\left(\frac{d\lambda_{k+1}}{b\tau} \log\left(\frac{1}{\epsilon}\right)\right)$ iterations to achieve an ϵ-accuracy. Hence, we have

$$
\begin{aligned}
Q(\epsilon) =& O\left(\frac{d\lambda_{k+1}}{b\tau} b \log\left(\frac{1}{\epsilon}\right) + \frac{d\lambda_{k+1}}{b\tau} \cdot dk \cdot \tan(U_k, V_0) \log\left(\frac{1}{\epsilon}\right) \right) \\
=& O\left(\frac{d\lambda_{k+1}}{\tau} \cdot \log(\tan(U_k, V_0)) \cdot \log\left(\frac{1}{\epsilon}\right) \right), \\
=& \tilde{O}\left(\frac{d\lambda_{k+1}}{\tau} \cdot \log\left(\frac{1}{\epsilon}\right) \right).
\end{aligned}
$$

□

Algorithm 4 Noisy Power Method.

```
**Input:** A positive definite matrix $H \in \mathbb{R}^{d \times d}$, orthonormal matrix $X \in \mathbb{R}^{d \times p}$, target rank $k$;
for $t = 0, \ldots$ until termination do
    $Y_t = HX_t + G_t$ for some noise matrix $G_t$;
    QR factorization: $Y_t = X_{t+1}R_{t+1}$, where $V_{t+1}$ consists of orthonormal columns.
end for
```

D Proof of Section 4

D.1 Proof of Lemma 4

Proof of Lemma 4. By the definition of $f_\mu(x)$, we have

$$\begin{aligned}\left\|\nabla^2 f_\mu(x) - \nabla^2 f(x)\right\| =& \left\|\nabla^2 \mathbb{E}\left[f(x+\mu u)\right] - \nabla^2 f(x)\right\| \\ =& \left\|\mathbb{E}\left[\nabla^2 f(x+\mu u)\right] - \nabla^2 f(x)\right\| \\ \le& \mathbb{E}\left[\left\|\nabla^2 f(x+\mu u) - \nabla^2 f(x)\right\|\right] \\ \le& \gamma\mu\mathbb{E}\left\|u\right\| \\ \le& \gamma\mu(d+1)^{1/2}.\end{aligned}$$

The first inequality is due to Jensen's inequality. The second inequality is by Eqn. (2.4). And the last inequality follows from Lemma 8. □

D.2 Proof of Lemma 5

Proof of Lemma 5. To get a convenient expression, we rewrite Eqn. (2.6) in another form by introducing a new integration variable $y = x + \mu u$:

$$f_\mu(x) = \frac{1}{\mu^d M}\int_{\mathbb{R}^d} f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right) dy.$$

Then, its gradient can be written as,

$$\nabla f_\mu(x) = \frac{1}{\mu^{d+2} M}\int_{E} f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right)(y-x)\, dy.$$

Then, we have

$$\begin{aligned}\nabla^2 f_\mu(x) =& -\frac{1}{\mu^{d+2}M}\int_E f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right) dy \\ &+ \frac{1}{\mu^{d+4}M}\int_E f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right)\cdot(y-x)(y-x)^\top dy \\ =& -\left[\frac{1}{\mu^2 M}\int_E f(x+\mu u)\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du\right] I_d + \frac{1}{\mu^2 M}\int_E f(x+\mu u)uu^\top\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du \\ =& \frac{1}{\mu^2 M}\int_E \left[f(x+\mu u) - f(x)\right] uu^\top\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du + \frac{1}{\mu^2}(f(x) - f_\mu(x))I_d. \qquad \text{(D.1)}\end{aligned}$$

Furthermore, $f_\mu(x)$ can also be defined as

$$f_\mu(x) = \frac{1}{M}\int_{\mathbb{R}^d} f(x-\mu u)\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du.$$

Similarly, by introducing $y = x - \mu u$, we have

$$f_\mu(x) = -\frac{1}{\mu^d M}\int_{\mathbb{R}^d} f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right) dy.$$

And its gradient is,

$$\nabla f_\mu(x) = -\frac{1}{\mu^{d+2} M}\int_{E} f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right)(y-x)\, dy.$$

Then, we have

$$\begin{aligned}
\nabla^2 f_\mu(x) =& \frac{1}{\mu^{d+2} M}\int_{E} f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right) dy \\
&- \frac{1}{\mu^{d+4} M}\int_{E} f(y)\ \exp\left(-\frac{1}{2\mu^2}\left\|y-x\right\|^2\right)\cdot (y-x)(y-x)^\top\, dy \\
=& -\left[\frac{1}{\mu^2 M}\int_E f(x-\mu u)\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du\right] I + \frac{1}{\mu^2 M}\int_E f(x-\mu u) u u^\top\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du \\
=& \frac{1}{\mu^2 M}\int_E \left[f(x-\mu u) - f(x)\right] u u^\top\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du + \frac{1}{\mu^2}(f(x) - f_\mu(x)) I_d. \qquad \text{(D.2)}
\end{aligned}$$

Combining the Eqn. (D.1) and (D.2), we have

$$\begin{aligned}
\nabla^2 f_\mu(x) =& \frac{1}{2}\left(\nabla^2 f_\mu(x) + \nabla^2 f_\mu(x)\right) \\
=& \frac{1}{M}\int_{\mathbb{R}^d} \frac{f(x+\mu u) + f(x-\mu u) - 2f(x)}{2\mu^2} u u^\top\ \exp\left(-\frac{1}{2}\left\|u\right\|^2\right) du + \frac{f(x) - f_\mu(x)}{\mu^2}\cdot I_d.
\end{aligned}$$

Therefore, we have

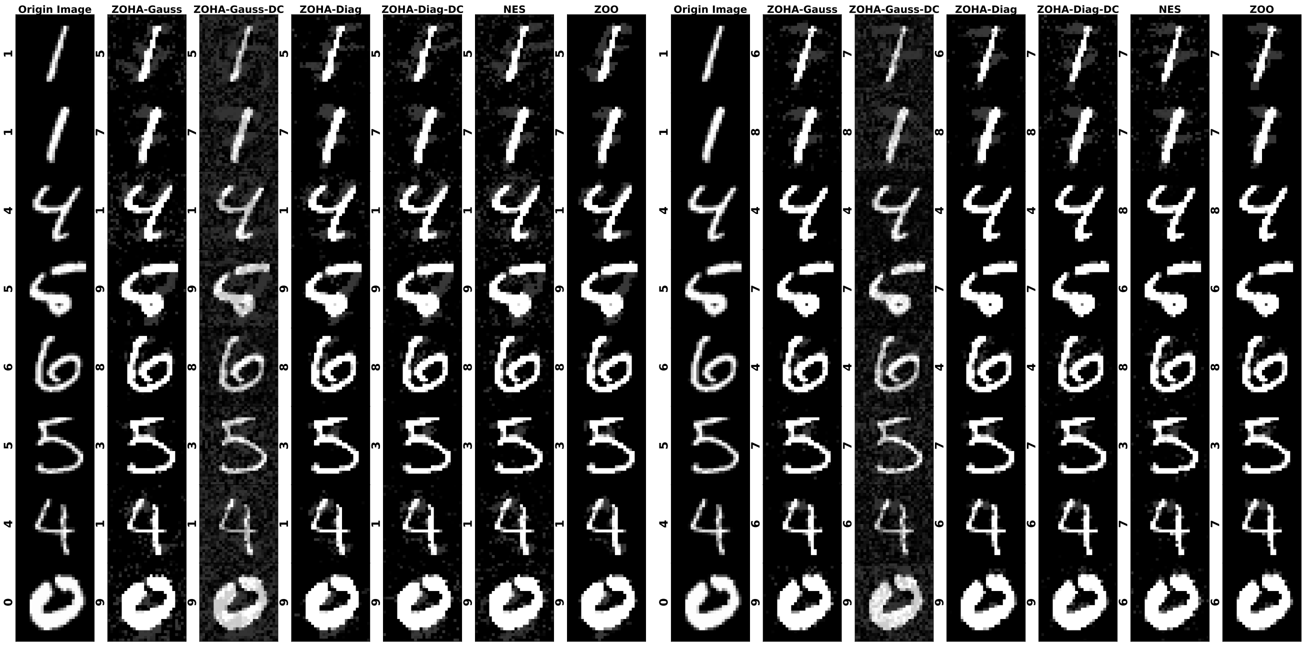
$$\mathbb{E}_u[\tilde{H}] = \nabla^2 f_\mu(x) + \left(\lambda - \frac{f(x) - f_\mu(x)}{\mu^2}\right)\cdot I_d.$$

The inequality $\nabla^2 f_\mu(x) \preceq \mathbb{E}_u[\tilde{H}]$ is because of $f_\mu(x) \geq f(x)$ which is implied by

$$f_\mu(x) = \mathbb{E}[f(x+\mu u)] \geq \mathbb{E}\left[f(x) + \mu\left\langle \nabla f(x), u\right\rangle\right] = f(x)$$

where the inequality is because of convexity of $f(x)$. □

E Visualization of Adversarial Attack



(a) Targeted

(b) Un-targeted

Figure 5: Adversarial examples on MNIST. The label is denoted on the left side.

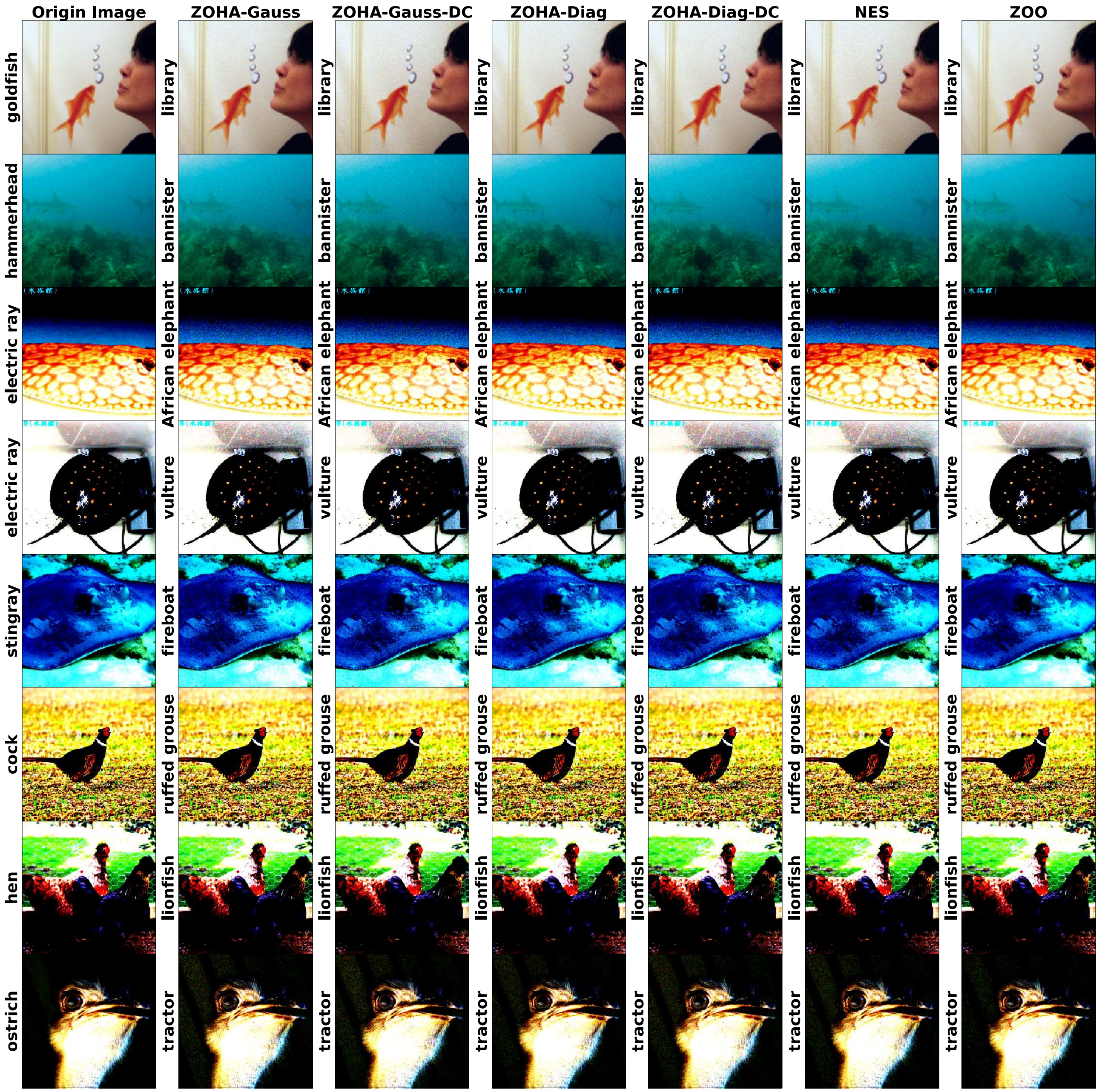


Figure 6: Targeted adversarial examples on ResNet50 and ImageNet. The label is denoted on the left side.

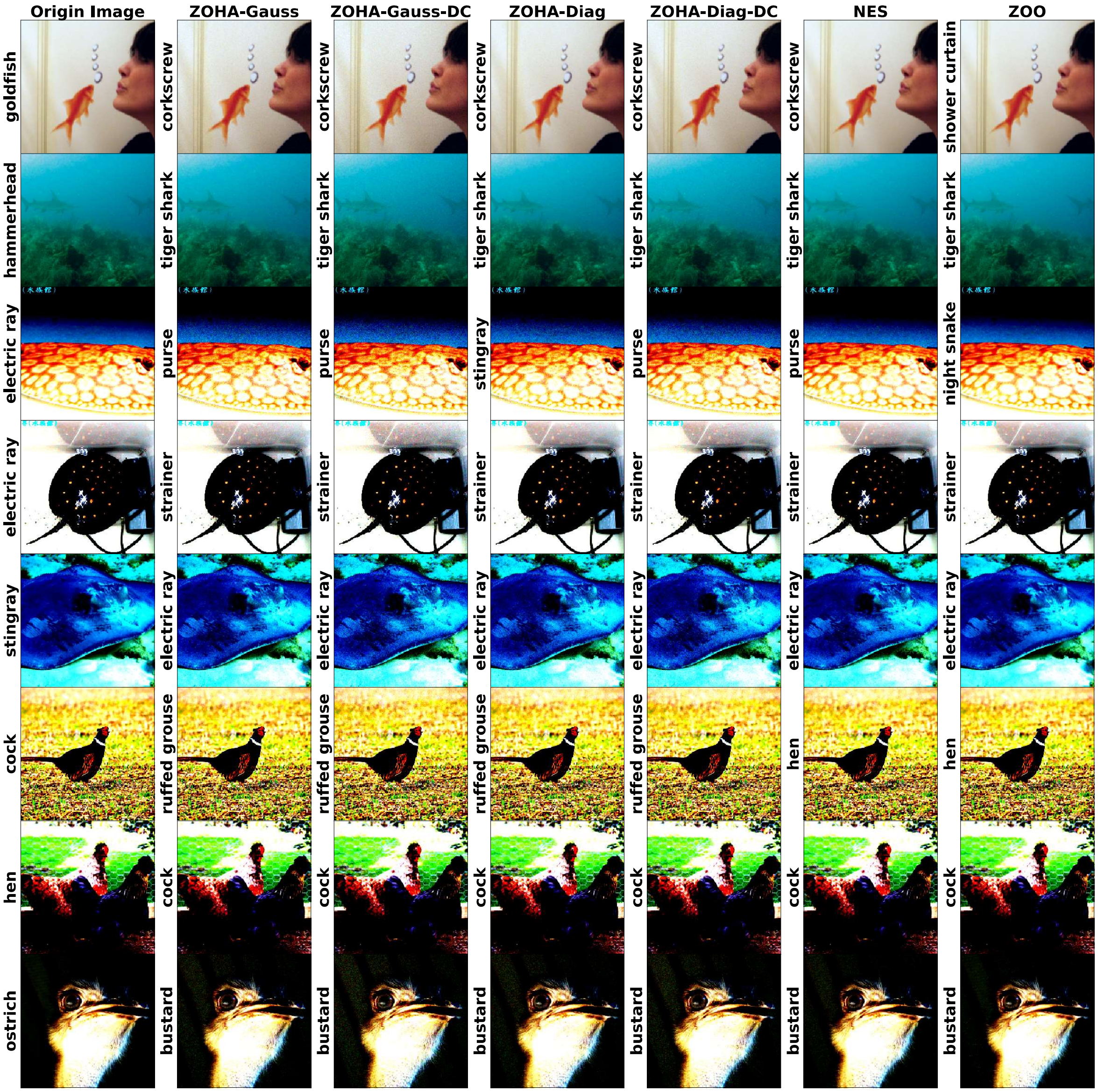


Figure 7: Un-targeted adversarial examples on ResNet50 and ImageNet. The label is denoted on the left side.