Uniformly Stable Algorithms for Adversarial Training and Beyond

Jiancong Xiao [* 1] **Jiawei Zhang** [* 2] **Zhi-Quan Luo** [3] **Asuman Ozdaglar** [2]

Abstract

In adversarial machine learning, neural networks suffer from a significant issue known as robust overfitting, where the robust test accuracy decreases over epochs (Rice et al., 2020). Recent research conducted by Xing et al. (2021); Xiao et al. (2022b) has focused on studying the uniform stability of adversarial training. Their investigations revealed that SGD-based adversarial training fails to exhibit uniform stability, and the derived stability bounds align with the observed phenomenon of robust overfitting in experiments. This finding motivates us to develop uniformly stable algorithms specifically tailored for adversarial training. To this aim, we introduce Moreau envelope-$\mathcal{A}$ (ME-$\mathcal{A}$), a variant of the Moreau Envelope-type algorithm. We employ a Moreau envelope function to reframe the original problem as a min-min problem, separating the non-strong convexity and non-smoothness of the adversarial loss. Then, this approach alternates between solving the inner and outer minimization problems to achieve uniform stability without incurring additional computational overhead. In practical scenarios, we demonstrate the efficacy of ME-$\mathcal{A}$ in mitigating the issue of robust overfitting. Beyond its application in adversarial training, this represents a fundamental result in uniform stability analysis, as ME-$\mathcal{A}$ is the first algorithm to exhibit uniform stability for weakly-convex, non-smooth problems.

1. Introduction

One of the interesting abilities of deep neural networks (DNNs) (Krizhevsky et al., 2012) is that they rarely suffer

---
[*]Equal contribution [1]University of Pennsylvania, PA, USA; [2]Massachusetts Institute of Technology, MA, USA; [3]The Chinese University of Hong Kong, Shenzhen, China. Correspondence to: Jiancong Xiao <jcxiao@upenn.edu>, Jiawei Zhang <jwzhang@mit.edu>.

from overfitting issues (Zhang et al., 2021). However, in the setting of adversarial training, this ability disappears, and overfitting becomes one of the most critical issues. Specifically, in a regular setting of SGD-based adversarial training shown in Figure 1, the robust test accuracy (orange line) starts to decrease after a particular epoch, while the robust training accuracy (blue line) continues to increase. This phenomenon is referred to as robust overfitting (Rice et al., 2020). It can be observed in experiments on common datasets such as SVHN, CIFAR-10/100.
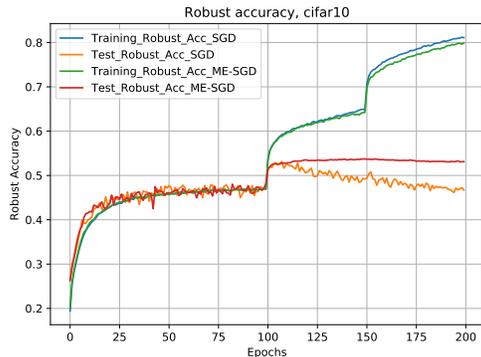
Figure 1. Demonstration of the occurrence of robust overfitting (orange line) and the mitigation of robust overfitting achieved by ME-$\mathcal{A}$ (red line).

Recent research has utilized uniform stability, a generalization measure in learning theory, to investigate this phenomenon (Xing et al., 2021; Xiao et al., 2022b). They have suggested that the non-smoothness of the adversarial loss may contribute to the issue of robust overfitting. Informally, uniform stability is the gap between the output parameters w of running an algorithm on two datasets S and S' that differ in at most one sample, denoted as $\|w(S) - w(S')\|$. In uniform stability analysis, assuming the standard training loss is non-convex and smooth, a well-known result given by (Hardt et al., 2016) is that applying stochastic gradient descent (SGD) to the standard loss yields uniform stability in $\mathcal{O}(T^q/n)$, where T represents the number of iterations, n is the number of samples, and $0 < q < 1$. However, the adversarial loss is non-smooth, even if we assume the standard loss is smooth. Consequently, the uniform stability bounds include an additional term in $\mathcal{O}(T^q \epsilon)$ (Xiao et al.,

Table 1. Comparison of uniform stability bounds of adversarial training. ME-$\mathcal{A}$ reduces the additional term in $\mathcal{O}(T^q \epsilon)$, which is a possible reason for robust overfitting. It also mitigates the robust overfitting issue in practice.

	Assumption (Std-loss)	Training	Algorithm	Stability Bound	No Overfitting
Hardt et al. (2016)	Nonconvex	Standard Training	SGD	$\mathcal{O}(T^q/n)$	✓
Xiao et al. (2022b)	Nonconvex	Adversarial Training	SGD	$\mathcal{O}(T^q \epsilon + T^q/n)$	✗
Ours	Nonconvex	Adversarial Training	**ME-$\mathcal{A}$**	$\mathcal{O}(T^q/n)$	✓

2022b), where ϵ is the attack intensity. The bound suggests that the robust test error increases as T grows, even when we have an infinite number of training samples ($n \to \infty$). Therefore, the derived uniform stability bounds align with the observations made in practical adversarial training.

This observation motivates us to develop uniformly stable algorithms for adversarial training in order to mitigate the issue of robust overfitting. Additionally, we define the adversarial loss as $h(w; z) = \max_{\|z - z'\| \le \epsilon} g(w; z)$, where z represents the sample, w is the training parameter, and g denotes the corresponding standard training loss. It has been proven that the additional term arises from the non-smoothness caused by the max operation in the adversarial loss (Xiao et al., 2022b). Consequently, our objective is to design uniformly stable algorithms specifically tailored for non-smooth optimization problems and then apply them to the adversarial training problem.

However, attaining uniform stability with non-smooth loss functions presents significant challenges. Common strategies in non-smooth optimization involve smoothing the loss function (*e.g.,* Moreau Yosida Smoothing (MYS) (Nesterov, 2005)). This approach is notably computationally inefficient when aiming for uniform stability, as proved by Bassily et al. (2019). In deep learning, this drawback is further magnified, and such methods are computationally intractable. Additionally, a notable study by Bassily et al. (2020) pointed out that SGD, a widely used optimization technique, does not ensure uniform stability for convex non-smooth problems.

Our approach involves utilizing the Moreau envelope function (Moreau, 1965), which is a classical tool employed in methods such as the proximal point method (PPM) and MYS, in a different way. Our approach is to separate the non-strong convexity and non-smoothness of the original convex non-smooth problem. Given training set S, the original problem (1.1):

$$\min_w \mathbb{E}_S h(w; z) \qquad (1.1)$$

is then reformulated into the min-min problem (1.2):

$$\min_u \min_w \mathbb{E}_S \left[h(w; z) + \frac{p}{2} \|w - u\|^2 \right]. \qquad (1.2)$$

Firstly, it is important to note that (1.2) is equivalent to (1.1) in terms of global solutions. Secondly, the inner problem ex-

hibits strong convexity and non-smoothness, while the outer problem is convex and smooth. Both of the two problems have at least one advantage for algorithm design.

Moreover, this analysis extends beyond convex problems to include non-convex ones as well. Building on the assumptions in Hardt et al. (2016); Xiao et al. (2022b) that the standard loss function $g(w; z)$ is non-convex and smooth, although the adversarial loss is both non-convex and non-smooth, it can still be demonstrated to be weakly convex. Therefore, ME-$\mathcal{A}$ can be applied in this setting.

The main results of this work are in three aspects.

1. **Algorithms for Adversarial Training.** Let $\mathcal{A}$ be a first-order algorithm used to solve the original problem (1.1), such as SGD or batch gradient descent (BGD). We introduce Moreau envelope-$\mathcal{A}$ (ME-$\mathcal{A}$), which alternates the application of $\mathcal{A}$ to the inner problem and GD to the outer problem of (1.2). We prove that ME-$\mathcal{A}$ achieves uniform stability for both the inner and outer problems, thereby achieving uniform stability for the entire problem without incurring additional computational overhead. The comparison of SGD and ME-$\mathcal{A}$ is provided in Table 1. ME-$\mathcal{A}$ improves over SGD in terms of uniform stability by reducing the term $\mathcal{O}(T^q \epsilon)$. In Figure 1 (red line), we demonstrate that ME-$\mathcal{A}$ effectively mitigates robust overfitting in practical scenarios.

2. **Understanding Adversarial Training.** In the previous studies of adversarial training, robust overfitting and sample complexity are usually considered to be related: DNNs tend to overfit adversarial examples, necessitating more samples to circumvent this issue (Schmidt et al., 2018; Rice et al., 2020). This paper provides us with further insights that robust generalization can be decomposed additively by robust overfitting and sample complexity, i.e.,

$$\underbrace{\text{Robust Generalization}} \le \underbrace{\text{Robust Overfitting}}_{\text{red line - orange line} \approx \mathcal{O}(T^q \epsilon)} + \underbrace{\text{Sample Complexity}}_{\text{blue line - red line} \approx \mathcal{O}(T^q/n)}.$$

Robust Overfitting. By employing ME-$\mathcal{A}$, the robust overfitting issue (in $\mathcal{O}(T^q \epsilon)$) is mitigated. DNNs fit the adversarial examples well, yet achieving the performance

ceiling (red line in Figure 1), within the constraints of the existing dataset size n. A widely used algorithm, SWA, plays a similar role as ME-$\mathcal{A}$ in adversarial training.

Sample Complexity. Considering the performance ceiling established at $\mathcal{O}(T^q/n)$, an increase in data volume is essential for enhancing this upper limit. Recent studies, such as those by Carmon et al. (2019) on pseudo-labeled data and Rebuffi et al. (2021) on generated data, have demonstrated their effectiveness in improving robust generalization. These findings lend support to our theoretical framework.

3. **Beyond Adversarial Training.** While our primary emphasis lies in adversarial training, we present a fundamental result in uniform stability analysis. ME-$\mathcal{A}$ is the first uniformly stable algorithm for weakly-convex, non-smooth problems, which is not accomplished by existing algorithms such as PPM and MYS. A comprehensive comparison is given in Section 5.

2. Related Work

Uniform Stability. The concept of stability can be traced back to the work of (Rogers & Wagner, 1978). In the context of statistical learning problems, it has been well developed through the analysis of algorithm-based generalization bounds (Bousquet & Elisseeff, 2002).

Smooth Cases. In smooth settings, Hardt et al. (2016) established strong bounds of stability. They demonstrated that several variants of SGD can simultaneously achieve uniform stability bounds in $\mathcal{O}(L^2 T\alpha/n)$ in convex settings and $\mathcal{O}(L^2 T^q/n)$ in non-convex settings, where α is the step size. This approach has been used in several studies to derive new generalization properties of SGD (Feldman & Vondrak, 2018; 2019). The work of (Chen et al., 2018) investigated the optimal trade-off between stability and convergence.

Proximal Methods. ME-$\mathcal{A}$ looks similar to PPM since they both use the Moreau envelope function. In smooth case, Yuan & Li (2023) have provided a thorough analysis of uniform stability for PPM. Additionally, Hardt et al. (2016) demonstrated that incorporating a proximal step following SGD steps does not degrade the uniform stability bound. In Section 5, we provide a more detailed discuss about the difference between ME-$\mathcal{A}$ and PPM.

Non-Smooth Cases. In adversarial training, Liu et al. (2020) proved that non-smoothness is an important issue, leading to bad robust accuracy. Bassily et al. (2020) investigated the stability of several variants of SGD on non-smooth loss functions. They demonstrated that the generalization bound contains an additional terms. Subsequent studies showed that some variants of SGD, such as pairwise-SGD (Yang et al., 2021) and Markov chain-SGD (Wang et al.,

2022), also possess this term. We list these work in Appendix C. Kanai et al. (2023) introduces the use of Entropy-SGD, a technique that applies SGLD to a smooth surrogate loss to enhance uniform stability. The study of (Lei, 2023) also investigated the topic of stability in non-convex non-smooth problems. They introduced a novel stability measure known as stability in gradient, which assesses the stability of non-convex problems. Remarkably, they also employed the Moreau envelope function, but for the definition of stability in gradient for non-differentiable function.

Uniform Convergence Analysis. Besides algorithmic generalization analysis, uniform convergence represents a different approach to generalization analysis in traditional learning theory. It offers generalization bounds for the function class with high probability, which is algorithm-independent. Uniform convergence analysis includes VC-dimension, Rademacher complexity, and Pac-Bayes analysis. Research by Cullina et al. (2018); Attias et al. (2022b;a) has established adversarial generalization bounds utilizing VC-dimension. For example, in finite adversarial examples cases, Attias et al. (2022b) provided the sample complexity of the generalization gap with respect to VC($\mathcal{H}$). Following that, Montasser et al. (2019) have proved that VC classes are robustly PAC-learnable only improperly, with respect to any arbitrary perturbation set, possibly of infinite size. Their approach relies on sample compression arguments, whereas uniform convergence does not hold. Regarding Rademacher complexity, robust generalization can be bounded by adversarial Rademacher complexity (Khim & Loh, 2018; Yin et al., 2019) for linear classifiers. It is extended to two-layer neural networks (Awasthi et al., 2020), FGSM attacks (Gao & Wang, 2021), and deep neural networks (Xiao et al., 2022a; Mustafa et al., 2022). Pac-Bayes analysis is another approach to provide norm-based control for generalization. Farnia et al. (2018) and Xiao et al. (2023) provided Pac-Bayesian bounds for adversarial generalization. Since these bounds are algorithm-independent, they cannot distinguish the generalization performance of different algorithms.

3. Preliminaries of Stability Analysis

Let $\mathcal{D}$ be an unknown distribution in the sample space $\mathcal{Z}$. Our goal is to find a model w with small population risk, defined as:

$$R_\mathcal{D}(w) = \mathbb{E}_{z\sim\mathcal{D}}h(w, z),$$

where $h(\cdot, \cdot)$ is the loss function which is possibly nonsmooth. Since we cannot get access to the objective $R_\mathcal{D}(w)$ directly due to the unknown distribution $\mathcal{D}$, we instead minimize the empirical risk built on a training dataset. Let $S = \{z_1, \ldots, z_n\} \sim \mathcal{D}^n$ be a sample dataset drawn i.i.d.

from $\mathcal{D}$. The empirical risk function is defined as:

$$R_S(w) = \frac{1}{n}\sum_{i=1}^{n} h(w, z_i).$$

Let $\bar{w}$ be the optimal solution of $R_S(w)$. Then, for the algorithm output $\hat{w} = A(S)$, we define the expected generalization gap as

$$\mathcal{E}_{gen} = \mathbb{E}_{S\sim\mathcal{D}^n, A}[R_\mathcal{D}(A(S)) - R_S(A(S))].$$

We define the the expected optimization gap as

$$\mathcal{E}_{opt} = \mathbb{E}_{S\sim\mathcal{D}^n, A}[R_S(A(S)) - R_S(\bar{w})].$$

To bound the generalization gap of a model $\hat{w} = A(S)$ trained by a randomized algorithm A, we employ the following notion of *uniform stability*.

Definition 3.1. A randomized algorithm A is ε-*uniformly stable* if for all data sets $S, S' \in \mathcal{Z}^n$ such that S and S' differ in at most one example, we have

$$\sup_z \mathbb{E}_A\left[h(A(S); z) - h(A(S'); z)\right] \le \varepsilon.$$

The following theorem shows that expected generalization gap can be attained from uniform stability.

Theorem 3.2 (Generalization in expectation (Hardt et al., 2016)). *Let A be ε-uniformly stable. Then, the expected generalization gap satisfies*

$$|\mathcal{E}_{gen}| = |\mathbb{E}_{S,A}[R_\mathcal{D}[A(S)] - R_S[A(S)]]| \le \varepsilon.$$

Hypothesis Class. As proved in (Xing et al., 2021; Xiao et al., 2022b), $h(w; z)$ is non-smooth even even though we assume that its standard counterpart is smooth. Therefore, we focus on non-smooth loss minimization. This class is denoted by $\mathcal{H}$ and is defined as follows:

$$\mathcal{H} = \{h : W \times \mathcal{Z} \to \mathbb{R} \mid L\text{-Lipschitz in } w, |W| = D_W\}.$$

In this paper, we explore both convex and non-convex settings. When the standard loss is convex, the adversarial loss is also convex. In cases where the standard loss is non-convex, the adversarial loss can still be demonstrated to be weakly convex, owing to the smoothness of the standard loss.

In experiments, we consider the following two losses for adversarial training.

Adversarial Loss. Let the loss function $h(w; z) = \max_{\|x-x'\|\le\epsilon} \ell(f_w(x'), y)$, where ϵ is the perturbation intensity. Here $f_w(\cdot)$ is a neural network parameterized by w, and $z = (x, y)$ is the input-label pair. If the neural networks are defined in a compact domain, *i.e.,* $\|x\| \le B, \forall x \in \mathcal{X}$, the loss function $h(w; z)$ is L-Lipschitz. It is shown that adversarial loss is L-Lipschitz given the standard loss is L-Lipschitz, and it is non-smooth even if the standard loss is smooth (Xiao et al., 2022b).

TRADES Loss. Let the loss function be

$$h(w; z) = \ell(f_w(x'), y) + \beta \max_{\|x-x'\|\le\epsilon} \ell(f_w(x'), f_w(x)),$$

where β is a hyperparameter (Zhang et al., 2019). Similar to the adversarial loss, the inner maximization problem induces the non-smoothness of the TRADES loss.

4. Moreau Envelope-$\mathcal{A}$

In this section, we introduce the algorithms, ME-$\mathcal{A}$[1], to achieve $\mathcal{O}(T^q/n)$-uniform stability for non-smooth loss minimization. Although our primary findings are framed within non-convex settings, we begin our discussion in convex settings to streamline the theoretical exposition.

4.1. Equivalent Problem

We use the Moreau envelope function to define the surrogate loss. Let

$$K(w, u; z) = h(w; z) + \frac{p}{2}\|w - u\|^2.$$

We can choose $p > 0$ to insure that $K(w, u; z)$ is strongly convex with respect to w. We define the Moreau envelope function of the empirical loss:

$$M(u; S) = \min_{w\in W} K(w, u; S) = \min_{w\in W}\frac{1}{n}\sum_{z\in S} K(w, u; z),$$

$$w(u; S) = \arg\min_{w\in W} K(w, u; S).$$

Employing the Moreau envelope function to the empirical loss (rather than the loss $h(w; z)$ as in MYS) is an important steps in our approach. We defer the comparison of ME-$\mathcal{A}$ and MYS to Section 5 to show why it is important. The following Lemma holds for the Moreau envelope function $M(u; S)$.

Lemma 4.1 (Equivalent Problem). *Assume that $h \in \mathcal{H}$. Let $p > 0$. Then, $\min_u M(u; S)$ has the same global solution set as $\min_w R_S(w)$.*

As $\min_u M(u; S)$ has the same global solutions as $\min_w R_S(w)$, the original problem is equivalent to the problem of minimizing $M(u; S)$, *i.e.,*

$$\min_w R_S(w)$$
$$\Leftrightarrow \quad \min_u \min_w \frac{1}{n}\sum_{z\in S} K(w, u; z).$$

Therefore, we can alternatively minimize the inner and outer problems to find the solutions of the original problem. Such decomposition allows us to disentangle the non-strong convexity and non-smoothness of the original problem. Specifically, the inner problem is strongly-convex and non-smooth,

---

[1]The initial version of the algorithm is refer to as Smoothed-SGDmax (Xiao et al., 2022e).

and the outer problem is convex and smooth. We can achieve uniform stability for both of the two problems. Below we provide the details.

Uniform Stability of Inner Minimization.   Based on the definition, $K(w, u; z)$ is strongly-convex. The inner minimization problem is a strongly-convex, non-smooth problem. The following Lemma shows that the minimizer of a strongly-convex problem is $\mathcal{O}(1/pn)$-uniformly stable.

Lemma 4.2 (Uniform Stability of Inner Minimization). *Assume that $h \in \mathcal{H}$. Let $p > 0$. For neighbouring S and S', we have*

$$\|w(u; S) - w(u; S')\| \leq 2L/(np).$$

Uniform Stability of Outer Minimization.   The outer problem is a p-smooth convex problem. It is due to the following properties of the Moreau envelope function $M(u, S)$.

Lemma 4.3 (Smoothness and Convexity of Moreau Envelops Functions). *Assume that $h \in \mathcal{H}$. Let $p > 0$. Then, $M(u; S)$ satisfies*

1. The gradient of $M(u; S)$ is $\nabla_u M(u; S) = p(u - w(u; S))$.

2. $M(u; S)$ is convex in u.

3. $M(u; S)$ is p-gradient Lipschitz continuous.

The proof of Lemma 4.3 is due to (Rockafellar, 1976) and also provided in Appendix. Based on the p-smooth property, the $\mathcal{O}(T^q/n)$-uniform stability of the outer problem is achieved by running gradient descent on $M(u; S)$.

Lemma 4.4 (Uniform Stability of Outer Minimization). *Assume h is a convex, L-Lipschitz function. Let $u^T(S)$ and $u^T(S')$ be the outputs of running GD on $M(u; S)$ and $M(u; S')$, respectively, with fixed stepsize $\alpha \leq 1/p$ for T steps. Then, the generalization gap satisfies*

$$\|u^T(S) - u^T(S')\| \leq \left(\frac{2LT\alpha}{n}\right).$$

Remark.   This upper bound is as tight as the result of running SGD on smooth convex finite-sum problems (Hardt et al., 2016). It is worth noticing that Lemma 4.4 is not a corollary of the result of (Hardt et al., 2016), because $M(u, S)$ is not in the form of finite-sum. The proof of Lemma 4.4 is based on Lemmas 4.2 and 4.3. It is deferred to Appendix.

4.2. Uniform Stability of ME-$\mathcal{A}$

Let $\mathcal{A}$ be a first-order (stochastic) algorithm for the original problem $R_S(w)$, *i.e.,* (BGD, SGD. In the equivalent problem, we directly apply $\mathcal{A}$ to the inner problem. Based on Lemma 4.4, we apply GD to the outer problem. The algorithm is provided in Algorithm 1).

Algorithm 1 Moreau Envelope-$\mathcal{A}$

1: Initialize w^0, u^0;
2: Choose stepsize $0 < \alpha_t \leq 1/p$ (or let $\tau_t = 1 - \alpha_t p$, $0 \leq \tau_t < 1$);
3: **for** $t = 0, 1, 2, \ldots, T$ **do**
4:    Let $w_0^t = w^t$;
5:    **for** $s = 0, 1, 2, \cdots, N$ **do**
6:       $w_{s+1}^t = \mathcal{A}(K(w_s^t, u^t; S))$;
7:    **end for**
8:    $w^{t+1} = w_N^t$;
9:    $u^{t+1} = u^t + \alpha_t p(w^{t+1} - u^t)$ (or $u^{t+1} = \tau_t u^t + (1 - \tau_t) w^{t+1}$);
10: **end for**

To develop the uniform stability of ME-$\mathcal{A}$, we first assume the optimization error of the inner problem. Let

$$\|w_N^t - w(u^t; S)\| \leq \varepsilon(\mathcal{A}),$$

i.e., the distance between the output of the inner minimization problem w_N^t and the optimal minimizer $w(u^t; S)$ in each iterations t is at most $\varepsilon(\mathcal{A})$. For example, the convergence rate of running SGD on strongly convex, non-smooth minimization problem is $\mathcal{O}(1/N)$ (Nemirovski & Yudin, 1983).

Theorem 4.5 (Generalization bound of ME-$\mathcal{A}$ in Convex Case). *Assume that h is convex and L-Lipschitz. Suppose we run ME-$\mathcal{A}$ with stepsize $\alpha_t \leq 1/p$ for T steps. The generalization gap satisfies*

$$\mathcal{E}_{gen} \leq L\left(\frac{2L}{n} + 2p\varepsilon(\mathcal{A})\right) \sum_{t=1}^{T} \alpha_t.$$

Furthermore, if algorithm $\mathcal{A}$ satisfies $\varepsilon(\mathcal{A}) = \mathcal{O}(1/pn)$, the generalization gap satisfies

$$\mathcal{E}_{gen} \leq \mathcal{O}\left(\frac{2L^2}{n}\right) \sum_{t=1}^{T} \alpha_t.$$

Remark: In strongly-convex, non-smooth problems, an error bound of $\varepsilon(\mathcal{A}) = \mathcal{O}(1/pn)$ can be achieved by $\mathcal{A}$=SGD with a diminishing stepsize (Nemirovski & Yudin, 1983). For further discussion, see Appendix. When the stepsize is fixed to $\alpha_t = \alpha$, it is shown that Algorithm 1 achieves a generalization bound of $\mathcal{O}(T\alpha/n)$.

4.3. Non-Convex Case

Our principal contribution is focused on non-convex settings, encompassing a function class that extends beyond the limitations of convex scenarios.

In this setting, it seems that nothing can be done since the adversarial loss is both non-smooth and non-convex. Fortunately, the smoothness of the standard loss guarantee that the adversarial loss is weakly convex defined as followed.

Definition 4.6. Let $l > 0$. A function is said to be $-l$-weakly convex if $\forall x$, $f(x) + l\|x\|^2/2$ is convex in x.

This can be attributed to the following reasons: If the standard loss is smooth, meaning it has a gradient that is Lipschitz continuous, then the standard loss exhibits both upper and lower curvature. The presence of lower curvature implies that the standard loss is weakly convex, as lower curvature is equivalent to weak convexity. Furthermore, when the maximum operation is applied to the standard loss to obtain the adversarial loss, the resulting function retains this property of weak convexity. For a more detailed discussion, see Davis & Drusvyatskiy (2019).

In this case, we require $p > l$ such that $M(u; S)$ is strongly convex. Firstly, we extend Lemma 4.1 to 4.3 to weakly convex cases, which are Lemma A.1 to A.2 presented in Appendix. Based on the Lemma, we can derive the stability-based generalization bounds for ME-$\mathcal{A}$ in weakly-convex case.

Theorem 4.7 (Generalization bound of ME-$\mathcal{A}$ in Weakly-Convex Case). *Assume that h is a weakly-convex, L-Lipschitz function. Suppose we run ME-$\mathcal{A}$ with diminishing stepsize $\alpha \leq 1/\beta t$ for T steps, where $\beta = \max\{p, pl/(p - l)\}$. Then, the generalization gap satisfies*

$$\mathcal{E}_{gen} \leq \mathcal{O}\left(\frac{2L^2 T^q}{\min\{l, p - l\}n} + \varepsilon(\mathcal{A})2LpT^q/\beta \right), \quad (4.1)$$

where $q = \beta c < 1$. Furthermore, if algorithm $\mathcal{A}$ satisfies $\varepsilon(\mathcal{A}) = \mathcal{O}(1/(p - l)n)$, the generalization gap satisfies

$$\mathcal{E}_{gen} \leq \mathcal{O}\left(\frac{2L^2 T^q}{\min\{l, p - l\}n} \right). \quad (4.2)$$

Notice that existing uniform stability algorithms on non-smooth loss requires convexity assumption. A main benefit of ME-$\mathcal{A}$ is that the algorithm can achieve $\mathcal{O}(T^q/n)$-uniform stability in weakly-convex cases. For comparison, by applying SGD to adversarial training, the robust generalization gap satisfies $\mathcal{E}_{gen} \leq \mathcal{O}(2LT^q\epsilon + \frac{2L^2 T^q}{n})$.

4.4. Convex Risk Minimization

Now, we turn to the excess risk minimization analysis of ME-$\mathcal{A}$ in convex case. The excess risk can be decomposed to the sum of optimization and generalization error, *i.e.*,

Excess risk $= R_{\mathcal{D}}(w) - \min_{w \in W} R_{\mathcal{D}}(w) \leq \mathcal{E}_{opt} + \mathcal{E}_{gen}$.

Theorem 4.8 (Optimization error of ME-$\mathcal{A}$ in Convex Case). *Assume that h is a convex, L-Lipschitz function. Suppose we run ME-$\mathcal{A}$ with stepsize $\alpha \leq 1/p$ for T steps. Then, the optimization error satisfies*

$$\mathcal{E}_{opt} \leq \mathcal{O}\left(\frac{\|u_0 - u^*\|^2}{2\alpha T} + \varepsilon(\mathcal{A})p\frac{\sum_{t=1}^{T} \|u^t - u^*\|^2}{T} \right)$$

$$= \mathcal{O}\left(\frac{D_0}{2\alpha T} + \varepsilon(\mathcal{A})pD_1 \right).$$

Furthermore, if algorithm $\mathcal{A}$ satisfies $\varepsilon(\mathcal{A}) = \mathcal{O}(1/p\alpha T)$, the optimization gap satisfies $\mathcal{E}_{opt} \leq \mathcal{O}\left(\frac{1}{T\alpha}\right)$.

A minimax lower bound of the excess risk is given in (Nemirovski & Yudin, 1983): $\min_w \max_{\mathcal{D}} \mathbb{E}_S[R_{\mathcal{D}}(w) - \min_{w \in W} R_{\mathcal{D}}(w)] \geq \Omega(LD_W/\sqrt{n})$. By setting $T\alpha = \mathcal{O}(\sqrt{n})$, ME-$\mathcal{A}$ achieves the optimal excess risk with respect to T and α, resulting in Excess risk $\leq \mathcal{E}_{opt} + \mathcal{E}_{gen} \leq \mathcal{O}(\sqrt{1/n})$. This means that ME-$\mathcal{A}$, with an optimal stopping criterion, achieves the minimax lower bound in $\Omega(1/\sqrt{n})$ for excess risk. On the other hand, by combining the lower bound of excess risk and the upper bound of optimization error, we can obtain a lower bound in $\Omega(T\alpha/n)$ for the generalization gap.

Iteration Complexity. To avoid introducing additional computational cost, we hope the number of outer iterations $T = \mathcal{O}(1)$. Then, $\alpha = \mathcal{O}(\sqrt{n})$. Based on the condition $\alpha \leq 1/p$, we should have $p \leq \mathcal{O}(1/\sqrt{n})$. In this setting, the iteration complexity of ME-$\mathcal{A}$ is the same as that of $\mathcal{A}$. For example, Let $\mathcal{A}=$ SGD, based on Theorem 4.5 and 4.8, it is required $\varepsilon(\mathcal{A}) = \mathcal{O}(1/pn)$ for achieving uniform stability and $\varepsilon(\mathcal{A}) = \mathcal{O}(1/p\alpha T) = \mathcal{O}(1/p\sqrt{n})$ for achieving $\mathcal{O}(1/T\alpha)$ optimization error. In Theorem B.1, if we assume $D_W \leq \mathcal{O}(\sqrt{n})$, $N = n^2$, we have $\varepsilon(\mathcal{A}) = \mathcal{O}(1/pn)$. In this case, it is required $\mathcal{O}(n^2)$ inner iterations and $\mathcal{O}(1)$ outer iterations. The total iteration complexity is $\mathcal{O}(n^2)$.

It is important to note that this rate cannot be further reduced for non-smooth problems (Bousquet & Elisseeff, 2002). In the strongly convex non-smooth case, achieving uniform stability also requires $\mathcal{O}(n^2)$ iterations. Consequently, the iteration complexity of both ME-$\mathcal{A}$ (for convex problems) and $\mathcal{A}$ (for strongly convex problems) is $\mathcal{O}(n^2)$.

To conclude this section, we consider the performance of ME-$\mathcal{A}$ in smooth setting. In this case, applying first order algorithms to the inner problem have a linear convergence rate. The iteration complexity is $\mathcal{O}(n)$. It matches the rate of running SGD on smooth problem (Hardt et al., 2016). Theoretically, the iteration complexity of ME-$\mathcal{A}$ and $\mathcal{A}$ is the same. In practice, let $\mathcal{A}$ be the update rule of a first-order multi-epochs algorithm. Then, we can choose T to be the number of epochs and N to be the number of iterations in each epochs. ME-$\mathcal{A}$ has the same computational cost as $\mathcal{A}$.

5. Comparison of ME-$\mathcal{A}$ with Existing Algorithms

ME-$\mathcal{A}$ looks similar to but essentially different from some existing algorithms. It is necessary to compare ME-$\mathcal{A}$ with these similar algorithms in detail and the summary of the comparison is provided in Table 2.

Stochastic Weight Averaging (SWA). SWA (Izmailov et al., 2018) (or moving averaging in different literatures) suggests using the weighted average of the iterates rather than the final one for inference. The update rules of SWA is $u^{t+1} = \tau^t u^t + (1 - \tau^t)w^{t+1}$. By Simply applying the analytical tools in the work of (Bassily et al., 2020), we can see that SWA is not guarantee to be uniformly stable. On the other hands, SWA can be regarded as ME-$\mathcal{A}$ when $p \to 0$. In Algorithm 1, if we denote $\tau^t = 1 - \alpha_t p$, Step 9 can be view as a weight averaging step. In Thmeorem 4.5, it is required that $0 < \alpha_t \leq 1/p$. Then, $0 \leq \tau^t = (1 - \alpha_t p) < 1$. Therefore, by fixing $\alpha_t p$ to be constant and letting $p \to 0$, ME-$\mathcal{A}$ is reduced to SWA. Our analysis provides an understanding of the generalization ability of SWA.

Empirical Risk Minimization (ERM). ERM (or weight decay in different literatures) is a technique used to regularize the empirical loss by adding a ℓ_2 regularization term, *i.e.,* the loss function is $h(w; z) + p\|w\|^2/2$. If we replace Step 9 with $u = 0$ in Algorithm 1, it is reduced to ERM. In convex case, the regularized loss becomes strongly convex and ensures uniform stability (Bousquet & Elisseeff, 2002). Feldman et al. (2020) introduced a variant of ERM called Phase-ERM and showed that it achieves uniform stability in $\mathcal{O}(n^2)$ steps.

However, such theoretical guarantee cannot be extended to weakly-convex problems. This limitation arises because the regularization term changes the global solutions of the problems, necessitating that the parameter p cannot be too large. In weakly-convex case, we may not have $p > l$, implying that $h(w; z) + p\|w\|^2/2$ may not exhibit strong convexity. As a result, ERM is not guaranteed to be uniformly stable in weakly-convex scenario.

5.1. Comparison of Moreau Envelope-Type Algorithms

Moreau Yosida Smoothing. MYS is originally proposed to solve non-smooth convex optimization problems (Nesterov, 2005). It uses the Moreau envelope function to smooth the non-smooth loss $h(w; z)$. Then, algorithms can be applied to the smooth surrogate. The loss of MYS is

$$\min_u \mathbb{E}_S \min_w \left[h(w; z) + \frac{p}{2}\|w - u\|^2 \right].$$

The uniform stability analysis of MYS is provided in (Bassily et al. (2019), cf. Theorem 4.4). It is shown that $\mathcal{O}(n^{4.5})$ steps are required to achieve uniform stability, which is inefficient. In deep learning, memorization cost is another issue. It is required to store n different networks (or (n/batch size) in BGD settings) in memory, which is intractable. Additionally, it is worth notice that the theoretical analysis of MYS and ME-$\mathcal{A}$ are different. The analysis of MYS use standard tools of applying SGD to finite sum smooth problems.

Proximal Point Methods. PPM uses the proximal operator $w(u)$ to be the update rule, *i.e.,* $u^{t+1} = w(u^t)$. Notice that it is equivalent to apply GD to the Moreau envelope function $M(u; S)$ with constant stepsize $\alpha_t = 1/p$ (or $\tau = 0$). Therefore, PPM can be regarded as a special case of ME-$\mathcal{A}$ ($\alpha = 1/p$, $\mathcal{A}$=global minimizer). In convex case, a by-product of our results is that PPM achieves uniform stability for non-smooth loss minimization. However, PPM is not guaranteed to be uniformly stable in weakly-convex case. In Theorem 4.7, diminishing stepsize is required for achieving $\mathcal{O}(T^q/n)$-uniform stability, but PPM is equivalent to GD with fixed stepsize.

6. Experiments

We focus on $\mathcal{A}$=SGD in experiments to verify the theoretical results. We start from a toy experiment that perfectly matches the theory. Let the L_1 loss be $h(w, z) = \|w - z\|_1$. It is 1-Lipschitz, since $|h(w_1, z) - h(w_2, z)| = \|\|w_1 - z\|_1 - \|w_2 - z\|_1\| \leq \|w_1 - w_2\|_1$, for all z. It is non-smooth due to the ℓ_1-norm. Therefore, L_1-loss $\in \mathcal{H}$.

Let the true distribution be a 10-dimensions Gaussian distribution, $\mathcal{D} = \mathcal{N}(0, I)$. We sample 10 to 100000 data to train w with SGD and ME-SGD. The results are shown in Figure 2 (a). When the number of samples increases, the generalization gap induced by SGD does not converge to zero. While using ME-SGD, the gap converges to 0 fastly.

6.1. Mitigating Robust Overfitting in $\mathcal{O}(T^q \epsilon)$

Next, we turn our focus on adversarial training[2]. We first study the robust overfitting issue. We mainly adopt the hyper-parameter settings of adversarial training in the work of (Gowal et al., 2020). Weight decay is set to be 5×10^{-4}. Based on Theorem 4.7, the step size α_t of updating u is set to be $1/pt$, then $\tau_t = 1 - \alpha_t p = (t - 1)/t$. Interestingly, this theoretical-driven stepsize is consistent with the default setting of SWA stepsize used in practice. Ablation studies for the choice of p and the choice of τ_t are provided in Appendix C.

To have a first glance at how ME-$\mathcal{A}$ mitigates robust overfitting, we consider the training procedure against ℓ_∞-PGD attacks (Madry et al., 2018a) on SVHN, CIFAR-10, and

---

[2]Code is publicly available at https://github.com/JiancongXiao/Moreau-Envelope-SGD.

Table 2. Comparison of ME-$\mathcal{A}$ with SWA, ERM, MYS and PPM on non-smooth minimization problem. Only ME-$\mathcal{A}$ achieves uniform stability for weakly-convex non-smooth problems.

Algorithms	Types	Difference to ME-$\mathcal{A}$	Convex		Weakly-Convex
			Stability	Complexity	Stability
SWA	-	$p = 0, \tau > 0$	✗	-	✗
ERM	Regularization	$u = 0$	✓	-	✗
(Phase)-ERM		$u = 0, p \to 0$	✓	$\mathcal{O}(n^2)$	✗
MYS	Moreau Envelope	$\mathbb{E}_S \min_w[\cdot] \Rightarrow \min_w \mathbb{E}_S[\cdot]$	✓	$\mathcal{O}(n^{4.5})$	✗
PPM (Our proof)		$\tau = 0$	✓	$\mathcal{O}(n^2)$	✗
ME-$\mathcal{A}$ (Ours)		N/A	✓	$\mathcal{O}(n^2)$	✓
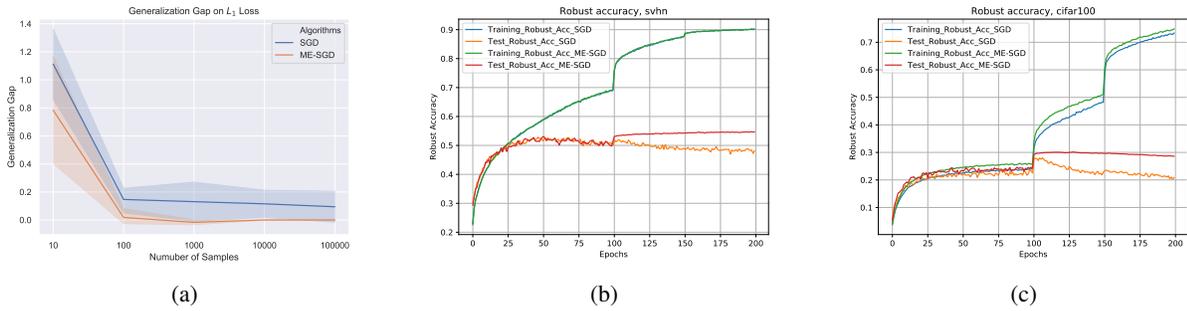
Figure 2. (a): Comparison of generalization gap induced by SGD and ME-SGD for the toy example. (b) and (c): Robust test accuracy of adversarial training using SGD and ME-$\mathcal{A}$ on SVHN, and CFAR-100, respectively.

CIFAR-100. For the attack algorithms, we use $\epsilon = 8/255$. The attack step size is set to be $\epsilon/4$. We use piece-wise learning rates, which are equal to $0.1, 0.01, 0.001$ for epochs 1 to 100, 101 to 150, and 151 to 200, respectively.

The experiments on CIFAR-10 is provided in Introduction. The experiments on SVHN and CIFAR-100 are provided in Figure 2 (b) and (c). It is shown that adversarially-trained models suffer from severe overfitting issues (Rice et al., 2020). From the perspective of uniform stability, it might be due to the additional term $\mathcal{O}(T^q \epsilon)$ induced by SGD. For SGD, the robust test accuracy starts to decrease at around the 100th epoch, which is a typical phenomenon of robust overfitting.

By employing ME-$\mathcal{A}$, we significantly mitigate the issue of robust overfitting. Our experiments on SVHN and CIFAR-10/100 demonstrate that the robust test accuracy no longer diminishes. With this approach, DNNs fit the training adversarial examples well, reaching a performance ceiling that is constrained by the size of the existing dataset, denoted as n.

6.2. Sample Complexity in $\mathcal{O}(T^q/n)$

Secondly, we examine the sample complexity characterized by $\mathcal{O}(T^q/n)$ as detailed in Theorem 4.7. Considering that

CIFAR-10 encompasses only 50,000 training samples, we utilize an additional pseudo-label dataset as introduced by Carmon et al. (2019) to analyze the sample complexity. Employing a greater proportion of pseudo-label data serves as a proxy for increasing the training dataset size. In Figure 3, we illustrate both the adversarial generalization gap (a) and the robust test accuracy (b). It is observed that by increasing the number of training data, denoted as n, the robust generalization gap decreases, subsequently enhancing the robust test accuracy.

Additivity of $\mathcal{O}(T^q \epsilon)$ and $\mathcal{O}(T^q/n)$. In Table 3, we study the interaction between applying ME-$\mathcal{A}$ and adding additional data. The baseline performance of WideResNet-28×10 on CIFAR-10 is reported in (Gowal et al., 2020). We adopt the AutoAttack (Croce & Hein, 2020), which is a collection of four attacks in default settings, to evaluate the performance of robust test errors. When applying the TRADES loss, the effect of ME-$\mathcal{A}$ is approximately 3%, regardless of the presence of additional data. The benefit of incorporating extra data contributes around 8% improvement, applicable to both SGD and ME-$\mathcal{A}$. A similar trend is observed with adversarial loss, with a small difference being that ME-$\mathcal{A}$ exerts a slightly lesser impact on adversarial loss, reducing to about 1%. These experiments support the

(a)

(b)
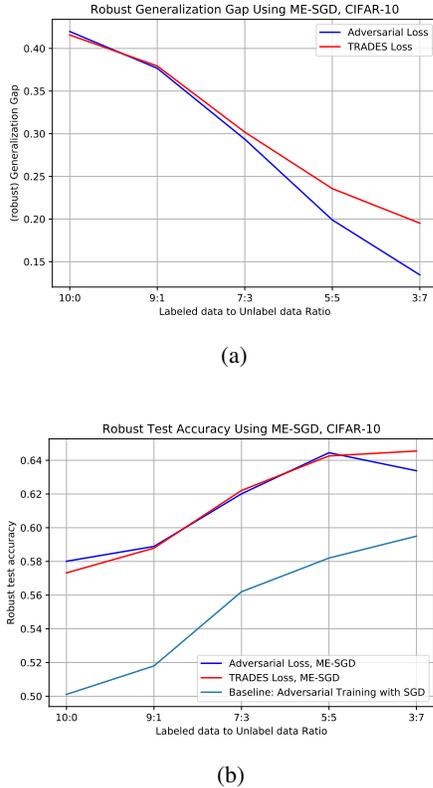
Figure 3. Robust generalization gap and robust test accuracy in the experiments of SGD and ME-$\mathcal{A}$.

assertion that there is an additive relationship between robust overfitting and sample complexity. The enhancements gained from reducing robust overfitting and from addressing sample complexity can be cumulatively superimposed.

6.3. Discussion of Exisitng Algorithms

In our final discussion, we study the empirical performance of methods listed in Table 2. ERM, essentially weight decay, is already a common practice in deep learning. MYS, however, is not implementable due to memory constraints. Regarding PPM, its experimentation is part of the ablation study on the parameter τ provided in Appendix C.2. When $\tau = 0$, its performance similar to that of SGD. Notably, none of these methods can mitigate robust overfitting. In contrast, SWA demonstrates practical improvements in robust generalization, achieving robust test accuracy comparable to ME-$\mathcal{A}$, as exemplified in Gowal et al. (2020). The similarity between SWA and ME-$\mathcal{A}$ suggests that our theory may offer insights into SWA, particularly when considering $p \to 0$. However, it is important to note that while SWA shows practical promise, it lacks provable results, unlike ME-$\mathcal{A}$, which is theoretically underpinned. This theoretical assurance is a significant advantage of ME-$\mathcal{A}$ over SWA.

Table 3. Robust test accuracy on adversarial loss and TRADES loss. The term '+data' means addition unlabeled data with labeled to unlabeled data ratio 3:7 is added to the training set. The notion ↓ means the improvement mainly comes from reducing the underlying term.

Algorithm	AutoAttack		Stability
	AT Loss	TRADES	
SGD	$50.80_{\pm0.23\%}$	51.91%	$\mathcal{O}(T^q\epsilon + T^q/n)$
+data	$58.41_{\pm0.25\%}$	59.45%	$\mathcal{O}(T^q\epsilon + \underline{T^q/n} \downarrow)$
ME-SGD	$51.66_{\pm0.16\%}$	$55.23_{\pm0.19\%}$	$\mathcal{O}(\underline{T^q\epsilon} \downarrow +T^q/n)$
+data	$59.14_{\pm0.18\%}$	$62.76_{\pm0.15\%}$	$\mathcal{O}(T^q\epsilon + T^q/n \downarrow)$

Recent studies, including (Rebuffi et al., 2021), indicate that robust generalization can be further enhanced by generating additional data using diffusion models. In our theoretical framework, the use of both pseudo-labeled data and generated data serves to optimize the terms $\mathcal{O}(T^q/n)$. In Table 4, we classify key existing methods for robust generalization into two distinct categories. This categorization helps validate our theoretical model.

Table 4. Existing approaches for improving adversarially robust generalization.

Algorithms	Types	Empirical	Theoretical
SWA	Robust Overfitting	✓	✗
ME-$\mathcal{A}$		✓	✓
Pseudo Labeled Data	Sample Complexity	✓	
Generated Data			✓

7. Conclusion

In this paper, we introduce ME-$\mathcal{A}$, an approach designed to achieve uniform stability for adversarial training under the assumption that the standard loss function is non-convex and smooth. The primary objective of ME-$\mathcal{A}$ is to mitigate robust overfitting, a phenomenon commonly observed in practice.

One of the limitations of our analysis is the assumption of $-l$-weak convexity of the adversarial loss function, or equivalently, the l-gradient Lipschitz assumption of the standard loss function. In practice, this assumption is reasonable for neural network training. If the loss function is differentiable and the training trajectory remains within a bounded region, it ensures that the loss function exhibits l-smoothness along the training trajectory. However, the main difficulty lies in estimating the value of l. Furthermore, designing new algorithms that are uniformly stable for non-convex and non-smooth problems remains a challenging task. In summary, we anticipate that ME-$\mathcal{A}$ will serve as an inspiration for the development of novel algorithms in deep learning.

Acknowledgements

Impact Statement

This paper is primarily theoretical in nature, focusing on complex concepts and in-depth analysis. The potential impact lies in the possibility that our theoretical findings could inform future practical applications and advancements in the field. This could include improved algorithms, enhanced understanding of neural network behaviors, or even influencing the development of more robust systems in various technological domains. Such impacts, while emerging indirectly from our theoretical work, hold the promise of contributing significantly to both the academic community and real-world applications over time.

References

Attias, I., Hanneke, S., and Mansour, Y. A characterization of semi-supervised adversarially robust pac learnability. *Advances in Neural Information Processing Systems*, 35: 23646–23659, 2022a.

Attias, I., Kontorovich, A., and Mansour, Y. Improved generalization bounds for adversarially robust learning. *Journal of Machine Learning Research*, 23(175):1–31, 2022b.

Awasthi, P., Frank, N., and Mohri, M. Adversarial learning guarantees for linear hypotheses and neural networks. In *International Conference on Machine Learning*, pp. 431–441. PMLR, 2020.

Bassily, R., Feldman, V., Talwar, K., and Guha Thakurta, A. Private stochastic convex optimization with optimal rates. *Advances in neural information processing systems*, 32, 2019.

Bassily, R., Feldman, V., Guzmán, C., and Talwar, K. Stability of stochastic gradient descent on nonsmooth convex losses. *Advances in Neural Information Processing Systems*, 33:4381–4391, 2020.

Bousquet, O. and Elisseeff, A. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. IEEE, 2017.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. S. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems*, pp. 11190–11201, 2019.

Chen, T., Zhang, Z., Liu, S., Chang, S., and Wang, Z. Robust overfitting may be mitigated by properly learned smoothening. In *International Conference on Learning Representations*, 2020.

Chen, Y., Jin, C., and Yu, B. Stability and convergence trade-off of iterative optimization algorithms. *arXiv preprint arXiv:1804.01619*, 2018.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pp. 2206–2216. PMLR, 2020.

Cullina, D., Bhagoji, A. N., and Mittal, P. Pac-learning in the presence of evasion adversaries. *arXiv preprint arXiv:1806.01471*, 2018.

Davis, D. and Drusvyatskiy, D. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.

Dong, C., Liu, L., and Shang, J. Label noise in adversarial training: A novel perspective to study robust overfitting. *Advances in Neural Information Processing Systems*, 35: 17556–17567, 2022.

Farnia, F., Zhang, J., and Tse, D. Generalizable adversarial training via spectral normalization. In *International Conference on Learning Representations*, 2018.

Feldman, V. and Vondrak, J. Generalization bounds for uniformly stable algorithms. *Advances in Neural Information Processing Systems*, 31, 2018.

Feldman, V. and Vondrak, J. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. In *Conference on Learning Theory*, pp. 1270–1279. PMLR, 2019.

Feldman, V., Koren, T., and Talwar, K. Private stochastic convex optimization: optimal rates in linear time. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 439–449, 2020.

Gao, Q. and Wang, X. Theoretical investigation of generalization bounds for adversarial learning of deep neural networks. *Journal of Statistical Theory and Practice*, 15 (2):1–28, 2021.

Gowal, S., Qin, C., Uesato, J., Mann, T., and Kohli, P. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

Hardt, M., Recht, B., and Singer, Y. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, pp. 1225–1234. PMLR, 2016.

Izmailov, P., Wilson, A., Podoprikhin, D., Vetrov, D., and Garipov, T. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pp. 876–885, 2018.

Kanai, S., Yamada, M., Takahashi, H., Yamanaka, Y., and Ida, Y. Relationship between nonsmoothness in adversarial training, constraints of attacks, and flatness in the input space. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

Khim, J. and Loh, P.-L. Adversarial risk bounds via function transformation. *arXiv preprint arXiv:1810.09519*, 2018.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Kurakin, A., Goodfellow, I. J., and Bengio, S. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pp. 99–112. Chapman and Hall/CRC, 2018.

Lei, Y. Stability and generalization of stochastic optimization with nonconvex and nonsmooth problems. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 191–227. PMLR, 2023.

Levi, M., Attias, I., and Kontorovich, A. Domain invariant adversarial learning. *Transactions on Machine Learning Research*, 2022.

Li, L. and Spratling, M. Understanding and combating robust overfitting via input loss landscape analysis and regularization. *Pattern Recognition*, 136:109229, 2023.

Liu, C., Salzmann, M., Lin, T., Tomioka, R., and Süsstrunk, S. On the loss landscape of adversarial training: Identifying challenges and how to overcome them. *Advances in Neural Information Processing Systems*, 33:21476–21487, 2020.

Liu, H., Zhong, Z., Sebe, N., and Satoh, S. Mitigating robust overfitting via self-residual-calibration regularization. *Artificial Intelligence*, 317:103877, 2023.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018a.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018b.

Mao, C., Zhong, Z., Yang, J., Vondrick, C., and Ray, B. Metric learning for adversarial robustness. *Advances in neural information processing systems*, 32, 2019.

Montasser, O., Hanneke, S., and Srebro, N. Vc classes are adversarially robustly learnable, but only improperly. In *Conference on Learning Theory*, pp. 2512–2530. PMLR, 2019.

Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

Moreau, J.-J. Proximité et dualité dans un espace hilbertien. *Bulletin de la Société mathématique de France*, 93:273–299, 1965.

Mustafa, W., Lei, Y., and Kloft, M. On the generalization analysis of adversarial learning. In *International Conference on Machine Learning*, pp. 16174–16196. PMLR, 2022.

Naseer, M., Khan, S., Hayat, M., Khan, F. S., and Porikli, F. A self-supervised approach for adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 262–271, 2020.

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

Nemirovski, A. S. and Yudin, D. B. Problem complexity and method efficiency in optimization. 1983.

Nesterov, Y. Smooth minimization of non-smooth functions. *Mathematical programming*, 103:127–152, 2005.

Ozdaglar, A., Pattathil, S., Zhang, J., and Zhang, K. What is a good metric to study generalization of minimax learners? *Advances in Neural Information Processing Systems*, 35:38190–38203, 2022.

Pang, T., Yang, X., Dong, Y., Xu, K., Zhu, J., and Su, H. Boosting adversarial training with hypersphere embedding. *Advances in Neural Information Processing Systems*, 33:7779–7792, 2020.

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium*

on security and privacy (EuroS&P), pp. 372–387. IEEE, 2016.

Rebuffi, S.-A., Gowal, S., Calian, D. A., Stimberg, F., Wiles, O., and Mann, T. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021.

Rice, L., Wong, E., and Kolter, Z. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pp. 8093–8104. PMLR, 2020.

Rockafellar, R. T. Monotone operators and the proximal point algorithm. *SIAM j. control optim.*, 14(5):877–898, August 1976.

Rogers, W. H. and Wagner, T. J. A finite sample distribution-free performance bound for local discrimination rules. *The Annals of Statistics*, pp. 506–514, 1978.

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems*, pp. 5014–5026, 2018.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014*, 2014.

Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*, 2018.

Tramer, F., Carlini, N., Brendel, W., and Madry, A. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020.

Wang, P., Lei, Y., Ying, Y., and Zhou, D.-X. Stability and generalization for markov chain stochastic gradient methods. *Advances in Neural Information Processing Systems*, 35:37735–37748, 2022.

Wang, Y., Li, L., Yang, J., Lin, Z., and Wang, Y. Balance, imbalance, and rebalance: Understanding robust overfitting from a minimax game perspective. *Advances in neural information processing systems*, 36, 2024.

Xiao, J., Fan, Y., Sun, R., and Luo, Z.-Q. Adversarial rademacher complexity of deep neural networks. *arXiv preprint arXiv:2211.14966*, 2022a.

Xiao, J., Fan, Y., Sun, R., Wang, J., and Luo, Z.-Q. Stability analysis and generalization bounds of adversarial training. *Advances in Neural Information Processing Systems*, 35: 15446–15459, 2022b.

Xiao, J., Qin, Z., Fan, Y., Wu, B., Wang, J., and Luo, Z.-Q. Adaptive smoothness-weighted adversarial training for multiple perturbations with its stability analysis. *arXiv preprint arXiv:2210.00557*, 2022c.

Xiao, J., Yang, L., Fan, Y., Wang, J., and Luo, Z.-Q. Understanding adversarial robustness against on-manifold adversarial examples. *arXiv preprint arXiv:2210.00430*, 2022d.

Xiao, J., Zhang, J., Luo, Z.-Q., and Ozdaglar, A. E. Smoothed-sgdmax: A stability-inspired algorithm to improve adversarial generalization. In *NeurIPS ML Safety Workshop*, 2022e.

Xiao, J., Sun, R., and Luo, Z.-Q. Pac-bayesian spectrally-normalized bounds for adversarially robust generalization. *Advances in Neural Information Processing Systems*, 36: 36305–36323, 2023.

Xing, Y., Song, Q., and Cheng, G. On the algorithmic stability of adversarial training. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=xz80iPFIjvG.

Yang, Z., Lei, Y., Lyu, S., and Ying, Y. Stability and differential privacy of stochastic gradient descent for pairwise learning with non-smooth loss. In *International Conference on Artificial Intelligence and Statistics*, pp. 2026–2034. PMLR, 2021.

Yin, D., Kannan, R., and Bartlett, P. Rademacher complexity for adversarially robust generalization. In *International Conference on Machine Learning*, pp. 7085–7094. PMLR, 2019.

Yu, C., Han, B., Shen, L., Yu, J., Gong, C., Gong, M., and Liu, T. Understanding robust overfitting of adversarial training and beyond. In *International Conference on Machine Learning*, pp. 25595–25610. PMLR, 2022.

Yuan, X.-T. and Li, P. Sharper analysis for minibatch stochastic proximal point methods: Stability, smoothness, and deviation. *Journal of Machine Learning Research*, 20:1–54, 2023.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.

Zhang, S., Gao, H., Zhang, T., Zhou, Y., and Wu, Z. Alleviating robust overfitting of adversarial training with consistency regularization. *arXiv preprint arXiv:2205.11744*, 2022.

A. Proof of Theorems

A.1. Proof of Lemma 4.1 and 4.3

The proof of Lemma 4.1 and 4.3 are related. It suffices to prove the Lemma in weakly-convex case. We first state the Lemma in weakly-convex case.

Lemma A.1. Assume that h is $-l$-weakly convex. Let $p > l$. Then, $M(u; S)$ satisfies

1. $\min_u M(u; S)$ has the same global solution set as $\min_w R_S(w)$.

2. The gradient of $M(u; S)$ is $\nabla_u M(u; S) = p(u - w(u; S))$.

3. $M(u; S)$ is $pl/(p - l)$-weakly convex in u.

4. $M(u; S)$ is $\max\{p, pl/(p - l)\}$-gradient Lipschitz continuous.

Lemma 4.1 and 4.3 hold by letting $l = 0$. To simplify the notation, we use $M(u)$ as a short hand notation of $M(u; S)$. Similar to $h(u)$, $K(u)$, and $w(u)$.

1. Let $w^* \in \arg\min R_S(w)$. We have

$$R_S(w^*) = K(w^*, u = w^*, S) \geq K(w(u), u = w^*, S) \geq R_S(w(u = w^*)).$$

Then, the equality holds. Therefore, $w = u = w^*$ is the optimal solution of both $\min_w R_S(w)$ and $\min_u M(u; S)$.

2. Since $K(w, u)$ is a $(p - l)$-strongly convex function, $w(u)$ is unique. Then

$$M(u) = h(w(u)) + \frac{p}{2}\|w(u) - u\|^2.$$

By taking the derivative of $M(u)$ with respect to u, we have

$$\nabla_u M(u) \quad = \quad \left[\frac{\partial w(u)}{\partial u}\right]^T \cdot \nabla_{w(u)} h(w(u)) + \left[\frac{\partial w(u)}{\partial u} - I\right]^T \cdot p(w(u) - u). \tag{A.1}$$

$$= \quad \left[\frac{\partial w(u)}{\partial u}\right]^T \cdot (\nabla_{w(u)} h(w(u)) + p(w(u) - u)) + p(u - w(u)). \tag{A.2}$$

Since $w(u)$ is the optimal solution of $K(w, u)$, we have

$$\nabla_{w(u)} K(w(u), u) = \nabla_{w(u)} h(w(u)) + p(w(u) - u) = 0. \tag{A.3}$$

Therefore, the first term in A.2 is equal to zero. We have $\nabla_u M(u) = p(u - w(u))$.

3. In Equation (A.3), take the derivatives with respect to u on both sides, we have

$$\left[\frac{\partial w(u)}{\partial u}\right]^T \nabla_w^2 h(w) + p(\left[\frac{\partial w(u)}{\partial u}\right]^T - I) = 0.$$

Organizing the terms, we have

$$\left[\frac{\partial w(u)}{\partial u}\right]^T (\nabla_w^2 h(w) + pI) = pI.$$

Since $h(w)$ is $-l$-weakly convex, $\nabla_w^2 h(w) + pI$ is positive definite. Then,

$$\left[\frac{\partial w(u)}{\partial u}\right]^T \prec \frac{p}{p - l} I. \tag{A.4}$$

Then,

$$\nabla_u^2 M(u) = [\frac{\partial}{\partial u} p(u - w(u))]^T = p(I - \left[\frac{\partial w(u)}{\partial u}\right]^T) \succ p(1 - \frac{p}{p - l})I.$$

Therefore, $M(u)$ is a $pl/(p-l)$-weakly convex function.

4. By Equation (A.4), we have

$$
\begin{aligned}
&\|\nabla M(u_1) - \nabla M(u_2)\| \\
=\ & p\|u_1 - w(u_1) - u_2 - w(u_2)\| \\
\leq\ & p\max\{\|u_1 - u_2\|, \|w(u_1) - w(u_2)\|\} \\
\leq\ & p\max\{1, l/(p-l)\}\|u_1 - u_2\| \\
=\ & \max\{p, pl/(p-l)\}\|u_1 - u_2\|.
\end{aligned}
$$

Therefore, $M(u; S)$ is $\max\{p, pl/(p-l)\}$-gradient Lipschitz continuous. $\qquad\square$

A.2. Proof of Lemma 4.2 and 4.4

Now we discuss the proof of the uniform stability of inner and outer minimization.

Proof of Lemma 4.2: It suffices to prove the Lemma in weakly-convex case. We first introduce the following Lemma of uniform stability of inner minimization.

Lemma A.2 (Uniform Stability of Inner Minimization in Weakly-Convex Case). *In weakly-convex case, for neighbouring S and S', we have*

$$
\|w(u; S) - w(u; S')\| \leq 2L/(n(p-\ell)).
$$

Proof. By the $(p-l)$-strongly convexity of $K(w, u; S)$, we have

$$
\begin{aligned}
&(p-l)\|w(u; S) - w(u; S')\| \\
\leq\ & \|\nabla K(w(u; S), u; S) - \nabla K(w(u; S'), u; S)\| \\
\leq\ & \|\nabla K(w(u; S), u; S) - \nabla K(w(u; S'), u; S')\| \\
& + \frac{1}{n}\|\nabla h(w(u; S'), z_i)\| + \frac{1}{n}\|\nabla h(w(u; S'), z_i')\| \\
=\ & \frac{1}{n}\|\nabla h(w(u; S'), z_i)\| + \frac{1}{n}\|\nabla h(w(u; S'), z_i')\| \\
\leq\ & \frac{2L}{n},
\end{aligned}
$$

where the second inequality is due to the definition of $K(w, u; S)$, the third one is due to the first-order optimally condition, and the last inequality is because of the bounded gradient of $h(w; z)$. $\qquad\square$

Next, we move to the proof of Lemma 4.4. Lemma 4.4 is not obtained from the work of (Hardt et al., 2016). Notice that

$$
M(u; S) = \min_{w \in W} \frac{1}{n} \sum_{z \in S} K(w, u; z) \neq \frac{1}{n} \sum_{z \in S} \min_{w \in W} K(w, u; z).
$$

$\min_u M(u; S)$ is not a finite sum problem. The analysis in (Hardt et al., 2016) can only be applied to finite sum problems. Lemma 4.4 requires a different proof. In summary, there are two steps:

1. Build the recursion from $\|u_S^t - u_{S'}^t\|$ to $\|u_S^{t+1} - u_{S'}^{t+1}\|$;

2. Unwind the recursion.

The main difference comes from the first step.

Step 1.

$$
\begin{aligned}
& \|u_S^{t+1} - u_{S'}^{t+1}\| \\
=\ & \|u_S^t - u_{S'}^t - \alpha_t(\nabla M(u_S^t; S) - \nabla M(u_{S'}^t; S'))\| \\
\leq\ & \|u_S^t - u_{S'}^t - \alpha_t(\nabla M(u_S^t; S) + \nabla M(u_{S'}^t; S))\| + \alpha_t\|\nabla M(u_{S'}^t; S') - \nabla M(u_{S'}^t; S)\| \\
\leq\ & \|u_S^t - u_{S'}^t\| + \alpha_t\|\nabla M(u_{S'}^t; S') - \nabla M(u_{S'}^t; S)\| \\
=\ & \|u_S^t - u_{S'}^t\| + \alpha_t p\|u_{S'}^t - u_{S'}^t - w(u_{S'}^t, S) + w(u_{S'}^t, S')\| \\
\leq\ & \|u_S^t - u_{S'}^t\| + \frac{2L\alpha_t}{n},
\end{aligned}
$$

where the second inequality is due to the non-expansive property of convex function , the last inequality is due to Lemma A.2.

Step 2. Unwinding the recursion, we have

$$
\|u_S^T - u_{S'}^T\| \leq \frac{2L\sum_{t=1}^{T}\alpha_t}{n}.
$$

$\square$

A.3. Proof of Theorem 4.5

Proof. We decompose $\|u_S^{t+1} - u_{S'}^{t+1}\|$ as

$$
\begin{aligned}
& \mathbb{E}\|u_S^{t+1} - u_{S'}^{t+1}\| \\
=\ & \mathbb{E}\|u_S^t - \alpha_t \nabla_u K(w_{N,S}^t, u_S^t; S) - u_{S'}^t + \alpha_t \nabla_u K(w_{N,S'}^t, u_{S'}^t; S')\| \\
\leq\ & \mathbb{E}\|u_S^t - \alpha_t \nabla_u M(u_S^t; S) - u_{S'}^t + \alpha_t \nabla_u M(u_{S'}^t; S')\| \\
+\ & 2\alpha_t \mathbb{E}\|\nabla_u K(w_{N,S}^t, u_S^t; S) - \nabla_u M(u_S^t; S)\| \\
\leq\ & \mathbb{E}\|u_S^t - u_{S'}^t\| + \frac{2L\alpha_t}{n} + 2\alpha_t p \mathbb{E}\|w_N^t - w(u^t)\| \\
\leq\ & \mathbb{E}\|u_S^t - u_{S'}^t\| + \frac{2L\alpha_t}{n} + 2\alpha_t p \varepsilon(\mathcal{A}).
\end{aligned}
$$

Unwind the recursion and let u^T be the output of the algorithm. We have

$$
\mathcal{E}_{gen} \ \leq\ L\Big(\frac{2L}{n} + 2p\varepsilon(\mathcal{A})\Big)\sum_{t=1}^{T}\alpha_t.
$$

If we choose $w(u^T)$ to be the algorithm output, we have

$$
\begin{aligned}
\mathcal{E}_{gen} \ \leq\ & L\mathbb{E}\|w(u_S^T; S) - w(u_{S'}^T; S')\| \\
=\ & L\mathbb{E}\Big\|u_S^T - \frac{1}{p}\nabla M(u_S^T, S) - u_{S'}^T - \frac{1}{p}\nabla M(u_{S'}^T; S'))\Big\| \\
\leq\ & L\mathbb{E}\|u_S^T - u_{S'}^T\| + \frac{2L^2}{np} \\
\leq\ & L\Big(\frac{2L}{n} + 2p\varepsilon(\mathcal{A})\Big)\sum_{t=1}^{T}\alpha_t + \frac{2L^2}{np}.
\end{aligned}
$$

(A.5)

where the first equality is due to $\nabla M(u; S) = p(u - w(u))$, the second inequality is due to the non-expansive propertiy of $M(u; S)$. $\square$

A.4. Proof of Theorem 4.7

The proof contains two steps. The first step is to build the recursion, which is based on the error bound and the decomposition of Lemma 4.4. The second step is to unwind the recursion, which is adopt from the analysis of uniform stability in non-smooth case (Hardt et al., 2016).

Proof:

Step 1. We decompose $\|u_S^{t+1} - u_{S'}^{t+1}\|$ as

$$
\begin{aligned}
& \|u_S^{t+1} - u_{S'}^{t+1}\| \\
= \ & \mathbb{E}\|u_S^t - \alpha_t \nabla_u K(w_{N,S}^t, u_S^t; S) - u_{S'}^t + \alpha_t \nabla_u K(w_{N,S'}^t, u_{S'}^t; S')\| \\
\leq \ & \mathbb{E}\|u_S^t - \alpha_t \nabla_u M(u_S^t; S) - u_{S'}^t + \alpha_t \nabla_u M(u_{S'}^t; S')\| \\
+ \ & 2\alpha_t \mathbb{E}\|\nabla_u K(w_{N,S}^t, u_S^t; S) - \nabla_u M(u_S^t; S)\| \\
\leq \ & \|u_S^t - u_{S'}^t - \alpha_t(\nabla M(u_S^t; S) - \nabla M(u_{S'}^t; S'))\| + 2\alpha_t p \varepsilon(\mathcal{A}) \\
\leq \ & \|u_S^t - u_{S'}^t - \alpha_t(\nabla M(u_S^t; S) + \nabla M(u_{S'}^t; S))\| + \alpha_t \|\nabla M(u_{S'}^t; S') - \nabla M(u_{S'}^t; S)\| + 2\alpha_t p \varepsilon(\mathcal{A}) \\
\leq \ & \|u_S^t - u_{S'}^t\| + \alpha_t \|\nabla M(u_S^t; S) - \nabla M(u_{S'}^t; S)\| + \alpha_t \|\nabla M(u_{S'}^t; S') - \nabla M(u_{S'}^t; S)\| + 2\alpha_t p \varepsilon(\mathcal{A}) \\
\leq \ & (1 + \alpha_t \beta)\|u_S^t - u_{S'}^t\| + \alpha_t \|\nabla M(u_{S'}^t; S') - \nabla M(u_{S'}^t; S)\| + 2\alpha_t p \varepsilon(\mathcal{A}), \quad \text{(A.6)}
\end{aligned}
$$

where the first inequality is due to triangular inequality, the second inequality is due to the assumption of the convergence rate of the inner minimization problem, and the third and fourth inequalities are due to triangular inequality. The last inequality is due to the gradient Lipschitz of $M(u; S)$ and $\beta = \max\{p, pl/p - l\}$. Then,

$$
\begin{aligned}
& \alpha_t \|\nabla M(u_{S'}^t; S') - \nabla M(u_{S'}^t; S)\| \\
= \ & \alpha_t p \|u_{S'}^t - u_{S'}^t - w(u_{S'}^t, S) + w(u_{S'}^t, S')\| \\
\leq \ & \frac{2Lp\alpha_t}{(p-l)n}, \quad \text{(A.7)}
\end{aligned}
$$

where the first inequality is due to the form of $\nabla M(u; S)$, the last equality is due to Lemma A.2.

Combining Equation (A.6) and (A.7), we have

$$
\begin{aligned}
& \|u_S^{t+1} - u_{S'}^{t+1}\| \\
\leq \ & (1 + \alpha_t \beta)\|u_S^t - u_{S'}^t\| + \frac{2Lp\alpha_t}{(p-l)n} + 2\alpha_t p \varepsilon(\mathcal{A}). \quad \text{(A.8)}
\end{aligned}
$$

Step 2. Let S and S' be two samples of size n differing in only a single example. Consider two trajectories $w_1^1, \ldots, w_1^T$ and $w_2^1, \ldots, w_2^T$ induced by running SGD on sample S and S', respectively. Let $\delta_t = \|w_1^t - w_2^t\|$. Let $t_0 \in \{0, 1, \ldots, n\}$, be the iteration that $\delta_{t_0} = 0$, but SGD picks two different samples form S and S' in iteration $t_0 + 1$, then

$$
\mathcal{E}_{gen} \leq \frac{t_0}{n} B + L\mathbb{E}\left[\delta_T \mid \delta_{t_0} = 0\right]. \quad \text{(A.9)}
$$

Let $\Delta_t = \mathbb{E}\left[\delta_t \mid \delta_{t_0} = 0\right]$, and $\alpha_t \leq c/(\beta t)$. Then,

$$
\begin{aligned}
\Delta_{t+1} &\leq (1 + \alpha_t \beta)\Delta_t + \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A})\right)\alpha_t \\
&= \left(1 + \frac{c\beta}{t}\right)\Delta_t + \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A})\right)\frac{c}{t} \\
&\leq \exp\left(\frac{c\beta}{t}\right)\Delta_t + \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A})\right)\frac{c}{t}.
\end{aligned}
$$

Here we used the fact that $1 + x \le \exp(x)$ for all x.

Using the fact that $\Delta_{t_0} = 0$, we can unwind this recurrence relation from T down to $t_0 + 1$. This gives

$$
\begin{aligned}
\Delta_T &\le \sum_{t=t_0+1}^{T} \left\{ \prod_{k=t+1}^{T} \exp\left(\tfrac{\beta c}{k}\right) \right\} \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A}) \right) \frac{c}{t} \\
&= \sum_{t=t_0+1}^{T} \exp\left(\beta c \sum_{k=t+1}^{T} \tfrac{1}{k} \right) \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A}) \right) \frac{c}{t} \\
&\le \sum_{t=t_0+1}^{T} \exp\left(\beta c \log(\tfrac{T}{t}) \right) \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A}) \right) \frac{c}{t} \\
&= \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A}) \right) c T^{\beta c} \sum_{t=t_0+1}^{T} t^{-\beta c - 1} \\
&\le \left(\frac{2Lp}{(p-l)n} + 2p\varepsilon(\mathcal{A}) \right) \frac{1}{\beta c} c \left(\frac{T}{t_0} \right)^{\beta c} \\
&\le \frac{p}{\beta} \left(\frac{2L}{(p-l)n} + 2\varepsilon(\mathcal{A}) \right) \left(\frac{T}{t_0} \right)^{\beta c},
\end{aligned}
$$

Plugging this bound into (A.9), we get

$$
\mathcal{E}_{gen} \le \frac{B t_0}{n} + \frac{Lp}{\beta} \left(\frac{2L}{(p-l)n} + 2\varepsilon(\mathcal{A}) \right) \left(\frac{T}{t_0} \right)^{\beta c}.
$$

Let $q = \beta c$. We select t0 to optimize the right hand side

$$
\mathcal{E}_{gen} \le \frac{B t_0}{n} + \frac{Lp}{\beta} \left(\frac{2L}{(p-l)n} + 2\varepsilon(\mathcal{A}) \right) \left(\frac{T}{t_0} \right)^{\beta c}.
$$

If algorithm $\mathcal{A}$ satisfies $\varepsilon(\mathcal{A}) = \mathcal{O}(1/(p-l)n)$,

$$
\mathcal{O}\left(\frac{2Lp T^{\beta c} \varepsilon(\mathcal{A})}{\beta} \right) = \mathcal{O}\left(\frac{2Lp T^{\beta c}}{\beta (p-l)n} \right) = \mathcal{O}\left(\frac{2L T^{\beta c}}{\min\{l, p-l\} n} \right)
$$

Let $q = \frac{\beta c}{1 + \beta c}$, Optimize over t_0. Then, we have

$$
\mathcal{E}_{gen} \le \mathcal{O}\left(\frac{2L^2 T^q}{\min\{l, p-l\} n} \right).
$$

$\square$

A.5. Proof of Theorem 4.8

Now we consider the convergence rate of $ME - \mathcal{A}$. In this part, we simply let $M(u)$ and $K(w, u)$ as short hand notations of $M(u; S)$ and $K(w, u; S)$.

By the p-gradient Lipschitzness of $M(u)$, we have

$$
M(u^{t+1}) \le M(u^t) + \langle \nabla M(u^t), u^{t+1} - u^t \rangle + \frac{p}{2} \|u^{t+1} - u^t\|^2.
$$

By the update rules of u^t, we have $u^{t+1} = u^t - \alpha_t \nabla K(w^{t+1}, u^t)$ or $u^{t+1} = u^t + \alpha_t p(w^{t+1} - u^t)$. Then,

$$
M(u^{t+1}) \le M(u^t) - \alpha_t \langle \nabla M(u^t), \nabla K(w^{t+1}, u^t) \rangle + \frac{p \alpha_t^2}{2} \|\nabla K(w^{t+1}, u^t)\|^2.
$$

Based on the convexity of $M(u)$, we have

$$M(u) \leq M(u^*) + \langle \nabla M(u), u - u^* \rangle.$$

Then

$$
\begin{aligned}
& M(u^{t+1}) - M(u^*) \\
\leq\ & \langle \nabla M(u^t), u^t - u^* \rangle - \alpha_t \langle \nabla M(u^t), \nabla K(w^{t+1}, u^t) \rangle + \frac{p\alpha_t^2}{2} \|\nabla K(w^{t+1}, u^t)\|^2 \\
=\ & \langle \nabla M(u^t), u^t - u^* - \alpha_t \nabla K(w^{t+1}, u^t) \rangle + \frac{p\alpha_t^2}{2} \|\nabla K(w^{t+1}, u^t)\|^2 \\
=\ & \langle \nabla M(u^t), u^{t+1} - u^* \rangle + \frac{p\alpha_t^2}{2} \|\nabla K(w^{t+1}, u^t)\|^2 \\
=\ & \langle \nabla K(w^{t+1}, u^t), u^{t+1} - u^* \rangle + \frac{p\alpha_t^2}{2} \|\nabla K(w^{t+1}, u^t)\|^2 \\
& + \langle \nabla M(u^t) - \nabla K(w^{t+1}, u^t), u^{t+1} - u^* \rangle.
\end{aligned}
\tag{A.10}
$$

Let $\alpha_t = \alpha \leq 1/p$. The sum of the first two terms is bounded by $\frac{1}{2\alpha}[\|u^t - u^*\|^2 - \|u^{t+1} - u^*\|^2]$, it is because

$$
\begin{aligned}
& \frac{1}{2\alpha}[\|u^t - u^*\|^2 - \|u^{t+1} - u^*\|^2] \\
=\ & \frac{1}{2\alpha}[2\alpha \langle \nabla K(w^{t+1}, u^t), u^t - u^* \rangle - \alpha^2 \|\nabla K(w^{t+1}, u^t)\|^2] \\
=\ & \langle \nabla K(w^{t+1}, u^t), u^t - u^* \rangle - \frac{\alpha}{2} \|\nabla K(w^{t+1}, u^t)\|^2 \\
=\ & \langle \nabla K(w^{t+1}, u^t), u^{t+1} - u^* \rangle + \frac{\alpha}{2} \|\nabla K(w^{t+1}, u^t)\|^2 \\
\geq\ & \langle \nabla K(w^{t+1}, u^t), u^{t+1} - u^* \rangle + \frac{p\alpha^2}{2} \|\nabla K(w^{t+1}, u^t)\|^2.
\end{aligned}
\tag{A.11}
$$

The Last term in Equation (A.10)

$$\langle \nabla M(u^t) - \nabla K(w^{t+1}, u^t), u^{t+1} - u^* \rangle \leq p\|w^{t+1} - w(u^t)\| \cdot \|u^{t+1} - u^*\|.$$

Then, we have

$$
\begin{aligned}
& M(u^{t+1}) - M(u^*) \\
=\ & \frac{1}{2\alpha}[\|u^t - u^*\|^2 - \|u^{t+1} - u^*\|^2] + \varepsilon(\mathcal{A})p\|u^{t+1} - u^*\|.
\end{aligned}
$$

Sum over $t = 0$ to T, we obtain

$$\mathcal{E}_{opt} \leq \mathcal{O}\left(\frac{\|u_0 - u^*\|^2}{2\alpha T} + \varepsilon(\mathcal{A})p \frac{\sum_{t=1}^{T} \|u^t - u^*\|^2}{T} \right) = \mathcal{O}\left(\frac{D_0}{2\alpha T} + \varepsilon(\mathcal{A})pD_1 \right).$$

Furthermore, if algorithm $\mathcal{A}$ satisfies $\varepsilon(\mathcal{A}) = \mathcal{O}(1/p\alpha T)$, the generalization gap satisfies

$$\mathcal{E}_{opt} \leq \mathcal{O}\left(\frac{1}{T\alpha} \right).$$

$\square$

B. Discussion on the Complexity of the Inner Problem

B.1. Convergence Rate of Strongly Convex Problems

In this section, we consider the convergence rate of the inner minimization problem. Notice that the inner problem is a $(p - l)$-strongly convex, non-smooth problem.

Lipschitz Case. We first consider the case that $K(w, u; z)$ is Lipschitz. If we assume that the domain W of w is bounded by $D_W/2$, since

$$\|\nabla_w K(w, u; z)\| = \|\nabla_w h(w; z) + p(w - u)\| \leq L + pD_W,$$

$K(w, u; z)$ has bounded gradient $L_K = L + pD_W$, i.e., $K(w, u; z)$ is L_K-Lipschitz. Then the convergence rate of running SGD on the inner problem can be obtained from classical strongly-convex optimization results.

Lemma B.1. *Given t and u^t, suppose we run SGD on $K(w, u^t, S)$ w.r.t. w with stepsize $c_s^t \leq 1/(p-l)s$ for N steps. w_N^t is approximately the minimizer with an error C_1^2/N, i.e.,*

$$E\|w_N^t - w(u^t)\|^2 \leq \frac{C_1^2}{N},$$

where $C_1 = (L + pD_W)/(p - l)$.

By (Nemirovski et al., 2009), running SGD on $K(w, u; S)$ with stepsize $c_s \leq 1/s(p - l)$ iccurs an optimization error in

$$E\|w_N - w(u)\|^2 \leq \frac{C_1^2}{N},$$

where $C_1 = (L + pD_W)/(p - l)$. Then, by Jensen's inequality, we have

$$\epsilon(SGD) = E\|w_N - w(u)\| \leq \sqrt{E\|w_N - w(u)\|^2} \leq \frac{C_1}{\sqrt{N}},$$

Non-Lipshitz Case. Without the $D_W/2$-bounded assumption, $K(w, u^t, S)$ is not Lipschitz, but the sum of a Lipshitz function ($h(w)$) and a gradient Lipshitz function ($\frac{p}{2}\|w - u\|^2$). We provide our convergence analysis in this case. In this case, we apply subgradient method for this strongly convex problem. Consider the following optimization problem:

$$\min_w \phi(w) := h(w) + \frac{p}{2}\|w - u\|^2,$$

where $h(\cdot)$ is convex, L-Lipschitz and has bounded subgradients. We perform subgradient descent to ϕ:

$$w^{t+1} = w^t - cg_h(w^t) - cp(w^t - u).$$

Theorem B.2. *We can obtain $\|w^t - w^*\| \leq \epsilon$ with $t \leq \tilde{\mathcal{O}}(1/\epsilon^2)$ steps, where the constant hidden in $\mathcal{O}$ only depends on L, $\|w^0 - w^*\|$ and p.*

Proof. We have

$$\begin{aligned}
&\|w^{t+1} - w^*\|^2 \\
=\ &\|w^t - cg^t - cp(w^t - u) - w^*\|^2 \\
=\ &\|(w^t - w^*) - c(g^t - g^*) - cp(w^t - w^*)\|^2,
\end{aligned}$$

where the last equality uses the optimality condition in w^* that $g^* + p(w^* - u) = 0$ for some $g^* \in \partial(h(w^*))$. We then have

$$\begin{aligned}
&\|(w^t - w^*) - c(g^t - g^*) - cp(w^t - w^*)\|^2 \\
=\ &(1 - cp)\|w^t - w^*\|^2 + c^2\|g^t - g^*\|^2 \\
&- (g^t - g^*)^T(w^t - w^*) \\
\leq\ &(1 - cp)\|w^t - w^*\|^2 + c^2 L^2,
\end{aligned}$$

where the last inequality uses the fact that $\|g^t\|, \|g^*\| \leq L$ and $(g^t - g^*)^T(w^t - w^*) \geq 0$. Let $\Delta_t = \|w^t - w^*\|^2$. Then we have the recursion

$$\Delta_{t+1} \leq (1 - cp)\Delta_t + c^2 L^2.$$

We have

$$(\Delta_{t+1} - cL^2/p) \leq (1 - cp)(\Delta_t - cL^2/p).$$

Take $c = \epsilon/2$ and we can attain the result. $\qquad\square$

C. Other Discussion

C.1. Additional Related work

Additional Related Work.   In addition to gradient descent, some variants of SGD are also proven to be non-uniformly stable for convex non-smooth problems. We list the stability of most related results in Table 5. We can see that there are also additional term in the bounds of the uniform stability of variants of SGD.

Adversarial Attack.   Adversarial examples were first introduced in (Szegedy et al., 2014). Since then, adversarial attacks have received enormous attention (Papernot et al., 2016; Moosavi-Dezfooli et al., 2016; Carlini & Wagner, 2017). Nowadays, attack algorithms have become sophisticated and powerful. For example, Autoattack (Croce & Hein, 2020) and Adaptive attack (Tramer et al., 2020). Real-world attacks are not always norm-bounded (Kurakin et al., 2018). Xiao et al. (2022d) considered non-ℓ_p attacks.

Adversarially Robust Generalization.   Even enormous algorithms were proposed to improve the robustness of DNNs (Madry et al., 2018b; Gowal et al., 2020; Rebuffi et al., 2021), the performance was far from satisfactory. One major issue is the poor robust generalization, or robust overfitting (Rice et al., 2020). A series of studies (Xiao et al., 2022c; Ozdaglar et al., 2022) have delved into the concept of uniform stability within the context of adversarial training. However, these analyses focused on general Lipschitz functions, without specific consideration for neural networks. There are many approaches designed for mitigating robust overfitting (Chen et al., 2020; Yu et al., 2022; Li & Spratling, 2023; Dong et al., 2022; Liu et al., 2023; Wang et al., 2024; Zhang et al., 2022). In addition to mitigating robust overfitting, several methods for adversarial training have been made employing, including metric learning (Mao et al., 2019; Pang et al., 2020), self-supervised learning (Naseer et al., 2020), domain invariant adversarial learning (Levi et al., 2022), and ensemble learning (Tramèr et al., 2018).

Table 5. Uniform Stability for different variants of SGD algorithms in non-smooth convex minimization problem. Here T is the number of iterations, n is the number of samples, and $\alpha > 0$ is the step size.

	Algorithms	Upper Bounds	Lower Bounds
(Bassily et al., 2020)	GD (full batch)	$\mathcal{O}(\sqrt{T}\alpha + T\alpha/n)$	$\Omega(\sqrt{T}\alpha + T\alpha/n)$
(Bassily et al., 2020)	SGD (w/replacement)	$\mathcal{O}(\sqrt{T}\alpha + T\alpha/n)$	$\Omega(\min\{1, T/n\}\sqrt{T}\alpha + T\alpha/n)$
(Bassily et al., 2020)	SGD (fixed permutation)	$\mathcal{O}(\sqrt{T}\alpha + T\alpha/n)$	$\Omega(\min\{1, T/n\}\sqrt{T}\alpha + T\alpha/n)$
(Yang et al., 2021)	SGD (Pairwise Learning)	$\mathcal{O}(\sqrt{T}\alpha + T\ln T\alpha/n)$	/
(Wang et al., 2022)	Markov chain-SGD	$\mathcal{O}(\sqrt{T}\alpha + T\alpha/n)$	/

C.2. Additional Experiments

In this section, we provide ablation study about the hyperparameter of ME-$\mathcal{A}$. Specifically, the value of p and the step size τ affect the performance of ME-$\mathcal{A}$.

Affect of p.   We first consider the affect of the value of p. When $p = 0$, ME-$\mathcal{A}$ reduces to SWA. These experiments also provide a comparison between ME-$\mathcal{A}$ and SWA. In the Theorem, p does not affect the uniform stability as long as $\alpha \leq 1/p$. But p is related to the optimal rate of iteration complexity. We switch p from 0 to 1×10^5 in the experiment of adversarial training with ME-$\mathcal{A}$ using TRADES loss on CIFAR-10. The results are provided in Table 6. We can see that the value of p does not give a major difference in the performance. The best performance is achieved when $p = 1$.

Affect of τ.   Secondly, we consider the affect of the value of τ_t. When $\tau_t = 0$, ME-$\mathcal{A}$ reduces to PPM. We simply let $\tau_t = \tau$, *i.e.,* fixed stepsize in the experiemnts. These experiments also provide a comparison between ME-$\mathcal{A}$ and PPM. In the Theorem, τ should be large and closed to 1, especial for large t. We switch τ from 0 to 0.995 in the experiment of adversarial training with ME-$\mathcal{A}$ using TRADES loss on CIFAR-10. The results are provided in Table 7. We can see that the robust accuracy decreases as τ decreases. The best performance is achieved when $\tau = 0.995$. Therefore, τ is better to be large. In the experiments in the main contents, τ_t increases as t.

Table 6. The affect of p in ME-$\mathcal{A}$. We switch p from 0 to 1×10^5 in the experiment of adversarial training with ME-$\mathcal{A}$ using TRADES loss.

$p =$	Robust Accuracy
0 (Reduced to SWA)	63.17
1×10^{-5}	63.38
1×10^{-4}	63.28
1×10^{-3}	63.19
1×10^{-2}	62.79
1×10^{-1}	62.69
1×10^{0}	63.86
1×10^{1}	62.89
1×10^{2}	63.47
1×10^{3}	62.40
1×10^{4}	62.79
1×10^{5}	63.41

Table 7. The affect of τ in ME-$\mathcal{A}$. We switch τ from 0 to 0.995 in the experiment of adversarial training with ME-$\mathcal{A}$ using TRADES loss.

$\tau =$	Robust Accuracy
0 (Reduced to PPM)	56.86
0.1	57.02
0.3	57.78
0.5	58.34
0.7	60.32
0.9	62.57
0.995	63.17