ALGORITHMIC STABILITY BASED GENERALIZATION BOUNDS FOR ADVERSARIAL TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper, we present a novel stability analysis of adversarial training and prove generalization upper bounds in terms of an expansiveness property of adversarial perturbations used during training and used for evaluation. These expansiveness parameters appear to not only govern the vanishing rate of the generalization error but also govern its scaling constant. Our bound attributes the robust overfitting in PGD-based adversarial training to the sign function used in the PGD attack, resulting in a bad expansiveness parameter. The peculiar choice of sign function in the PGD attack appears to impact adversarial training both in terms of (inner) optimization and in terms of generalization, as shown in this work. This aspect has been largely overlooked to date. Going beyond the sign-function based PGD attacks, we further show that poor expansiveness properties exist in a wide family of PGD-like iterative attack algorithms, which may highlight an intrinsic difficulty in adversarial training.

1 INTRODUCTION

Deep neural networks, despite their great success, have been shown vulnerable to adversarial attacks (Szegedy et al., 2014; Goodfellow et al., 2015), where carefully constructed small modifications of the input may cause the network to output a wrong prediction. A large body of works (Madry et al., 2019; Zhang et al., 2019; Croce et al., 2020; Shaham et al., 2018; Qin et al., 2019; Shafahi et al., 2019; Wong et al., 2020) then propose revised training algorithms to combat adversarial attacks. These algorithms, usually referred to as adversarial training (or AT in this paper), among which the dominant approaches, such as PGD based AT (Madry et al., 2019), involve perturbing the input in a way similar to adversarial attacks to hopefully maximize the loss function within a prescribed radius (referred to as "inner maximization"). Although these AT algorithms allow the learned model to defend, to some extent, against adversarial attacks, significant challenges remain.

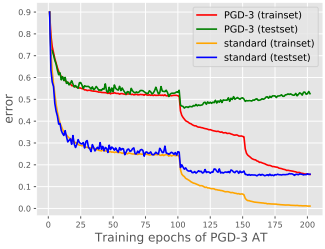


Figure 1: The learning curve of a model trained by AT on CIFAR-10 with 3-step PGD. The standard error as well as the error against the same 3-step PGD attack are measured during AT on both the training and testing sets. The step size for PGD and the perturbation radius w.r.t the $\infty-$norm are respectively set to $7/255$ and $8/255$. The learning rate is decayed at the 100th and the 150th epoch.

First, generalization for such training algorithms is much more difficult, a phenomenon known as "robust overfitting"(Rice et al., 2020). Specifically, Rice et al. (2020) shows that on the CIFAR-10 dataset (Krizhevsky et al., 2009), the model trained by AT using 10-step PGD attack is still vulnerable to the same 10-step PGD attack on the testing set. Our additional experiments (e.g., Figure 1) suggests that this is quite common. In Figure 1, we perform AT with a 3-step PGD and measure the error of the model against 3-step PGD attack as well as its standard error in the training process. We observe that the model trained with 3-step PGD is still vulnerable to the same PGD attack on the testing set. After the first learning rate decay (the 100th epoch), the testing error w.r.t the 3-step PGD starts to rise, similar to the observations in Rice et al. (2020).

Second, it is much more difficult to develop theoretical understanding of the generalization behavior for models obtained from AT, comparing with those from standard training. In that direction, some

theoretical works consider the setting where the inner maximization is perfectly solved, e.g., in Yin et al. (2018); Awasthi et al. (2020). However, such settings are invalid for more complex neural networks, where the closed-form solution for the inner maximization is unavailable. Another line of works use uniform stability to analyze the generalization of AT. In Xing et al. (2021), the adversarial loss is assumed convex and non-smooth and AT is regarded as standard SGD on this loss, whereby an existing generic bound for non-smooth loss in Bassily et al. (2020) is invoked for analysis. As pointed out in Xiao et al. (2022b), the bound obtained in Xing et al. (2021) is independent of the specific choice of loss function used for training and insufficient to reflect the difference between AT and SGD observed in practice. The work of Xiao et al. (2022b) argues that the adversarial loss is approximately smooth and derive bounds based on the stability framework of SGD in Hardt et al. (2016). The work of Wang et al. (2024), built upon Xiao et al. (2022b), extends the analysis of AT to the data-dependent stability framework in Kuzborskij & Lampert (2018). But the bounds obtained in both Xiao et al. (2022b) and Wang et al. (2024) do not vanish with sample size.

To overcome these limitations and shed new light in understanding robust overfitting, we present in this work novel stability analysis for the generalization of models learned using an arbitrary AT algorithm. Specifically, we isolate two aspects in the problem scope. The first is the loss function used for performance evaluation (on both testing and training sets), which can in general be considered as a modification of the standard loss to a version induced by a perturbation map J and which reduces to the adversarial loss when J is specialized to a particular form J^* and to the standard loss when J is specialized to the trivial identity map J^{id}. The second is the perturbation map π used during AT, corresponding to the solution heuristics used for solving the inner maximization problem. When allowing π to potentially deviate from the perturbation J^*, we include in our study the case where the inner maximization is not solved exactly. Additionally, considering $\pi = J$ allows us to study robust overfitting as examplified in Figure 1 where training using a particular attack results in poor generalization when evaluation on the testing set is under the *same* attack.

In this setting, we carry out a stability analysis and present novel generalization bounds for models trained using AT with an arbitrary adversarial perturbation π and evaluated on a loss induced by an arbitrary perturbation J. At the heart of our analysis is the introduction of a notion of "expansiveness" for the perturbation maps (J and π), which governs the behavior of the derived bounds. Specifically, we show that whenever the expansiveness parameter of J is strictly bounded, our generalization bounds vanish with sample size n as $\mathcal{O}(1/n)$ and a small expansiveness parameter of π further helps generalization. On the other hand, when the J-loss (i.e., the loss induced by perturbation J) is defined with J taken as the sign-PGD perturbation, the expansiveness parameter of J is no longer bounded. In this case, our bound reveals an intrinsic tension between the stability parameter, and the perturbation radius, and the ambient data dimension, in their respective roles on generalization – specifically, the bound converges to a constant. When considering $\pi = J$, this helps to explain the robust overfitting phenomenon of sign-PGD AT as shown in Figure 1. Additional advantages of our bounds include the following. Our generic bound (Theorem 4.2) is applicable to AT algorithms based on any form of adversarial perturbations. Our bounds do not rely on any assumption on the adversarial loss directly, since we only make assumptions on the standard loss and all properties of the adversarial loss are induced via perturbation map J. Finally, varying the form of J potentially enables this framework to be applicable to settings where generalization on other performance metrics is of interest.

We zoom into models trained with multi-step PGD, and further demonstrate that the sign function used in the perturbation is an important cause of robust overfitting for such AT methods. We experimentally replace the sign function in PGD with a smooth approximation $\tanh_\gamma$, where $\tanh_\gamma(x) = \tanh(\gamma x)$ and the parameter γ controls the smoothness of the function and hence the expansiveness of the PGD perturbation (decreasing γ decreases the expansiveness). Our experiments show that reducing γ results in smaller generalization gaps. These results validate our bound and its implication on generalization. Interestingly our experiments also reveal that sign-PGD appears as a stronger attack than $\tanh_\gamma$-PGD and the raw gradient (RG)-PGD attack, even on the training set. Performing AT with $\tanh_\gamma$-PGD and RG-PGD may be inadequate for defending against the sign-PGD attacks on the training set. Our observations suggest that sign-function, a building block of PGD-based AT, appears to play a peculiar role: comparing with the $\tanh_\gamma$ counter-part, the sign function helps to better solve the inner maximization problem but at the same time cause the perturbation π to suffer from bad expansiveness and results in poor generalization. This aspect of

sign-PGD has been largely overlooked to date, since most theoretical analysis of PGD removes the sign function in their consideration (i.e., studying RG-PGD instead).

In this work, we also recognize sign-PGD as an iterative method for solving the inner maximization problem where each step is principled by a locally linear approximation of the loss function. Based on this principle, we extend sign-PGD to a wider family of perturbations. We show theoretically that every member in this family suffers from poor expansiveness. This result seems to point to certain intrinsic difficulty in training models adversarially.

2 Other related works

Robust generalization Beyond investigations via algorithmic stability perspectives (e.g., Xing et al. (2021); Xiao et al. (2022b); Wang et al. (2024)), robust generalization has also been studied under the uniform convergence framework with conventional statistical learning tools such as VC dimension (Montasser et al., 2019), Rademacher complexity (Khim & Loh, 2019; Yin et al., 2018; Awasthi et al., 2020; Xiao et al., 2022a; Attias et al., 2018) and other PAC learning frameworks (Cullina et al., 2018; Diochnos et al., 2019). Moreover, robust generalization has been investigated via the curvature of the local minima of the loss landscape: AT is observed to have a tendency to reach sharper minima (Liu et al., 2020), and flatter minima usually results in better generalization (Wu et al., 2020). The work of Chen et al. (2020) observes that robust overfitting can be alleviated by smoothing the model prediction via knowledge distillation. The difficulty of achieving robust generalization has also been attributed to the inadequate expressive power of practical deep learning models (Li et al., 2022), insufficient sample size for models to generalize (Schmidt et al., 2018) as well as and the model's tendency to interpolate "hard training instances" (Liu et al., 2021).

Uniform stability Uniform stability was first introduced by the landmark work of Bousquet & Elisseeff (2002). An influential work by Hardt et al. (2016) adapts this framework to analyze the uniform stability of SGD with smooth loss functions, explaining the effectiveness of SGD in training neural networks. Since then, many studies have built upon Hardt et al. (2016) to develop stability bounds for SGD with non-smooth losses (e.g., Bassily et al. (2020); Lei & Ying (2020)). Data-dependency in stability analysis is introduced in Kuzborskij & Lampert (2018), and uniform stability for more sophisticated variants of SGD is also studied (e.g., Mou et al. (2018); Chen et al. (2018)). Additionally, works such as Farnia & Ozdaglar (2021); Lei et al. (2021) have explored algorithmic stability in general minimax problems. These studies are more closely related to generative adversarial networks (GANs), rather distant from the standard settings of adversarial training.

3 Problem setup and preliminaries

Over any real vector space, we will use $\|\cdot\|_p$ to denote the p-norm and abbreviate the Euclidean norm (i.e., 2-norm) as $\|\cdot\|$. For a vector $x \in \mathbb{R}^d$, $x[i]$ denotes the ith coordinate of x.

We consider the standard setting of supervised learning, where the training samples are instance-label pairs, (x_i, y_i)'s, drawn i.i.d from an underlying data distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. Here the input space $\mathcal{X}$ is $\mathbb{R}^d$ and the label space $\mathcal{Y}$ is a finite. We restrict to parameterized models, e.g., neural networks, in which the model parameter w lives in a subset $\mathcal{W}$ of some real vector space. We use $f(w, x, y)$ to denote the loss value of (x, y) under model parameter w, where a standard choice of loss function (e.g. 0-1 loss, cross-entropy loss, etc.) is absorbed in f. For example, $f(w, x, y)$ can be the cross-entropy loss of the a neural network with parameter w on sample (x, y).

The central object of this study is adversarial training, which allows the learned model to resist adversarial attacks. Each adversarial attack (or adversarial perturbation) on input x is assumed to live in an ∞-norm ball $\mathbb{B}_\infty(x, \epsilon) := \{t \in \mathbb{R}^d : \|t - x\|_\infty \leq \epsilon\}$ with radius ϵ and centered at x.

Perturbation induced loss Let J be a function mapping $\mathcal{W} \times \mathcal{X} \times \mathcal{Y}$ to $\mathcal{X}$ satisfying $J(x; y, w) \in \mathbb{B}_\infty(x, \epsilon)$. Then $J(x; y, w)$ may be regarded as a perturbation of x by a magnitude of up to ϵ (under ∞-norm). We then define the *perturbation J induced loss* or simply *J-loss* by

$$f_J(w, x, y) := f\left(w, J(x; y, w), y\right) \tag{1}$$

Let $J^*(x; y, w) := \arg\max_{\hat{x}\in\mathbb{B}(x,\epsilon)} f(w, \hat{x}, y)$, and $J^{\mathrm{id}}(x; y, w) := x$. Then it is easy to verify that when $J = J^*$, $f_J(w, x, y)$ is the adversarial loss $\max_{\hat{x}\in\mathbb{B}(x,\epsilon)} f(w; \hat{x}, y)$ —for which reason, we will denote the adversarial loss f_{J^*} by f^* for simplicity —and when $J = J^{\mathrm{id}}$, $f_J(w, x, y)$ is the standard loss $f(w, x, y)$. We will soon encounter other forms of J-loss.

Generalization w.r.t the induced loss Let the training set $S = \{(x_i, y_i)\}_{i=1}^n$ be drawn from $\mathcal{D}^n$. Consider a learning algorithm A, which when applied on S gives rise to a learned model parameter $w = A(S)$. Notably w entails randomness, due to the random sampling of S and the possible intrinsic randomness in A. The population risk and empirical risk w.r.t J-loss are defined respectively as:

$$R_{\mathcal{D}}[A(S); J] := \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[f_J(A(S), x, y)\right] \quad \text{and} \quad R_S[A(S); J] := \frac{1}{n}\sum_{i=1}^{n} f_J(A(S), x_i, y_i) \tag{2}$$

Note that $R_{\mathcal{D}}[A(S); J]$ and $R_S[A(S); J]$ are both random variables. The expected generalization gap w.r.t the J-loss is then

$$\mathrm{GG}_n(J, A) := \mathbb{E}_{S,A}\left[R_{\mathcal{D}}[A(S); J] - R_S[A(S); J]\right] \tag{3}$$

where expectation over A refers to averaging over the intrinsic randomness in A. Specially, we will call $\mathrm{GG}_n(J^{\mathrm{id}}, A)$ and $\mathrm{GG}_n(J^*, A)$ respectively the *standard generalization* gap and the *robust generalization* gap of the algorithm A.

The generalization gap can be analyzed by exploiting the tool of uniform stability (Bousquet & Elisseeff, 2002). We say that the algorithm A is $\rho-$uniformly stable w.r.t J-loss, if

$$\Delta_n(J, A) := \sup_{S\simeq S'} \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \mathbb{E}_A[f_J\left(A(S), x, y\right) - f_J\left(A(S'), x, y\right)] \leq \rho \tag{4}$$

Here $S \simeq S'$ denotes two datasets that each contains n samples but differ in at most one. It is shown in Hardt et al. (2016) that uniform stability implies generalization in expectation, namely,

Lemma 3.1 (Hardt et al. (2016))**.** *For any perturbation J and any algorithm A,*

$$\mathrm{GG}_n(J, A) \leq \Delta_n(J, A) \tag{5}$$

The lemma is due to that the analysis Hardt et al. (2016) applies to arbitrary loss functions, including the $J-$loss defined above. In our work, we will consider the family of f that are Lipschitz and gradient-Lipschitz with respect to both x and w in the following sense: there exist positive constants $L_{\mathcal{X}}$, $L_{\mathcal{W}}$, $\Gamma_{\mathcal{X}}$ and β such that for any $y \in \mathcal{Y}$, any $x, x' \in \mathcal{X}$ and any $w, w' \in \mathcal{W}$

$$|f(w', x', y) - f(w, x, y)| \leq L_{\mathcal{X}}\|x - x'\| + L_{\mathcal{W}}\|w - w'\| \tag{6}$$

$$\|\nabla_{w'} f(w', x', y) - \nabla_w f(w, x, y)\| \leq \Gamma_{\mathcal{X}}\|x - x'\| + \beta\|w - w'\| \tag{7}$$

Similar Lipschitzness and smoothness assumptions are also used in other stability analysis literature, as in Hardt et al. (2016); Farnia & Ozdaglar (2021); Xiao et al. (2022b); Wang et al. (2024).

With the Lipschitz condition of f, the uniform stability w.r.t f_J can be related with the notion of the uniform argument stability (UAS), a notion coined in Bassily et al. (2020), as well as an "expansiveness" property of J, which we will soon define. Specifically the UAS parameter of A is

$$\delta_n(A) := \sup_{S\simeq S'} \mathbb{E}_A\|A(S) - A(S')\| \tag{8}$$

and for any given $c \geq 0$, we define the c-expansiveness of perturbation J as

$$q_c(J) := \sup_{(x,y)} \sup_{w,w':\|w-w'\|>c} \frac{\|J(x; y, w) - J(x; y, w')\|}{\|w - w'\|} \tag{9}$$

We note that such a notion of expansiveness reduces to a Lipschitz condition when $c = 0$. It measures the sensitivity of an operator to the perturbation of its input, sharing similarity with the Lipschitz condition but provide extra benefit when analyzing operators whose Lipschitz constant is unbounded. When taking $c > 0$, this expansiveness, however, excludes measuring sensitivity for perturbation with magnitude lower than c. This consideration is motivated by the fact that in

practice, extremely small perturbation do not arise. Additionally, this expansiveness behaves nicely, i.e., being bounded, even for non-continuous operators, such as those defined via the sign function, to arise later in this paper.

For any given S and S' differing by only one element and every $c^* \geq 0$, let $Q(S, S'; c^*)$ denote the probability (under the probability measure induced by the randomness in A) that $\|A(S) - A(S')\| < c^*$. Specifically, let S_* and S'_* denote two training sets with $S_* \simeq S'_*$ which achieve the supremum in the definition of $\Delta_n(J, A)$ in (4). We write $Q(c^*)$ in place of $Q(S_*, S'_*; c^*)$ for simplicity.

Lemma 3.2. *If the loss function f satisfies the Lipschitz condition (6), then for any $c^* \geq 0$,*

$$\Delta_n(J, A) \leq (L_{\mathcal{W}} + q_{c^*}(J) L_{\mathcal{X}}) \delta_n(A) + L_{\mathcal{X}} Q(c^*) \cdot 2\epsilon\sqrt{d} \tag{10}$$

The proof of this lemma is deferred to Appendix A. In the remainder of this paper, we will use this bound to analyze the generalization of adversarial training (AT) algorithms. We will show, for most cases, that this bound vanishes with sample size n by choosing a judicious choice of c^*. The only case in which a vanishing bound is not attainable is sign-PGD based AT, where the bound converges to a constant. This may reveal some intrinsic difficulty in generalization for such AT algorithm.

4 Uniform Stability of Adversarial Training

Lemma 3.2 suggests that the expansiveness of the perturbation J, which is used to define the J-loss f_J, plays a role in generalization. We now take A as an *adversarial training (AT) algorithm* where we will show that the expansiveness of the perturbation used in the AT training algorithm A plays another role by impacting the UAS parameter $\delta_n(A)$.

AT algorithms We consider the following iterative AT algorithm. At each iteration of AT, it first draw a training sample $(x_{i_t}, y_{i_t}) \in S$ and then updates the model parameter w_t according to

$$x_t^{\mathrm{adv}} = \pi(x_{i_t}; y_{i_t}, w_t) \tag{11}$$

$$w_{t+1} = w_t - \tau_t \nabla_{w_t} f(w_t, x_t^{\mathrm{adv}}, y_{i_t}) \tag{12}$$

Here $\tau_t \in \mathbb{R}_+$ denotes the step size of the gradient descend at the iteration t, $i_t \in \{1, \cdots, n\}$ is drawn uniformly and independently (across t) from $\{1, 2, \ldots, n\}$, and $\pi(x_{i_t}; y_{i_t}, w_t)$ denotes perturbation of x_{i_t} within $\mathbb{B}_\infty(x_{i_t}, \epsilon)$. We note that ideally π should be $J^*(x_{i_t}; y_{i_t}, w_t)$ but in practice it is only an approximation of it due to the difficulty in acquiring the exact solution. Additionally and more critically, we note that, despite that both π and J refer to perturbations, the two notions in this paper may be completely different. Specifically, J induces the J-loss, which is used as a performance metric (evaluated either on the training set or on the testing set), whereas π denotes the perturbation operation applied during adversarial training. Although in some cases π is J or is related to J, there are scenarios in which π and J are completely decoupled, for example, when we perform adversarial training but choose to evaluate the model using the standard loss, i.e., using J^{id}-loss. In a later section, we will see more cases in which J and π are completely different. As a minor comment, we note that when the perturbation π in (11) is chosen as the identity map J^{id}, the AT algorithm reduces to the standard stochastic gradient descend (SGD) algorithm. Finally, as we may look into various choices of π in AT algorithms, we use A_π to denote an AT algorithm, emphasizing its dependence on π. Under such notations, we may even consider "mis-matched generalization gap", namely, $\mathrm{GG}_n(J, A_\pi)$ with $J \neq \pi$, for example, $J = J^{\mathrm{id}}$ and π is a particular adversarial perturbation.

Note that although x_t^{adv} is also a function of w_t, the derivative operator in (12) does not go through π, an option consistent with the standard AT implementation as in Madry et al. (2019); Rice et al. (2020).

We now present an upper bound for the UAS of AT.

Theorem 4.1. *Suppose that f satisfies the conditions (6) and (7). If we run A_π for T steps with step sizes $\tau_t \leq \frac{1}{\beta}$, there exists a constant $c > 0$ such that we have*

$$\delta_n(A_\pi) \leq \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} (2 + q_c(\pi)\Gamma_{\mathcal{X}}/\beta)^t \tag{13}$$

We defer the proof of the theorem to Appendix A. With the upper bound of the UAS, an upper bound for the mismatched generalization gap can be immediately derived according to (5) and (10) as below:

Theorem 4.2. *Under the condition of Theorem 4.1, for any $c^* \geq 0$, there exists a constant $c > 0$, such that*

$$\mathrm{GG}_n(J, A_\pi) \leq (L_{\mathcal{X}} q_{c^*}(J) + L_{\mathcal{W}}) \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} (2 + q_c(\pi)\Gamma_{\mathcal{X}}/\beta)^t + L_{\mathcal{X}} Q(c^*) \cdot 2\epsilon\sqrt{d} \tag{14}$$

The bound in (14) also includes as a special case the "matched" generalization gap $\mathrm{GG}_n(J, A_J)$, where the perturbation used in adversarial training is identical to that defining performance metric, as is typical in the adversarial training literature. Beyond the Lipschitz and smoothness conditions of f, the expansiveness parameters of π and J turn out to also influence the generalization of AT algorithms, as suggested in the generalization bound (14). This has been overlooked by the previous stability analysis as in Xing et al. (2021); Xiao et al. (2022b); Wang et al. (2024).

The behavior of the bound in (14) clearly depends on $Q(c^*)$. We now show that with additional conditions, one can choose a c^* to either remove the term containing $Q(c^*)$ or make $Q(c^*)$ also vanish with n.

For example, if the perturbation J has bounded Lipschitz constant q^*, that is $q_{c^*}(J) \leq q^* < \infty$ for any $c^* \geq 0$, then taking $c^* = 0$ simply results in the following bound that vanishes as $\mathcal{O}(1/n)$.

$$\mathrm{GG}_n(J, A_\pi) \leq (L_{\mathcal{X}} q^* + L_{\mathcal{W}}) \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} (2 + q_c(\pi)\Gamma_{\mathcal{X}}/\beta)^t \tag{15}$$

On the other hand, if the second moment of the random variable $\|A(S_*) - A(S'_*)\|$ has a fast vanishing rate with n, one can choose c^* to decay with n at a judicious choice of rate, pushing $Q(c^*)$ to vanish faster than $1/n$, resulting in the bound in the following form

$$\mathrm{GG}_n(J, A_\pi) \leq (L_{\mathcal{X}} q_{c^*}(J) + L_{\mathcal{W}}) \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} (2 + q_c(\pi)\Gamma_{\mathcal{X}}/\beta)^t + o(1/n) \tag{16}$$

We defer the proof of (16) to Appendix A.

Convex loss and strongly convex loss When f is further assumed to be convex or strongly convex, a tighter UAS upper bound can be attained.

Theorem 4.3. *Suppose that $f(\cdot, x, y)$ is convex for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and satisfies the conditions (6) and (7). If we run A_π for T steps with step sizes $\tau_t \leq \frac{1}{\beta}$, we have*

$$\delta_n(A_\pi) \leq \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} (1 + q_c(\pi)\Gamma_{\mathcal{X}}/\beta)^t \tag{17}$$

If we further assume $f(\cdot, x, y)$ is $\mu-$strongly convex, we have

$$\delta_n(A_\pi) \leq \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} \left(1 - \frac{\mu}{2\beta} + \Gamma_{\mathcal{X}} q_c(\pi)/\beta\right)^t \tag{18}$$

As shown, performing AT using convex loss functions results in a tighter upper bound compared to the non-convex functions. When f is strongly convex, the bound can be tightened again. In fact, in the strongly convex case, if $q_c(\pi)$ is small enough, the UAS upper bound can be made independent with the number of iteration T.

Corollary 4.4. *Suppose that f is $\mu-$strongly convex and satisfies the conditions (6) and (7). Suppose that $q_c(\pi) < \mu/(2\Gamma_{\mathcal{X}})$ and we run A_π for T steps with step sizes $\tau_t \leq \frac{1}{\beta}$, we have*

$$\delta_n(A_\pi) \leq \frac{4L_{\mathcal{W}}}{n(\mu - 2q_c(\pi)\Gamma_{\mathcal{X}})} \tag{19}$$

The proofs of Theorem 4.3 and Corollary 4.4 are deferred to Appendix A. Notably, when π is chosen as the identity map, we have $q_c(\pi) = 0$ and A_π reduces to the standard SGD algorithm. In this case, our UAS upper bounds matches the bounds in Hardt et al. (2016) up to constants.

Comparison with existing UAS bounds for AT The work in Farnia & Ozdaglar (2021) derives UAS bounds for the AT-like algorithm (refer to as GDmax in their paper) under the assumption that f is strongly concave in $\mathcal{X}$. Our work goes beyond this restricted setting and derive UAS bounds without this assumption. In Xing et al. (2021), the stability of AT is analyzed by treating AT as standard SGD with an adversarial loss (i.e., f^*) and invoke the generic bound in Bassily et al. (2020) for non-smooth losses, while assuming f^* to be non-smooth. The source of the non-smoothness is however not explained in their work; additionally since the bound in Bassily et al. (2020) is developed for SGD with any non-smooth convex functions, it fails to explain the notable difference between SGD and AT observed in practice. The UAS bounds proposed in Xiao et al. (2022b), as well as the bound in Wang et al. (2024), include terms that do not vanish with increasing sample size. Our bounds overcome this limitation, vanishing with the sample size.

5 Revisit of PGD-based AT

We now discuss the AT algorithm A_π when π is taken as the PGD perturbation (Madry et al., 2019), which we denote by π^{PGD}. To begin, associated with any (x, y) and any weight parameter w, we define and one-step PGD map $T_{x,y,w}$ by

$$T_{x,y,w}(x') = \Pi_{\mathbb{B}_\infty(x,\epsilon)}\left[x' + \lambda G\left(\nabla_{x'} f(w, x', y)\right)\right] \tag{20}$$

Here x' is any point in $\mathbb{R}^d$, G is a mapping from $\mathbb{R}^d$ to $\mathbb{R}^d$, possibly taking various forms, which we will specify momentarily, λ is another step size, and $\Pi_{\mathbb{B}_\infty(x,\epsilon)} : \mathbb{R}^d \to \mathbb{B}_\infty(x, \epsilon)$ denotes the projection onto the set $\mathbb{B}_\infty(x, \epsilon)$, namely, $\Pi_{\mathbb{B}_\infty(x,\epsilon)}(x') = \arg\min_{\tilde{x}\in\mathbb{B}_\infty(x,\epsilon)} \|\tilde{x} - x'\|_2$. The K-step PGD perturbation π^{PGD} is then defined as the K-fold compositions of the (same) mapping $T_{x,y,w}$:

$$\pi^{\mathrm{PGD}}(x; y, w) := T^K_{x,y,w}(x) := \left(\underbrace{T_{x,y,w} \circ T_{x,y,w} \circ \ldots T_{x,y,w}}_{K \text{ times}}\right)(x) \tag{21}$$

In the well-known PGD attack (Madry et al., 2019), the mapping G is taken as the sign function and is applied element-wisely on the gradients(see Wong et al. (2020); Andriushchenko & Flammarion (2020); Wang et al. (2021); Rice et al. (2020); Dong et al. (2021); Wu et al. (2020)), Theoretical analyses of PGD (as in Deng et al. (2020); Fu & Wang (2023); Bubeck et al. (2015)) often considers the "raw-gradient" version, namely taking G as the identity map. In our work, we will show that the choice of G, this peculiar and largely overlooked building block in PGD, in fact has non-negligible impact on the generalization performance of PGD-based AT.

To begin, we assume that the gradient $\nabla_x f$ is Lipschitz, namely, that there exist positive constants η and $\Gamma_{\mathcal{W}}$ such that for any $y \in \mathcal{Y}$, any $x, x' \in \mathcal{X}$ and any $w, w' \in \mathcal{W}$

$$\|\nabla_{x'} f(w', x', y) - \nabla_x f(w, x, y)\| \le \eta\|x - x'\| + \Gamma_{\mathcal{W}}\|w - w'\| \tag{22}$$

Lemma 5.1. *(Expansiveness of PGD) Suppose that f satisfies the condition (22) and the mapping G is $\alpha-$ Lipschitz.*

$$q_c(\pi^{\mathrm{PGD}}) \le \min\left(\sum_{i=0}^{K-1} \mu^i \nu, \frac{2\sqrt{d}\epsilon}{c}\right) \tag{23}$$

where $\nu = \lambda\alpha\Gamma_{\mathcal{W}}$ and $\mu = 1 + \lambda\alpha\eta$.

We defer the proof to Appendix A.

For all J-losses for which $q_c(J)$ is uniformly bounded by q^*, plugging this bound to (15) immediately gives a generalization bound that vanishes as $\mathcal{O}(1/n)$. However, one of the most important J-loss, the one defined using sign-PGD attack, fails to satisfy this boundedness condition and the bound (15) does not apply.

To carefully study such a setting, let $J^{\mathrm{sign-PGD}} := \pi^{\mathrm{sign-PGD}}$, where $\pi^{\mathrm{sign-PGD}}$ is π^{PGD} with function G taken as the sign function. We have the following results.

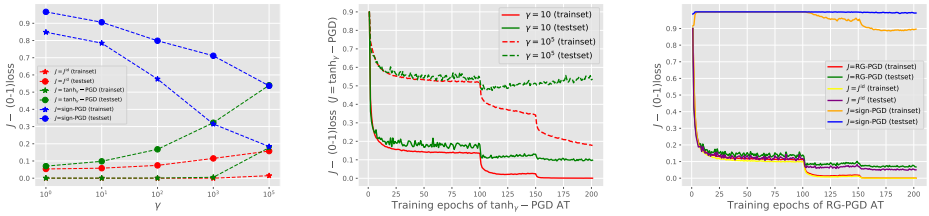
Corollary 5.2. *Let $J = J^{\mathrm{sign-PGD}}$. Suppose that for any S and S' with $S \simeq S'$, $\|A(S) - A(S')\| < B$ with probability 1. Under the condition of Theorem 4.1, for any $\rho > 0$, there exists some N (depending on ρ), such that when $n > N$,*

$$\mathrm{GG}_n(J, A_\pi) < (1 - \delta_n(A_\pi)/B)\, L_{\mathcal{X}} \cdot 2\epsilon\sqrt{d} + \rho. \tag{24}$$

The proof is left in Appendix A. Note that for $J^{\mathrm{sign-PGD}}$-loss and without additional information on $\|A(S) - A(S')\|$, it appears difficult to arrive at a generalization bound that vanishes with n and the bound given here converges to a constant. Although this may not mean that AT with $J^{\mathrm{sign-PGD}}$-loss does not have a vanishing generalization error, it nonetheless reveals certain intrinsic difficulty of generalization for this setting. Specifically, for large n, the perturbation radius (that defines the J-loss) and the input dimension appear to fight against the UAS parameter $\delta_n(A_\pi)$; when UAS parameter decreases – which pushes towards better generalization, $\epsilon\sqrt{d}$ is amplified more significantly – causing poorer generalization.

To investigate how the expansiveness property affects generalization, we consider a smooth approximation of the sign function by a $\tanh$ function, i.e., $\mathrm{sgn}(x) \approx \tanh_\gamma(x) := \tanh(\gamma x)$. Notably, the approximation error here vanishes with increase γ. By replacing $\mathrm{sgn}(x)$ in PGD AT with $\tanh_\gamma(x)$, we may control the expansiveness of π^{PGD}.

Experiments We conduct experiment for PGD-AT when G is chosen as $\tanh_\gamma$ as well as the identity map. Specially, for π^{PGD} with different choice of G, we refer to it as "sign-PGD" when $G(x) = \mathrm{sgn}(x)$, as "$\tanh_\gamma$-PGD" when $G(x) = \tanh_\gamma(x)$ and as "raw gradient (RG)-PGD" when $G(x) = x$. In all the experiments, we primarily consider the J-loss defined in (1) as our evaluation metric, with the loss function in f taken as the 0-1 loss and refer to this metric as J-(0-1) loss. We mainly use J from $\{\tanh_\gamma\text{-PGD}, \text{sign-PGD}, J^{\mathrm{id}}\}$. The experiments are conducted on CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009) and SVHN(Netzer et al., 2011). Our experimental setting is elaborated in Appendix B, which follows from the setting in Rice et al. (2020).



(a) (b) (c)

Figure 2: Experiments on CIFAR-10. (a) Models trained with $\tanh_\gamma$-PGD AT with different γ and evaluated by J-(0-1) loss on the training and testing set. (b) J-(0-1) loss with $J = \tanh_\gamma$-PGD measured along the training trajectories of two sets of $\tanh_\gamma$-PGD AT. (c) J-(0-1) loss measured along the trajectory of the RG-PGD AT with different choice of J.

Figure 2 (a) presents the results of experiments conducted on CIFAR10, where the models are trained using $\tanh_\gamma$-PGD AT (i.e., A_π with $\pi = \tanh_\gamma$-PGD) with various γ values. Each model is trained for 200 epochs and is evaluated using the J-(0-1) loss for $J \in \{\tanh_\gamma\text{-PGD}, \text{sign-PGD}, J^{\mathrm{id}}\}$ (distinguished by colors), where γ matches the corresponding value in π. We use star-shaped dots and circle-shaped dots to respectively denote the J-(0-1) loss measured on the training set and the testing set. The gaps between each pairs of curves in the same color category then represents the generalization gap of the trained models evaluated by different J-(0-1) loss. By decreasing γ in π, the generalization gaps reduce, as shown by the narrowing gaps across all pairs of the curves in the same color. The observed experimental results demonstrate that AT with less expansive π tends to achieve a smaller generalization gap, consistent with the generalization bound of (14). Similar trends are also observed on SVHN and CIFAR100 (see Appendix D Figure 4).

Due to the mismatch between π and J, the model trained by the algorithm A_π may still have a large empirical risk $\mathbb{E}[R_S[A_\pi(S), J]]$, which in turn results in a high population risk $\mathbb{E}[R_{\mathcal{D}}[A_\pi(S), J]]$ even if the generalization gap $\mathrm{GG}_n(J, A_\pi)$ is small. This is illustrated in Figure 2 (a) as the blue star-shaped curve consistently stays higher than the green star-shaped curve with a notably large gap. As γ increases, the $\tanh_\gamma$ function gradually approaches the sign function, leading to an intersection of the green and the blue curves. This indicates that sign-PGD is a stronger perturbation compare to the $\tanh_\gamma$-PGD, as the model trained with $\tanh_\gamma$-PGD AT can still be vulnerable to the sign-PGD attack on the training set.

The seminal work by Tsipras et al. (2018) found that AT can negatively impact standard generalization. They constructed specific data models to demonstrate that achieving robustness and standard generalization can be inherently conflicting, suggesting an unavoidable trade-off between these two goals. This phenomenon has been extensively studied in subsequent research (Zhang et al., 2020; 2019; Yang et al., 2020; Raghunathan et al., 2020; Javanmard et al., 2020; Pang et al., 2022). Our experimental results offer further insights into this phenomenon from the perspective of algorithmic stability. Specifically, we find that the decline in standard generalization performance caused by AT can be attributed to the poor expansiveness condition of the sign-PGD method employed in AT. As shown by the trend of the red circle-shaped curve in Figure 2 (a), AT does not always harm standard generalization; a reduction in the J^{id}-(0-1) loss is observed as γ decreases. This suggests that the trade-off identified by Tsipras et al. (2018) might be a side effect of the sign-PGD AT and is not necessarily unavoidable.

Figure 2 (b) plots the J-(0-1) loss with $J = \pi$ evaluated along the trajectory of the $\tanh_\gamma$-PGD AT with $\gamma = 10$ (the solid curves) and $\gamma = 10^5$ (the dashed curves). The dashed curves exhibit a phenomenon similar to robust overfitting observed in Rice et al. (2020): after the first learning rate decay (the 100th epoch), as the training loss continuously decreases, the testing loss starts to elevate. This phenomenon does not appear in the AT with $\gamma = 10$, as shown in the trend of the solid curves. We conduct additional experiments for RG-PGD AT. As shown in Figure 2 (c), the generalization gap remains small across all groups of J-(0-1) loss throughout the training. Similar to the previous results, the model trained by this AT variant exhibits notable vulnerability to the sign-PGD perturbation, as indicated by the consistently high values of the orange and blue curves. These findings demonstrate that removing or altering the sign function in PGD leads to a non-negligible influence on both robust generalization and resistance to perturbations on the training set. This highlights the crucial role of the sign function in PGD-AT, which deserves a more careful and further in-depth investigation.

6 Revisit of sign function in PGD

For simplicity, we write $f(w, x, y)$ as $f(x)$ hereafter. The sign-PGD perturbation can be treated as an iterative optimization algorithm for solving the constrained optimization problem $\max_{\hat{x}\in\mathbb{B}_\infty(x,\epsilon)} f(\hat{x})$. It is related to the sign gradient methods, which has been used for different purposes, such as for training neural networks (e.g., Riedmiller & Braun (1992)) and for gradient compression (e.g., Bernstein et al. (2018)).

We now show that the sign gradient method can be viewed as a Steepest Descend (or ascend in our context) Method (SDM) w.r.t a $\infty-$norm ball (e.g., see Chapter 9.4 in Boyd & Vandenberghe (2004)). Specifically, for the loss $f(x^k)$ at the kth iteration in SDM, it updates x^k by finding a steepest ascend direction v within a small neighborhood of x^k such that the loss $f(x^{k+1})$ with $x^{k+1} = x^k + v$ is locally maximized. Such a neighborhood can be chosen as a $p-$norm ball around x^k (i.e., $\mathbb{B}_p(x^k, \lambda_p)$) with a small radius λ_p. Finding v introduces a new optimization problem: $\max_{v\in\mathbb{B}_p(x^k,\lambda_p)} f(x^k+v)$, which is then approximately solved by replacing $f(x^k+v)$ with its linear approximation around x^k, namely, solving $\max_{v\in\mathbb{B}_p(x^k,\lambda_p)} f(x^k) + \nabla f(x^k)^T v$ which is equivalent to solving $\max_{v\in\mathbb{B}_p(x^k,\lambda_p)} \nabla f(x^k)^T v$ whose closed form solution is

$$v^* = \lambda_p G_p(\nabla f(x^k)), \quad \text{where} \quad G_p(\nabla f(x^k)) := \frac{\operatorname{sgn}(\nabla f(x^k)) \odot |\nabla f(x^k)|^{q-1}}{\|\nabla f(x^k)\|_q^{q-1}} \tag{25}$$

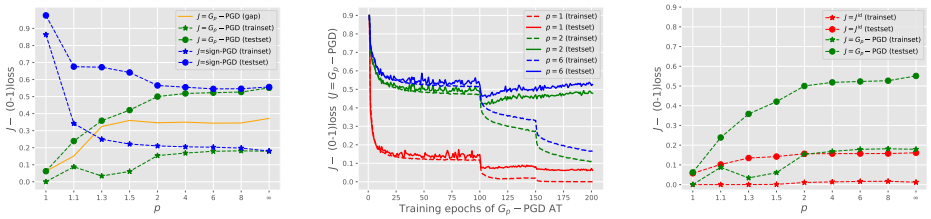
where we require $1/p + 1/q = 1$. The operator $\odot$ denotes the element-wise product. The closed form (25) then gives the following updating rule of SDM as

$$x^{k+1} = x^k + \lambda G_p(\nabla f(x^k)) \tag{26}$$

As a special case, when $p = 1$ with $q = \infty$, SMD turns into the coordinate gradient method with $G_1(\nabla f(x^k)) = \operatorname{sgn}(\max_i \nabla f(x^k)[i])e_i$ and $i = \arg\max_j |\nabla f(x^k)[j]|$, where e_i denotes the standard basis vector. When $p = q = 2$, we have $G_2(\nabla f(x^k)) = \nabla f(x^k)/\|\nabla f(x^k)\|_2$

When $p = \infty$ with $q = 1$, the mapping G_∞ reduces to the sign function, indicating that the sign-PGD is indeed a (projected) SDM w.r.t $\mathbb{B}_\infty(x^k, \lambda_\infty)$. It is then curious to investigate *the generalization performance of the model trained by AT using the G_p-PGD with $p \neq \infty$.*[1]. We conduct

[1]Note that in the G_p-PGD we still consider projecting onto $\mathbb{B}_\infty(x, \epsilon)$ when p is taken other than ∞.



(a) (b) (c)

Figure 3: Experiments for G_p-PGD AT: (a) Model trained with various p values and evaluated by J-(0-1) loss with $J = \pi$ and $J =$ sign-PGD. (b) Training curves of the AT with various p values. (c) Standard generalization performance of the models trained by the AT, where the green curves are copied from (a) for a clearer presentation.

experiments for the G_p-PGD-based AT following the same experimental setting as in the previous section, except that λ_p is adjusted to maintain the same volume of the balls $\mathbb{B}_p(0, \lambda_p)$ across different p values (details in Appendix C). Figure 3 (a) presents the experimental results on CIFAR10 (results on the other datasets are in Appendix D Figure 5 and 6). The models are trained by A_π with $\pi = G_p$-PGD for various p and are evaluated by the J-(0-1) loss with $J = \pi$ (green curves) as well as $J =$sign-PGD (blue curves). The yellow curve represents the generalization gap for models trained with G_p-PGD. As shown, a larger p tends to result in larger generalization gaps. Indeed, nearly all G_p-PGD with $p \geq 1.3$ cause notably overfitting in AT with generalization gaps exceeding 30%. The consistently higher position of blue star-shaped curves over the green star-shaped curve also suggests that sign-PGD is the strongest perturbations among the G_p-PGD. Figure 3 (b) further exhibits the overfitting in G_p-PGD AT by plotting training curves for $p = \{1, 2, 6\}$, where continued training causes a rise of the testing errors (the blue and green curves), in contrast with the red curves, which demonstrate a good generalization. Figure 3 (c) shows how the G_p-PGD AT affect standard generalization where the red curves deontes the J-(0-1) loss with $J = J^{\mathrm{id}}$ and the green curves are copied from Figure 3 (a) for a clearer comparison. An enlarging standard generalization gap is also observed in G_p-PGD AT with larger p.

The observed overfitting caused by the G_p-PGD family is potentially attributed to that nearly all the members in $\{G_p : p \in [1, \infty]\}$ have a poor Lipschitzness, as shown in the following lemma, which leads to a bad expansiveness of G_p-PGD.

Lemma 6.1. *Consider the mapping $G_p : \mathbb{R}^d \to \mathbb{R}^d$ specified in (25) with $p \in [1, \infty]$. Let $\mathcal{I} := \{1, \cdots, d\}$. If G_p is α_p-Lipschitz over the set $H(r) \subseteq \mathbb{R}^d$ with $H(r) := \{x \in \mathbb{R}^d : \min_{i \in \mathcal{I}} |x[i]| \geq r\}$ for some $r > 0$, then we have*

$$\alpha_p \geq \frac{1}{r d^{\frac{1}{p}}} \tag{27}$$

We defer the proof in Appendix A. This lower bound also implies that α_p is unbounded in $\mathbb{R}^d$, noting that the lower bound approaches infinity as $r \to 0$. Except for this extreme case, it is reasonable to assume that the gradients $\nabla_x f(x)$ lies in a set $H(r)$ with sufficiently small r where all the members in $\{G_p : p \in [1, \infty]\}$ have a bounded but large Lipschitz constant. Noteworthy, the lower bound increases, as p ranges from 1 to infinity, suggesting that the increased generalization gap in Figure 3 (a) is attributed to the increasing expansiveness of G_p-PGD caused by the rise in α_p.

7 Limitations

The main limitation of this work is that we have only developed an upper bound for the generalization of AT algorithms. Like all up-bound based theoretical results, such an approach is adequate for understanding performance guarantees but may be inadequte to explain poor generalization. Nonetheless, our experimental results have suggested that our upper bound may well explain robust overfitting.

REFERENCES

Maksym Andriushchenko and Nicolas Flammarion. Understanding and improving fast adversarial training. *Advances in Neural Information Processing Systems*, 33:16048–16059, 2020.

Idan Attias, Aryeh Kontorovich, and Yishay Mansour. Improved generalization bounds for robust learning. *CoRR*, abs/1810.02180, 2018. URL `http://arxiv.org/abs/1810.02180`.

Pranjal Awasthi, Natalie Frank, and Mehryar Mohri. Adversarial learning guarantees for linear hypotheses and neural networks. *CoRR*, abs/2004.13617, 2020. URL `https://arxiv.org/abs/2004.13617`.

Raef Bassily, Vitaly Feldman, Cristóbal Guzmán, and Kunal Talwar. Stability of stochastic gradient descent on nonsmooth convex losses. *Advances in Neural Information Processing Systems*, 33: 4381–4391, 2020.

Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signsgd: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pp. 560–569. PMLR, 2018.

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

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

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

Tianlong Chen, Zhenyu Zhang, Sijia Liu, Shiyu Chang, and Zhangyang Wang. Robust overfitting may be mitigated by properly learned smoothening. In *International Conference on Learning Representations*, 2020.

Yuansi Chen, Chi Jin, and Bin Yu. Stability and convergence trade-off of iterative optimization algorithms. *arXiv preprint arXiv:1804.01619*, 2018.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020.

Daniel Cullina, Arjun Nitin Bhagoji, and Prateek Mittal. Pac-learning in the presence of evasion adversaries, 2018.

Zhun Deng, Hangfeng He, Jiaoyang Huang, and Weijie Su. Towards understanding the dynamics of the first-order adversaries. In *International Conference on Machine Learning*, pp. 2484–2493. PMLR, 2020.

Dimitrios I. Diochnos, Saeed Mahloujifar, and Mohammad Mahmoody. Lower bounds for adversarially robust PAC learning. *CoRR*, abs/1906.05815, 2019. URL `http://arxiv.org/abs/1906.05815`.

Yinpeng Dong, Ke Xu, Xiao Yang, Tianyu Pang, Zhijie Deng, Hang Su, and Jun Zhu. Exploring memorization in adversarial training. *arXiv preprint arXiv:2106.01606*, 2021.

Farzan Farnia and Asuman Ozdaglar. Train simultaneously, generalize better: Stability of gradient-based minimax learners. In *International Conference on Machine Learning*, pp. 3174–3185. PMLR, 2021.

Shaopeng Fu and Di Wang. Theoretical analysis of robust overfitting for wide dnns: An ntk approach. *arXiv preprint arXiv:2310.06112*, 2023.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International conference on machine learning*, pp. 1225–1234. PMLR, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. *CoRR*, abs/1603.05027, 2016. URL http://arxiv.org/abs/1603.05027.

Adel Javanmard, Mahdi Soltanolkotabi, and Hamed Hassani. Precise tradeoffs in adversarial training for linear regression. In *Conference on Learning Theory*, pp. 2034–2078. PMLR, 2020.

Justin Khim and Po-Ling Loh. Adversarial risk bounds via function transformation, 2019.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Ilja Kuzborskij and Christoph Lampert. Data-dependent stability of stochastic gradient descent. In *International Conference on Machine Learning*, pp. 2815–2824. PMLR, 2018.

Yunwen Lei and Yiming Ying. Fine-grained analysis of stability and generalization for stochastic gradient descent. In *International Conference on Machine Learning*, pp. 5809–5819. PMLR, 2020.

Yunwen Lei, Zhenhuan Yang, Tianbao Yang, and Yiming Ying. Stability and generalization of stochastic gradient methods for minimax problems. In *International Conference on Machine Learning*, pp. 6175–6186. PMLR, 2021.

Binghui Li, Jikai Jin, Han Zhong, John Hopcroft, and Liwei Wang. Why robust generalization in deep learning is difficult: Perspective of expressive power. *Advances in Neural Information Processing Systems*, 35:4370–4384, 2022.

Chen Liu, Mathieu Salzmann, Tao Lin, Ryota Tomioka, and Sabine Süsstrunk. On the loss landscape of adversarial training: Identifying challenges and how to overcome them. *Advances in Neural Information Processing Systems*, 33:21476–21487, 2020.

Chen Liu, Zhichao Huang, Mathieu Salzmann, Tong Zhang, and Sabine Süsstrunk. On the impact of hard adversarial instances on overfitting in adversarial training. *arXiv preprint arXiv:2112.07324*, 2021.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019.

Omar Montasser, Steve Hanneke, and Nathan Srebro. VC classes are adversarially robustly learnable, but only improperly. *CoRR*, abs/1902.04217, 2019. URL http://arxiv.org/abs/1902.04217.

Wenlong Mou, Liwei Wang, Xiyu Zhai, and Kai Zheng. Generalization bounds of sgld for non-convex learning: Two theoretical viewpoints. In *Conference on Learning Theory*, pp. 605–638. PMLR, 2018.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

Tianyu Pang, Min Lin, Xiao Yang, Jun Zhu, and Shuicheng Yan. Robustness and accuracy could be reconcilable by (proper) definition. In *International Conference on Machine Learning*, pp. 17258–17277. PMLR, 2022.

Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through local linearization. *Advances in neural information processing systems*, 32, 2019.

Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John Duchi, and Percy Liang. Understanding and mitigating the tradeoff between robustness and accuracy. *arXiv preprint arXiv:2002.10716*, 2020.

Leslie Rice, Eric Wong, and J. Zico Kolter. Overfitting in adversarially robust deep learning. *CoRR*, abs/2002.11569, 2020. URL https://arxiv.org/abs/2002.11569.

Martin Riedmiller and Heinrich Braun. Rprop: a fast adaptive learning algorithm. In *Proc. of the Int. Symposium on Computer and Information Science VII*, 1992.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. *Advances in neural information processing systems*, 31, 2018.

Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *Advances in neural information processing systems*, 32, 2019.

Uri Shaham, Yutaro Yamada, and Sahand Negahban. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. *Neurocomputing*, 307:195–204, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2014.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *arXiv preprint arXiv:1805.12152*, 2018.

Yihan Wang, Shuang Liu, and Xiao-Shan Gao. Data-dependent stability analysis of adversarial training. *arXiv preprint arXiv:2401.03156*, 2024.

Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. On the convergence and robustness of adversarial training. *arXiv preprint arXiv:2112.08304*, 2021.

Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. In *International Conference on Machine Learning (ICML)*, 2023.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

Dongxian Wu, Yisen Wang, and Shutao Xia. Revisiting loss landscape for adversarial robustness. *CoRR*, abs/2004.05884, 2020. URL `https://arxiv.org/abs/2004.05884`.

Jiancong Xiao, Yanbo Fan, Ruoyu Sun, and Zhi-Quan Luo. Adversarial rademacher complexity of deep neural networks, 2022a.

Jiancong Xiao, Yanbo Fan, Ruoyu Sun, Jue Wang, and Zhi-Quan Luo. Stability analysis and generalization bounds of adversarial training. *Advances in Neural Information Processing Systems*, 35:15446–15459, 2022b.

Yue Xing, Qifan Song, and Guang Cheng. On the algorithmic stability of adversarial training. *Advances in neural information processing systems*, 34:26523–26535, 2021.

Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Ruslan Salakhutdinov, and Kamalika Chaudhuri. Adversarial robustness through local lipschitzness. *CoRR*, abs/2003.02460, 2020. URL `https://arxiv.org/abs/2003.02460`.

Dong Yin, Kannan Ramchandran, and Peter L. Bartlett. Rademacher complexity for adversarially robust generalization. *CoRR*, abs/1810.11914, 2018. URL `http://arxiv.org/abs/1810.11914`.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016. URL `http://arxiv.org/abs/1605.07146`.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.

Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan S. Kankanhalli. Attacks which do not kill training make adversarial learning stronger. *CoRR*, abs/2002.11242, 2020. URL `https://arxiv.org/abs/2002.11242`.

A PROOFS

Proof of Lemma 3.2

$$\Delta_n(A, f_J) = \sup_{S \simeq S'} \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \mathbb{E}_A[f\left(A(S), J(x; y, A(S)), y\right) - f\left(A(S'); J(x; y, A(S')), y\right)] \tag{28}$$

$$= \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \mathbb{E}_A[f\left(A(S_*), J(x; y, A(S_*)), y\right) - f\left(A(S'_*); J(x; y, A(S'_*)), y\right)] \tag{29}$$

$$\leq \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \mathbb{E}_A[L_{\mathcal{X}}\|J(x; y, A(S_*)) - J(x; y, A(S'_*))\| + L_{\mathcal{W}}\|A(S_*) - A(S'_*)\|] \tag{30}$$

$$= \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \mathbb{E}_A L_{\mathcal{X}}\|J(x; y, A(S_*)) - J(x; y, A(S'_*))\| + L_{\mathcal{W}}\mathbb{E}_A\|A(S_*) - A(S'_*)\| \tag{31}$$

$$\leq \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \mathbb{E}_A L_{\mathcal{X}}\|J(x; y, A(S_*)) - J(x; y, A(S'_*))\| + L_{\mathcal{W}} \sup_{S\simeq S'} \mathbb{E}_A\|A(S) - A(S')\| \tag{32}$$

The inequality (30) is derived based on the condition (6). We now deal with the first term in (32).

For shorter notation, let $D(S_*, S'_*) := \|A(S_*) - A(S'_*)\|$. For any number $c^* \geq 0$, let $Q(S_*, S'_*; c^*) := \Pr(D(S_*, S'_*) < c^*)$. For any x, y we have

$$\mathbb{E}_A[L_{\mathcal{X}}\|J(x; y, A(S_*)) - J(x; y, A(S'_*))\|] \tag{33}$$

$$=(1 - Q(S_*, S'_*; c^*))\mathbb{E}_A\left[L_{\mathcal{X}}\|J(x; y, A(S_*)) - J(x; y, A(S'_*))\| \mid D(S_*, S'_*) \geq c^*\right] \tag{34}$$

$$+Q(S_*, S'_*; c^*)\mathbb{E}_A[L_{\mathcal{X}}\|J(x; y, A(S_*)) - J(x; y, A(S'_*))\| \mid D(S_*, S'_*) < c^*] \tag{35}$$

$$\leq(1 - Q(S_*, S'_*; c^*))\mathbb{E}_A[q_{c^*}(J)L_{\mathcal{X}}D(S_*, S'_*) \mid D(S_*, S'_*) \geq c^*] + Q(S_*, S'_*; c^*)L_{\mathcal{X}}2\epsilon\sqrt{d} \tag{36}$$

$$\leq q_{c^*}(J)L_{\mathcal{X}}\mathbb{E}_A D(S_*, S'_*) + Q(S_*, S'_*; c^*)L_{\mathcal{X}}2\epsilon\sqrt{d} \tag{37}$$

$$\leq q_{c^*}(J)L_{\mathcal{X}} \sup_{S\simeq S'} \mathbb{E}_A D(S, S') + Q(S_*, S'_*; c^*)L_{\mathcal{X}}2\epsilon\sqrt{d} \tag{38}$$

The derivation above start by splitting the expectation into two conditional expectations conditioned on two complementary events (see the terms (34) and (35)) and then utilize the $c-$expansiveness property of J as well as the condition that $J(x, y, w) \in \mathbb{B}_\infty(x, \epsilon)$ to individually derive the first and second terms in (36). Plug the final expression above back in (32), the lemma is proved. □

Proof of the Theorem 4.1 Consider the AT algorithm specified in (11) and (12). For two datasets S and S' differing in only one sample and respectively containing n samples, let $\{w_t\}_{t=1}^T$ and $\{w'_t\}_{t=1}^T$ respectively denote the sequences of model parameters generated by running AT on S and S' for T iterations. Let c denote the smallest non-zero value of $\|w_t - w'_t\|$ across t and across the randomness of A when running AT algorithm A on S and S'. (Note that such a choice of c may be overly pessimistic, but it suffices to obtain the desired rate of vanishing of the generalization bound in this theorem). For arbitrary iteration $t \in \{1, \cdots, T-1\}$, we have

$$\begin{aligned}
&\mathbb{E}\|w_{t+1} - w'_{t+1}\| \\
\leq&\mathbb{E}\|w_t - \tau_t\nabla_{w_t} f\left(w_t, \pi(x; y, w_t), y\right) + \tau_t\nabla_{w'_t} f\left(w'_t, \pi(x; y, w_t), y\right) - w'_t\| \\
&+ \mathbb{E}\|\tau_t\nabla_{w'_t} f\left(w'_t, \pi(x'; y', w'_t), y'\right) - \tau_t\nabla_{w'_t} f\left(w'_t, \pi(x; y, w_t), y\right)\|
\end{aligned} \tag{39}$$

Here the expectation is taken over all the randomness in w_t and w'_t. We use (x, y) and (x', y') respectively to denote the samples selected by the AT algorithm from S and S' at the iteration t. Inequality (39) is derived by adding and subtracting the term $\tau_t\nabla_{w'_t} f\left(w'_t, \pi(x; y, w_t), y\right)$ and then applying the triangle inequality. For the first term in (39), we have that

$$\begin{aligned}
&\mathbb{E}\|w_t - \tau_t\nabla_{w_t} f\left(w_t, \pi(x; y, w_t), y\right) + \tau_t\nabla_{w'_t} f\left(w'_t, \pi(x; y, w_t), y\right) - w'_t\| \\
&\leq \mathbb{E}\|w_t - w'_t\| + \tau_t\beta\mathbb{E}\|w_t - w'_t\|
\end{aligned} \tag{40}$$

by utilizing the triangle inequality and the condition (7). To deal with the second term in (39), we consider that at each iteration, with probability $1 - 1/n$ the samples selected by AT respectively

from S and S' are the same. We have

$$\begin{aligned}
&\mathbb{E}\|\tau_t \nabla_{w_t'} f\left(w_t', \pi(x'; y', w_t'), y'\right) - \tau_t \nabla_{w_t'} f\left(w_t', \pi(x; y, w_t), y\right)\| \\
&\leq \left(1 - \frac{1}{n}\right) \tau_t \Gamma_{\mathcal{X}} \mathbb{E}\|\pi(x; y, w_t') - \pi(x; y, w_t)\| + \frac{2\tau_t L_{\mathcal{W}}}{n} && (41)\\
&\leq \left(1 - \frac{1}{n}\right) \tau_t \Gamma_{\mathcal{X}} q_c(\pi) \mathbb{E}\|w_t - w_t'\| + \frac{2\tau_t L_{\mathcal{W}}}{n} && (42)
\end{aligned}$$

The first term in (41) and (43) make use of the condition (7) and then the expansiveness condition of π. Since f is $L_{\mathcal{W}}-$ Lipschitz w.r.t $\mathcal{W}$, we have $\|\nabla_w f(w; x, y)\| \leq L_{\mathcal{W}}$ for $\forall x, y, w$. The second term in (41) then follows.

Putting together and considering the step sizes $\tau_t \leq \frac{1}{\beta}$, we have

$$\begin{aligned}
&\mathbb{E}\|w_{t+1} - w_{t+1}'\| \\
&\leq (1 + \beta\tau_t + (1 - 1/n)\Gamma_{\mathcal{X}} q_c(\pi)\tau_t)\, \mathbb{E}\|w_t - w_t'\| + \frac{2\tau_t L_{\mathcal{W}}}{n} && (43)\\
&\leq (1 + \beta\tau_t + \Gamma_{\mathcal{X}} q_c(\pi)\tau_t)\, \mathbb{E}\|w_t - w_t'\| + \frac{2\tau_t L_{\mathcal{W}}}{n} && (44)\\
&\leq (2 + \Gamma_{\mathcal{X}} q_c(\pi)/\beta)\, \mathbb{E}\|w_t - w_t'\| + \frac{2L_{\mathcal{W}}}{n\beta} && (45)
\end{aligned}$$

Unravelling the recursion, we have

$$\mathbb{E}\|w_T - w_T'\| \leq \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} \zeta^t \tag{46}$$

where we take $\zeta = 2 + \Gamma_{\mathcal{X}} q_c(\pi)/\beta$. □

Proof of (16) Let $a > 2$ be a constant. For shorter notation let $Z = \|A_\pi(S_*) - A_\pi(S_*')\|$. We will show that if the second moment $\mathbb{E}Z^2 = \mathcal{O}(\frac{1}{n^a})$, we can take $c^* = \mathbb{E}Z - t$ with $t = \Omega(\frac{1}{n^b})$ and $b \in (1, a/2)$, such that the probability $Q(c^*)$ decay at the rate of $\frac{1}{n^{a-2b}}$. This is due to that

$$\begin{aligned}
Q(c^*) &= \Pr\left[Z \leq c^*\right] && (47)\\
&= \Pr\left[Z \leq \mathbb{E}Z - t\right] && (48)\\
&\leq \Pr\left[t \leq |Z - \mathbb{E}Z|\right] && (49)\\
&\leq \frac{\mathrm{Var}(Z)}{t^2} && (50)\\
&\leq \frac{\mathbb{E}Z^2}{t^2} && (51)\\
&\leq \mathcal{O}\left(\frac{1/n^a}{1/n^{2b}}\right) = \mathcal{O}\left(\frac{1}{n^{a-2b}}\right) && (52)
\end{aligned}$$

where the inequality (50) is based on the Chebyshev's inequality. Note that such a choice of t will guarantee that $c^* > 0$ such that the derivation above is nontrivial. This is because Theorem 4.1 implies that $\mathbb{E}Z \leq \delta_n(A_\pi) = \mathcal{O}(\frac{1}{n})$ and therefore $c^* = \mathcal{O}(\frac{1}{n} - \frac{1}{n^b})$. Taking $b > 1$ guarantees that $c^* > 0$.

Proof of the Theorem 4.3 and Corollary 4.4 The proof is based on a slight modification of the proof in Theorem 4.1. We start from the inequality (39). For the first term in (39), since that the loss function f is convex and $\tau_t \leq 1/\beta < 2/\beta$, according to Lemma 3.7.2 in Hardt et al. (2016), we have

$$\begin{aligned}
&\mathbb{E}\|w_t - \tau_t \nabla_{w_t} f\left(w_t, \pi(x; y, w_t), y\right) + \tau_t \nabla_{w_t'} f\left(w_t', \pi(x; y, w_t), y\right) - w_t'\| \\
&\leq \mathbb{E}\|w_t - w_t'\| && (53)
\end{aligned}$$

When f is further assumed to be $\mu-$ strongly convex, we have that $\mu \leq \beta$ since f is also $\beta-$smooth, implying that $\tau_t \leq \frac{1}{\beta} \leq \frac{2}{\beta+\mu}$. According to Lemma 3.7.3 in Hardt et al. (2016), we have inequality

(54) as

$$\begin{aligned}
&\mathbb{E}\|w_t - \tau_t \nabla_{w_t} f\left(w_t, \pi(x; y, w_t), y\right) + \tau_t \nabla_{w_t'} f\left(w_t', \pi(x; y, w_t), y\right) - w_t'\| \\
&\leq \left(1 - \frac{\beta\mu\tau_t}{\beta+\mu}\right) \mathbb{E}\|w_t - w_t'\| \qquad (54)\\
&\leq \left(1 - \frac{1}{2}\tau_t\mu\right) \mathbb{E}\|w_t - w_t'\| \qquad (55)
\end{aligned}$$

In fact, since $\mu \leq \beta$, we also have $1 \leq \frac{2\beta}{\beta+\mu}$ and thus $\tau_t\mu \leq \frac{2\tau_t\mu\beta}{\beta+\mu}$ with $\tau_t\mu \leq 1$. The inequality (54) can be further simplified as (55).

The second term in (39) follows the same derivation as in the proof of Theorem 4.1. Putting together, when f is convex, we have

$$\begin{aligned}
&\mathbb{E}\|w_{t+1} - w_{t+1}'\| \\
&\leq (1 + \Gamma_{\mathcal{X}} q_c(\pi)\tau_t)\, \mathbb{E}\|w_t - w_t'\| + \frac{2\tau_t L_{\mathcal{W}}}{n} \qquad (56)\\
&\leq (1 + \Gamma_{\mathcal{X}} q_c(\pi)/\beta)\, \mathbb{E}\|w_t - w_t'\| + \frac{2L_{\mathcal{W}}}{n\beta} \qquad (57)
\end{aligned}$$

when f is $\mu-$ strongly convex, we have

$$\begin{aligned}
&\mathbb{E}\|w_{t+1} - w_{t+1}'\| \\
&\leq \left(1 - \frac{1}{2}\tau_t\mu + \Gamma_{\mathcal{X}} q_c(\pi)\tau_t\right) \mathbb{E}\|w_t - w_t'\| + \frac{2\tau_t L_{\mathcal{W}}}{n} \qquad (58)\\
&\leq \left(1 - \frac{\mu}{2\beta} + \Gamma_{\mathcal{X}} q_c(\pi)/\beta\right) \mathbb{E}\|w_t - w_t'\| + \frac{2L_{\mathcal{W}}}{n\beta} \qquad (59)
\end{aligned}$$

Unravelling the recursion, we have

$$\mathbb{E}\|w_T - w_T'\| \leq \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} \zeta^t \qquad (60)$$

with $\zeta = 1 + \Gamma_{\mathcal{X}} q_c(\pi)/\beta$ when f is convex and $\zeta = 1 - \frac{\mu}{2\beta} + \Gamma_{\mathcal{X}} q_c(\pi)/\beta$ when f is $\mu-$ strongly convex. For the strongly convex case, if we let $q_c(\pi) < \frac{\mu}{2\Gamma_{\mathcal{X}}}$, we have $\zeta < 1$. In this case, the geometric series $\sum_{t=0}^{T} \zeta^t$ converges as $T \to \infty$ and entails a closed form. The bound in (60) can therefore be further simplified as

$$\begin{aligned}
\mathbb{E}\|w_T - w_T'\| &\leq \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{T} \zeta^t \\
&\leq \frac{2L_{\mathcal{W}}}{n\beta} \sum_{t=0}^{\infty} \zeta^t \qquad (61)\\
&= \frac{2L_{\mathcal{W}}}{n\beta} \frac{1}{1-\zeta} \qquad (62)\\
&= \frac{4L_{\mathcal{W}}}{n(\mu - 2q_c(\pi)\Gamma_{\mathcal{X}})} \qquad (63)
\end{aligned}$$

This derives the bound in Corollary 4.4. □

Proof of Lemma 5.1 To establish the proof, we first discuss the expansive property of the one step PGD perturbation T. For arbitrary $\hat{x} \in \mathcal{X}$, we have

$$\begin{aligned}
&\|T_{x,y}(\hat{x}; w) - T_{x,y}(\hat{x}; w')\| \qquad (64)\\
=&\left\|\Pi_{\mathbb{B}_\infty(x,\epsilon)}\left[\hat{x} + \lambda G\left(\nabla_{\hat{x}} f(w, \hat{x}, y)\right)\right] - \Pi_{\mathbb{B}_\infty(x,\epsilon)}\left[\hat{x} + \lambda G\left(\nabla_{\hat{x}} f(w', \hat{x}, y)\right)\right]\right\| \qquad (65)\\
\leq&\lambda \left\|G\left(\nabla_{\hat{x}} f(w, \hat{x}, y)\right) - G\left(\nabla_{\hat{x}} f(w', \hat{x}, y)\right)\right\| \qquad (66)\\
\leq&\lambda\alpha \left\|\nabla_{\hat{x}} f(w, \hat{x}, y) - \nabla_{\hat{x}} f(w', \hat{x}, y)\right\| \qquad (67)\\
\leq&\lambda\alpha\Gamma_{\mathcal{W}} \|w - w'\| \qquad (68)
\end{aligned}$$

The inequality (66) is due to that the projection operation $\Pi_{\mathbb{B}_\infty(x,\epsilon)}$ is 1-expansive. The inequalities (67) and (68) are derived based on the Lipschitz condition of G and $\nabla_x f$.

For fixed $w \in \mathcal{W}$, we have for arbitrary $x', x'' \in \mathcal{X}$

$$\begin{aligned}
&\|T_{x,y}(x';w) - T_{x,y}(x'';w)\| && (69)\\
=&\left\|\Pi_{\mathbb{B}_\infty(x,\epsilon)}\left[x' + \lambda G\left(\nabla_{x'} f(w,x',y)\right)\right] - \Pi_{\mathbb{B}_\infty(x,\epsilon)}\left[x'' + \lambda G\left(\nabla_{x''} f(w,x'',y)\right)\right]\right\| && (70)\\
\leq&\|x' + \lambda G\left(\nabla_{x'} f(w,x',y)\right) - x'' + \lambda G\left(\nabla_{x''} f(w,x'',y)\right)\| && (71)\\
\leq&\|x' - x''\| + \lambda\alpha\left\|\nabla_{x'} f(w,x',y) - \nabla_{x''} f(w,x'',y)\right\| && (72)\\
\leq&(1+\lambda\alpha\eta)\|x' - x''\| && (73)
\end{aligned}$$

The derivation here follows the similar idea as above, utilizing the 1-expansiveness condition of $\Pi_{\mathbb{B}_\infty(x,\epsilon)}$ as well as the Lipschitz condition of G and the smoothness condition of f w.r.t $\mathcal{X}$.

We now derive the upper bound for the expansiveness of π^{PGD}. With a little abuse of notation, let $x_K = T^K_{x,y}(x;w)$ and similarly $x'_K = T^K_{x,y}(x;w')$. For shorter notation, let $\nu = \lambda\alpha\Gamma_{\mathcal{W}}$ and $\mu = 1 + \lambda\alpha\eta$

$$\begin{aligned}
&\|\pi^{\mathrm{PGD}}(x;y,w) - \pi^{\mathrm{PGD}}(x;y,w')\| && (74)\\
=&\|T^K_{x,y}(x;w) - T^K_{x,y}(x;w')\| && (75)\\
=&\|T_{x,y}(x_{K-1};w) - T_{x,y}(x'_{K-1};w')\| && (76)\\
\leq&\|T_{x,y}(x_{K-1};w) - T_{x,y}(x_{K-1};w')\| + \|T_{x,y}(x_{K-1};w') - T_{x,y}(x'_{K-1};w')\| && (77)\\
\leq&\pi\|w - w'\| + \mu\|x_{K-1} - x'_{K-1}\| && (78)\\
=&\pi\|w - w'\| + \mu\|T_{x,y}(x_{K-2};w) - T_{x,y}(x'_{K-2};w')\| && (79)\\
\leq&\sum_{i=0}^{K-1}\mu^i\nu\|w - w'\| && (80)
\end{aligned}$$

Note that the bound (80) holds for any choice of w, w'. On the other hand, using the condition that $T_{x,y}(\hat{x};w) \in \mathbb{B}_\infty(x,\epsilon)$, we can derive that for any $w, w' \in \mathcal{W}$ with $\|w - w'\| > c$,

$$\|T_{x,y}(\hat{x};w) - T_{x,y}(\hat{x};w')\| \leq 2\sqrt{d}\epsilon = \frac{2\sqrt{d}\epsilon}{\|w - w'\|}\|w - w'\| \leq \frac{2\sqrt{d}\epsilon}{c}\|w - w'\| \tag{81}$$

Putting together, we have

$$q_c(\pi^{\mathrm{PGD}}) \leq \min\left(\sum_{i=0}^{K-1}\mu^i\nu, \frac{2\sqrt{d}\epsilon}{c}\right) \tag{82}$$

This completes the proof. □

Proof of Corollary 5.2 We first establish the following result.

For any non-negative random variable Z bounded below B and any $c^* > 0$,

$$\Pr[Z \leq c^*] \leq \frac{B - \mathbb{E}(Z)}{B - c^*} \tag{83}$$

This result simply follows from $\Pr[Z \leq c^*] = \Pr[B - Z \geq B - c^*]$ and applying the Markov Inequality to random variable $B - Z$.

Now let $Z = A(S) - A(S')$ and $c^* = Bn^{-1/2}$ in Theorem 4.2. The second term in bound of Theorem 4.2 then reduces to $\left(1 - \frac{\sup_{S\simeq S'}\mathbb{E}\|A(S)-A(S')\|}{B(1-n^{-1/2})}\right)L_{\mathcal{X}} \cdot 2\epsilon\sqrt{d}$, which converges to $\left(1 - \frac{\sup_{S\simeq S'}\mathbb{E}\|A(S)-A(S')\|}{B}\right)L_{\mathcal{X}} \cdot 2\epsilon\sqrt{d}$ with n. It can be verified that the first term in the bound of Theorem 4.2 vanishes with n (as $n^{-1/2}$). The corollary then follows. □.

Proof of Lemma 6.1 The proof is established by noticing that all members in the set $\tilde{H}(r) := \{x \in \mathbb{R}^d : |x[i]| = r, \forall i \in \mathcal{I}\}$ achieves $1/(rd^{\frac{1}{p}})-$Lipschitz and thus the Lipschitz constant over $H(r)$ is greater than it. Specifically, for any $x, \hat{x} \in \tilde{H}(r)$ with $x \neq \hat{x}$, let $\mathcal{I}_- := \{i \in \mathcal{I} : \text{sgn}(x[i]) \neq \text{sgn}(\hat{x}[i])\}$ and $\mathcal{I}_+ := \mathcal{I} - \mathcal{I}_-$. We have

$$\|G(x) - G(\hat{x})\|_2 \tag{84}$$

$$= \left\| \frac{\text{sgn}(x) \odot |x|^{q-1}}{\|x\|_q^{q-1}} - \frac{\text{sgn}(\hat{x}) \odot |\hat{x}|^{q-1}}{\|\hat{x}\|_q^{q-1}} \right\|_2 \tag{85}$$

$$= \left(\sum_{i=1}^{d} \left| \frac{\text{sgn}(x[i])|x[i]|^{q-1}}{\|x\|_q^{q-1}} - \frac{\text{sgn}(\hat{x}[i])|\hat{x}[i]|^{q-1}}{\|\hat{x}\|_q^{q-1}} \right|^2 \right)^{\frac{1}{2}} \tag{86}$$

$$= \left(\sum_{j \in \mathcal{I}_+} \left| \frac{\text{sgn}(x[j])|x[j]|^{q-1}}{\|x\|_q^{q-1}} - \frac{\text{sgn}(\hat{x}[j])|\hat{x}[j]|^{q-1}}{\|\hat{x}\|_q^{q-1}} \right|^2 + \sum_{k \in \mathcal{I}_-} \left| \frac{\text{sgn}(x[k])|x[k]|^{q-1}}{\|x\|_q^{q-1}} - \frac{\text{sgn}(\hat{x}[k])|\hat{x}[k]|^{q-1}}{\|\hat{x}\|_q^{q-1}} \right|^2 \right)^{\frac{1}{2}} \tag{87}$$

$$= \left(\sum_{k \in \mathcal{I}_-} \left| \frac{2r^{q-1}}{r^{q-1} d^{\frac{1}{p}}} \right|^2 \right)^{\frac{1}{2}} \tag{88}$$

$$= \sqrt{|\mathcal{I}_-|} \frac{2}{d^{\frac{1}{p}}} \tag{89}$$

where $|\mathcal{I}_-|$ denotes the cardinality of the set $\mathcal{I}_-$. The equality (88) is derived by noting that the first term in (87) is zero since $|x[j]| = |\hat{x}[j]|$ and $\text{sgn}(|x[j]|) = \text{sgn}(|\hat{x}[j]|)$ for each $j \in \mathcal{I}_+$ and noting that $\|x\|_q = rd^{\frac{1}{q}}$ for any $x \in \tilde{H}(r)$. The power term $\frac{q-1}{q}$ is replaced by $\frac{1}{p}$ since $1/q + 1/p = 1$. We also have

$$\|x - \hat{x}\|_2 \tag{90}$$

$$= \left(\sum_{i=1}^{d} |x[i] - \hat{x}[i]|^2 \right)^{\frac{1}{2}} \tag{91}$$

$$= \left(\sum_{j \in \mathcal{I}_+} |x[j] - \hat{x}[j]|^2 + \sum_{k \in \mathcal{I}_-} |x[k] - \hat{x}[k]|^2 \right)^{\frac{1}{2}} \tag{92}$$

$$= \left(\sum_{k \in \mathcal{I}_-} |2r|^2 \right)^{\frac{1}{2}} \tag{93}$$

$$= 2r\sqrt{|\mathcal{I}_-|} \tag{94}$$

Putting together, we have that for any $x, \hat{x} \in \tilde{H}(r)$ with $x \neq \hat{x}$,

$$\frac{\|G(x) - G(\hat{x})\|_2}{\|x - \hat{x}\|_2} = \frac{1}{rd^{\frac{1}{p}}} \leq \sup_{\substack{x', x'' \in Q(r) \\ x' \neq x''}} \frac{\|G(x') - G(x'')\|_2}{\|x' - x''\|_2} = \alpha_p \tag{95}$$

This completes the proof. □

B HYPER-PARAMETER SETTINGS FOR THE EXPERIMENTS

In our experiments, we follow the settings in Rice et al. (2020): The perturbation radius is set to be $\epsilon = 8/255$ w.r.t the $\infty-$norm for the three datasets. The pre-activation ResNet 18 (PRN-18) model (He et al., 2016) is used for CIFAR-10 and SVHN. The Wide ResNet 34 (WRN-34) model (Zagoruyko & Komodakis, 2016) is used for CIFAR-100. We set $K = 10$ for all the PGD variants with $\lambda = 2/255$ on CIFAR-10 and CIFAR-100, and set $\lambda = 1/255$ for SVHN. The initial learning

rate of AT is set to be 0.1 for CIFAR-10 and CIFAR-100 and set to be 0.01 for SVHN. The learning rate is decayed by 0.1 at the 100th and the 150th epoch of the training. The batch size is set to be 128 and a weight decay of 5×10^{-4} is used for all the experiments. The experiments are conducted on our internal GPU clusters. Training PRN-18 on CIFAR-10 and SVHN for 200 epochs spends around 18 hours with two NVIDIA V100 GPUs, and training WRN-34 on CIFAR-100 requires around three days to complete with the same computing resources.

C COMPUTING λ_p

The volume of $\mathbb{B}_p(0, \lambda_p)$ is computed by

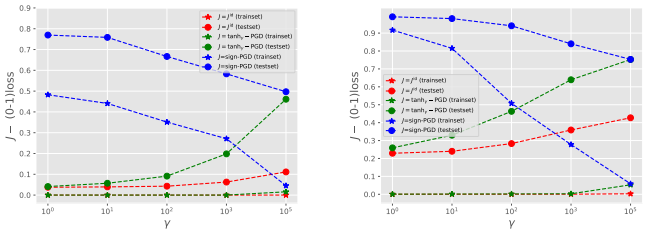
$$\mathrm{vol}\left(\mathbb{B}_p(0, \lambda_p)\right) = \frac{\left(2\Gamma\left(\frac{1}{p}+1\right)\right)^d}{\Gamma\left(\frac{d}{p}+1\right)} \lambda_p^d \tag{96}$$

Here $\Gamma(\cdot)$ denotes the Euler's gamma function. For p other than ∞, to make $\mathrm{vol}\left(\mathbb{B}_p(0, \lambda_p)\right) = \mathrm{vol}\left(\mathbb{B}_\infty(0, \lambda_\infty)\right)$, we have

$$\lambda_p = \exp\left\{\frac{1}{d}\ln\Gamma(\frac{d}{p}+1) + \ln\frac{\lambda_\infty}{\Gamma(\frac{1}{p}+1)}\right\} \tag{97}$$

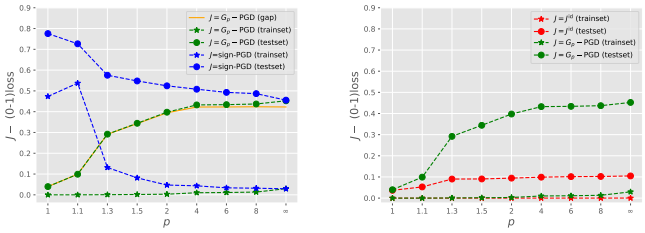
In the experiments, the value of λ_∞ (i.e., the step size for the sign-PGD) is set to be the same as in Section 5 and values for other λ_p is computed from (97).

D OMITTED FIGURES



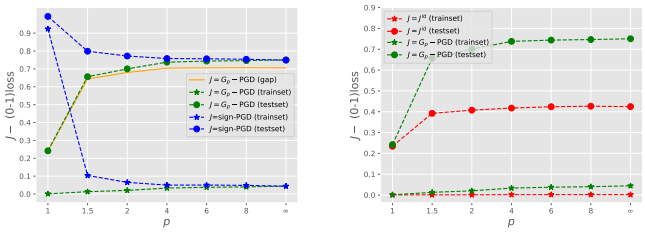
(a) (b)

Figure 4: Experiments in Figure 2 reproduced on SVHN and CIFAR-100.



(a) (b)

Figure 5: Experiments in Figure 3 reproduced on SVHN.



(a) (b)

Figure 6: Experiments in Figure 3 reproduced on CIFAR-100.

E ADDITIONAL RESULTS FOR REBUTTAL

This section has been temporarily added to present the experimental results or additional theoretical analysis requested by the reviewers during the rebuttal period. We will integrate selected results into the final version of the paper at the end of the discussion period.

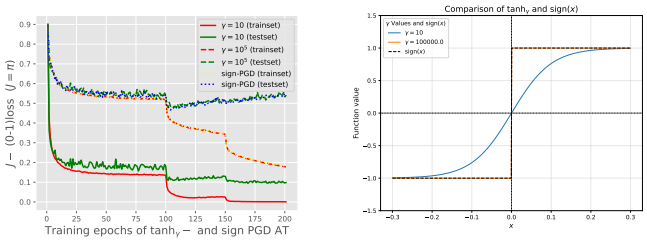
E.1 FOR REVIEWER E9T9

Regarding your question *"Have you tried larger γs? $\gamma = 10^5$ seems to be very far from sign-PGD in Figure 2 (b)"*, we would like to clarify that $\gamma = 10^5$ is indeed sufficiently larger for the $\tanh_\gamma$ function to closely approximate the sign function.

To illustrate this, we have reproduced Figure 2 (b) as Figure 7 (a), including the training trajectory of sign-PGD AT (plotted as yellow and blue dotted curves). It is observed that the dotted curves

almost entirely overlap with the dashed curves, indicating that the behavior of sign-PGD AT is almost identical to that of the $\tanh_\gamma$-PGD AT with $\gamma = 10^5$.

Furthermore, in Figure 7 (b), we compare the sign function with the $\tanh_\gamma$ function for $\gamma = 10$ and 10^5. The plot clearly demonstrates that when $\gamma = 10^5$, the $\tanh_\gamma$ function closely approximate the sign function.



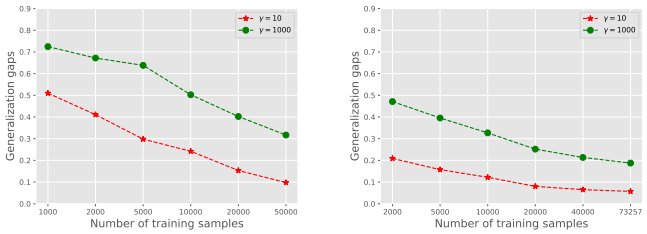
(a) (b)

Figure 7: (a) Reproduction of Figure 2 where we include the trajectory of sign-PGD AT as comparison. (b) Comparison of the $\tanh_\gamma$ function and the sign function.

Regarding your next question *"Can you construct an experiment where the dependence on n is displayed?"*, we have conducted AT experiments using various fraction of the training set from CIFAR-10 and SVHN. The experimental results are presented in Figure 8 (a) and (b).

For each dataset, we perform $\tanh_\gamma-$PGD AT with $\gamma = 10$ and 1000. The generalization gaps for each AT setups are measured by the J-(0-1) loss, where J is taken as the same $\tanh_\gamma-$PGD attack used during the respective training processes. The results clearly show that the generalization gap decreases as the size of the training data increases.

Additionally, we oberve that $\tanh_\gamma-$PGD AT with smaller γ consistently achieves smaller generalization gaps, as evidenced by the red curve being consistently lower than the green curve in each figure. These results align well with our theoretical analysis.



(a) CIFAR-10 (b) SVHN

Figure 8: Generalization gaps of $\tanh_\gamma-$PGD AT using different number of training data for CIFAR-10 and SVHN.

E.2 For Reviewer FPBQ

Reply to concern (1) Regarding your concern *"While the paper considers the algorithmic stability of PGD attack, a missing component is the convergence of PGD attack..."*, we here present a convergence analysis for the PGD attacks defined in (21).

Specifically, we consider PGD attacks with the mapping G that satisfies the following condition:

$$\nabla_x f(w, x, y)^T G(\nabla_x f(w, x, y)) > 0 \tag{98}$$

for any $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and any $w \in \mathcal{W}$. Note that this condition simply requires the direction of the modified gradient $G(\nabla_x f(w, x, y))$ to align near the direction of the original gradient, within 90 degree angle.

Lemma E.1 (Convergence of PGD)**.** *Suppose that $f(w,x,y)$ satisfies the condition (22). Let $x^* = J^*(x;y,w)$ and suppose that $\nabla_x f(w,x^*,y) = 0$. Suppose $\|G(\nabla_x f(w,x,y))\|^2 \leq C$ for any (w,x,y). For any mapping G that satisfies the condition (98), performing the $K-$step PGD (21) with step size $\lambda = \frac{1}{\sqrt{K}}$ results in*

$$f(w,x^*,y) - \frac{1}{K}\sum_{k=1}^{K} f(w,x^k,y) \leq \frac{(2C+d^*)}{2K} + \frac{d^*(\eta^2+\eta+1)}{2} \tag{99}$$

where $d^* = \max_{k\in\{1,\cdots,K\}} \|x^k - x^*\|^2$ and $x^k := T^k_{x,y}(x;w)$ denotes the perturbed instance generated by the $k-$step PGD with $k \leq K$.

We defer the proof of this lemma to the subsection E.2.1. The lemma upper bounds the difference between the maximal loss $f(w,x^*,y)$ and the average of the losses achieved by K-step PGD (averaged over the K steps). If the achieved loss $f(w,x^k,y)$ increases over the K steps, the bound becomes

$$f(w,x^*,y) - f(w,x^K,y) \leq \frac{(2C+d^*)}{2K} + \frac{d^*(\eta^2+\eta+1)}{2}$$

Notably this upper bound decays with K, but converges to a positive constant. This should come as no surprise since without stronger conditions or knowledge on f (e.g., concavity), it is hopeless to have PGD attacks to reach the true maximal loss value $f(w,x^*,y)$.

If we further assume loss functions $f(w,x,y)$ to be concave in x and consider the "raw-gradient (RG)"-PGD where the mapping G is taken as the identity map, we have the following convergence upper bound for PGD by directly adapting the Theorem 3.7 in Bubeck et al. (2015):

Lemma E.2 (Convergence of RG-PGD with concave functions)**.** *Suppose that $f(w,x,y)$ satisfies the condition (22) and is concave in x. Let the mapping G in (20) be the identity map. Then the $K-$step PGD (21) with step size $\lambda = \frac{1}{\eta}$ satisfies*

$$f(w,x^*,y) - f(w,x^K,y) \leq \frac{3\eta\|x-x^*\|^2 + f(w,x^*,y) - f(w,x,y)}{K} \tag{100}$$

where $x^* = J^*(x;y,w)$ and $x^K := T^K_{x,y}(x;w)$.

The bound obviously vanishes with K.

Trade-off between robustness and generalization We here rewrite the notation in (21) as

$$\pi_K^{\mathrm{PGD}}(x;y,w) := T^K_{x,y}(x;w) \tag{101}$$

to emphasize its dependency on K in the PGD attack.

We define the *(expected) robustness gap (on training set)* as

$$\mathrm{RG}(J^*,\pi) := \mathbb{E}_{S,A}\left[R_S[A_\pi(S),J^*] - R_S[A_\pi(S),\pi]\right] \tag{102}$$

This term characterizes the robustness of a model on the training set against J^* when it is trained by AT using some other adversarial perturbation π.

For shorter notation, let $w = A_\pi(S)$ and consider $\mathrm{RG}(J^*,\pi_K^{\mathrm{PGD}})$. We have

$$\begin{aligned}
&\mathrm{RG}(J^*,\pi_K^{\mathrm{PGD}}) \\
=&\mathbb{E}_{S,A}\left[\frac{1}{n}\sum_{i=1}^{n} f(w,J^*(x_i;y_i,w),y_i) - f(w,\pi_K^{\mathrm{PGD}}(x_i;y_i,w),y_i)\right] && (103)\\
\leq& \sup_{(x,y,w)}\left[f(w,J^*(x;y,w),y) - f(w,\pi_K^{\mathrm{PGD}}(x;y,w),y)\right] && (104)\\
=& \sup_{(x,y,w)}\left[f(w,x^*,y) - f(w,x^K,y)\right] && (105)
\end{aligned}$$

where $x^* = J^*(x; y, w)$ and $x^K := \pi_K^{\mathrm{PGD}}(x; y, w)$. Since the results in Lemma E.1 and E.2 apply for arbitrary choice of (w, x, y), they suggest that a smaller robustness gap $\mathrm{RG}(J^*, \pi)$ can be achieved for π_K^{PGD} with larger K.

However, Lemma 5.1 on the other hand suggests that π_K^{PGD} with smaller K tends to achieve a smaller expansiveness parameter $q_c(\pi_K^{\mathrm{PGD}})$ and therefore the corresponding generalization gap $\mathrm{GG}_n(J^*, A_\pi)$ with $\pi = \pi_K^{\mathrm{PGD}}$ tends to be smaller for smaller K.

In summary, the theoretical analysis in this section characterizes the potential trade-off between generalization and the "effectiveness of PGD attack" (measured by $\mathrm{RG}(J^*, \pi_K^{\mathrm{PGD}})$) as was brought up in your comments – We thank you for this pointer, which has helped improve this paper.

Reply to concern (3) Regarding your concern that *" Please consider using some SOTA methods from RobustBench, e.g., leveraging synthetic data in adv training, to conduct the experiments."*, we have conducted additional experiments on the CIFAR-10 dataset following the AT framework in Wang et al. (2023) where the model is trained to minimize the TRADES loss proposed in Zhang et al. (2019) and an additional 1M synthetic dataset is used in the training. We will simply call the AT method used in Wang et al. (2023) and Zhang et al. (2019) as TRADES.

For completeness, we here briefly introduce the TRADES method.

TRADES At each iteration of TRADES, it first draws a training sample $(x_{i_t}, y_{i_t}) \in S$ and then updates the model parameter w_t according to

$$x_t^{\mathrm{adv}} = \pi(x_{i_t}; y_{i_t}, w_t) \tag{106}$$

$$w_{t+1} = w_t - \tau_t \nabla_{w_t} F(w_t, x_{i_t}, x_t^{\mathrm{adv}}, y_{i_t}) \tag{107}$$

where the loss function F is the TRADES loss, consisting of the standard loss and a KL-divergence regularization term:

$$F(w, x, x', y) := f(w, x, y) + \beta \mathrm{KL}(h_w(x) || h_w(x')) \tag{108}$$

where $h : \mathcal{X} \to \mathbb{R}^{|\mathcal{Y}|}$ is a classifier with parameter w. Here we consider $f(w, x, y)$ in the form of $f(w, x, y) = \ell(h_w(x), y)$ where ℓ is the cross-entropy loss. The regularization coefficient $\beta > 0$ is a hyper-parameter.

We consider the following variant of the one-step PGD mapping, defined as

$$\tilde{T}_{x,w}(x') = \Pi_{\mathbb{B}_\infty(x,\epsilon)} \left[x' + \lambda G \left(\nabla_{x'} \mathrm{KL}(h_w(x) || h_w(x')) \right) \right] \tag{109}$$

where the only difference with the PGD attack in (20) is that the perturbation is generated according to the KL divergence term in (108) rather than the standard loss f. Note that using the KL divergence loss makes the perturbation mapping unrelated to the label y.

Its $K-$ step variant $\tilde{\pi}^{\mathrm{PGD}}$ is then

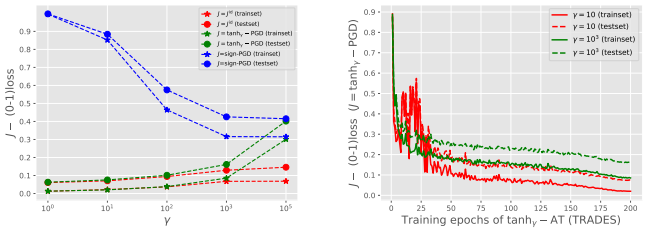
$$\tilde{\pi}^{\mathrm{PGD}}(x; w) := \tilde{T}_{x,w}^K(x) \tag{110}$$

In TRADES, the perturbation in (106) is taken as $\tilde{\pi}^{\mathrm{PGD}}$ rather than π^{PGD}, and the mapping G in (109) is taken as the sign function as in Wang et al. (2023).

We conduct experiments to observe if replacing the sign function with the $\tanh_\gamma$ function would affect the generalization performance of TRADES. We follow the same setup and hyper-parameter settings in Wang et al. (2023) and perform TRADES with $G = \tanh_\gamma$ for $\gamma = \{1, 10, 100, 10^3, 10^5\}$. Specially, we call this type of TRADES as the $\tanh_\gamma-$TRADES.

Models in each experiments are trained for 200 epochs. The trained models are then evaluated by the J-(0-1) loss with J taken from $\{\tanh_\gamma\text{-PGD}, \text{sign-PGD}, J^{\mathrm{id}}\}$. We would like to note that when evaluating the model, we use the PGD attacks defined in (20) rather than (109) for the J-(0-1) loss.

Experimental results are presented in Figure 9 (a) where a phenomenon similar to that in Figure 2 (a) is observed. When model is trained by $\tanh_\gamma$-TRADES with smaller γ, reduced generalization gaps are observed (indicated by the reduced gaps between the dot-shaped and star-shaped curves within each color category).



(a) (b)

Figure 9: (a) Models trained with $\tanh_\gamma$-TRADES for different γ and evaluated by the J-(0-1) loss on the training and testing set. (b) J-(0-1) loss with $J = \tanh_\gamma$-PGD measured along the training trajectories of two sets of $\tanh_\gamma$-TRADES.

Comparing Figure 2 (a) with 9 (a), one may notice that for larger γ, the generalization gaps of $\tanh_\gamma$-TRADES appears to be smaller than those of $\tanh_\gamma$-PGD-AT. This difference is likely due to the additional 1M synthetic data used in $\tanh_\gamma$-TRADES while our PGD-AT experiments only utilize the original training dataset which contains far less number of training examples.

We have also measured the J-(0-1) loss with J taken as the $\tanh_\gamma$-PGD along the training trajectories of $\tanh_\gamma$-TRADES on both the training and the testing sets. The results, shown in Figure 9 (b), use different colors to distinguish $\tanh_\gamma$-TRADES with different γ values. Solid and dashed curves respectively represent the J-(0-1) loss on the training and the testing set. It shows that the solid curves drops faster than the dashed curves, indicating that J-(0-1) loss decreases more rapidly for the $\tanh_\gamma$-TRADES with smaller γ.

In summary, the experimental results indicate that, similar to PGD-AT, the choice of perturbation operators in TRADES also affects its training and generalization performance. On the other hand, we also note that the current analysis in this paper does not fully address the impact of the sign function in other adversarial training frameworks, particularly those involving delicate regularization terms, such TRADES. The key difference between TRADES and our set up is in the form of perturbation: our set up restricts the perturbation to a transformation of the gradient of the standard loss, whereas in TRADES alike approaches, the perturbation is a transformation of the gradient of other quantities. Nevertheless, we expect that the general methodology presented in this paper can be adapted to broader families of adversarial training frameworks. – We sincerely thank the reviewer for bringing up this question, and we will make an effort in that direction.

E.2.1 Proof of Lemma E.1

Since the following proof does not depend on the choice of w and y, for simplicity we will write $f(w, x, y)$ as $f(x)$ and $\nabla_x f(w, x, y)$ as $\nabla f(x)$ hereafter.

To establish the proof of Lemma E.1, we first present and prove the following intermediate result.

Lemma E.3. *Suppose that the gradients of $f(x)$ satisfies the Lipschitz condition (22) and the mapping G in the PGD attack satisfies the condition (98). We have*

$$f(x') - f(x) - \lambda G(\nabla f(x))^T (x' - x) \leq \frac{\eta + 1}{2}\|x' - x\|^2 + \frac{1}{2}\|\nabla f(x)\|^2 + \frac{\lambda^2}{2}\|G(\nabla f(x))\|^2 \quad (111)$$

for any x' and x.

Proof. By the fundamental theorem of calculus, we have

$$f(x') - f(x) = \int_0^1 \frac{d}{dt} f(x + t(x' - x))dt \quad (112)$$

$$= \int_0^1 \nabla f(x + t(x' - x))^T (x' - x)dt \quad (113)$$

We therefore have

$$
\begin{aligned}
& f(x') - f(x) - \lambda G(\nabla f(x))^T (x' - x) \\
= & \int_0^1 \left[\nabla f(x + t(x' - x)) - \lambda G(\nabla f(x))\right]^T (x' - x) dt \qquad (114) \\
= & \int_0^1 \left[\nabla f(x + t(x' - x)) - \nabla f(x)\right]^T (x' - x) dt + \left[\nabla f(x) - \lambda G(\nabla f(x))\right]^T (x' - x) \qquad (115)
\end{aligned}
$$

For the first term in (115), we have

$$
\begin{aligned}
& \int_0^1 \left[\nabla f(x + t(x' - x)) - \nabla f(x)\right]^T (x' - x) dt \\
\leq & \int_0^1 \|\nabla f(x + t(x' - x)) - \nabla f(x)\| \|x' - x\| dt \qquad (116) \\
\leq & \int_0^1 \eta \|t(x' - x)\| \|x' - x\| dt \qquad (117) \\
= & \frac{\eta}{2} \|x' - x\|^2 \qquad (118)
\end{aligned}
$$

where inequality (116) follows from the Cauchy−Schwarz inequality and inequality (117) is due to that the gradient of f is $\eta-$Lipschitz (i.e., condition (22)).

For the second term in (115), we have

$$
\begin{aligned}
& \left[\nabla f(x) - \lambda G(\nabla f(x))\right]^T (x' - x) \\
\leq & \frac{1}{2} \|\nabla f(x) - \lambda G(\nabla f(x))\|^2 + \frac{1}{2} \|x' - x\|^2 \qquad (119) \\
= & \frac{1}{2} \|\nabla f(x)\|^2 + \frac{1}{2} \|\lambda G(\nabla f(x))\|^2 - 2\lambda G(\nabla f(x))^T \nabla f(x) + \frac{1}{2} \|x' - x\|^2 \qquad (120) \\
\leq & \frac{1}{2} \|\nabla f(x)\|^2 + \frac{1}{2} \|\lambda G(\nabla f(x))\|^2 + \frac{1}{2} \|x' - x\|^2 \qquad (121)
\end{aligned}
$$

Inequality (119) is due to that for any vector a and b we have $a^T b \leq \frac{1}{2}\|a\|^2 + \frac{1}{2}\|b\|^2$. Inequality (121) is derived based on the condition (98) that $G(\nabla f(x))^T \nabla f(x) > 0$.

The proof is completed by combining (118) and (121) together. □

We now present the proof for Lemma E.1.

Proof. For simplicity, we write $f(w, x^*, y)$ as $f(x^*)$. Let $x^k := T^k_{x,y}(x; w)$. Additionally, let $\tilde{x}^{k+1} = x^k + \lambda G(\nabla f(x^k))$ and we therefore have $x^{k+1} = \Pi_{\mathbb{B}_\infty(x,\epsilon)}(\tilde{x}^{k+1})$.

According to Lemma E.3, we have that for any $k \leq K$

$$
\begin{aligned}
& f(x^*) - f(x^k) \\
\leq & \lambda G(\nabla f(x^k))^T (x^* - x^k) + \frac{\eta + 1}{2} \|x^* - x^k\|^2 + \frac{1}{2} \|\nabla f(x^k)\|^2 + \frac{\lambda^2}{2} \|G(\nabla f(x^k))\|^2 \qquad (122)
\end{aligned}
$$

For the first term in (122), we have that

$$
\begin{aligned}
& \lambda G(\nabla f(x^k))^T (x^* - x^k) \\
= & (\tilde{x}^{k+1} - x^k)^T (x^* - x^k) \qquad (123) \\
= & \frac{1}{2} \left(\|\tilde{x}^{k+1} - x^k\|^2 + \|x^* - x^k\|^2 - \|\tilde{x}^{k+1} - x^*\|^2\right) \qquad (124) \\
\leq & \frac{1}{2} \left(\|\tilde{x}^{k+1} - x^k\|^2 + \|x^* - x^k\|^2 - \|x^{k+1} - x^*\|^2\right) \qquad (125) \\
= & \frac{1}{2} \left(\|\lambda G(\nabla f(x^k))\|^2 + \|x^* - x^k\|^2 - \|x^{k+1} - x^*\|^2\right) \qquad (126)
\end{aligned}
$$

where equality (124) is due to that for any vector a and b, we have $2a^T b = \|a\|^2 + \|b\|^2 - \|a - b\|^2$. Inequality (125) follows from the fact that since $x^* \in \mathbb{B}_\infty(x, \epsilon)$, we have $\|\tilde{x}^{k+1} - x^*\|^2 \leq \|\Pi_{\mathbb{B}_\infty(x,\epsilon)}(\tilde{x}^{k+1}) - x^*\|^2 = \|x^{k+1} - x^*\|^2$

For the other terms in (122), we have that

$$
\begin{aligned}
&\frac{\eta+1}{2}\|x^* - x^k\|^2 + \frac{1}{2}\|\nabla f(x^k)\|^2 + \frac{\lambda^2}{2}\|G(\nabla f(x^k))\|^2 \\
=&\frac{\eta+1}{2}\|x^* - x^k\|^2 + \frac{1}{2}\|\nabla f(x^k) - \nabla f(x^*)\|^2 + \frac{\lambda^2}{2}\|G(\nabla f(x^k))\|^2 && (127)\\
\leq&\frac{\eta+1}{2}\|x^* - x^k\|^2 + \frac{\eta^2}{2}\|x^k - x^*\|^2 + \frac{\lambda^2 C}{2} && (128)\\
\leq&\frac{d^*(\eta^2+\eta+1)}{2} + \frac{\lambda^2 C}{2} && (129)
\end{aligned}
$$

where equality (127) is derived based on the conditions that $\nabla f(x^*) = 0$. Inequality (128) is derived according to the Lipschitz condition of the gradients (22) and the condition that $\|G(\nabla f(x))\|^2 \leq C$ for any x.

Combining the results, we have that

$$
\begin{aligned}
&f(x^*) - \frac{1}{K}\sum_{k=1}^{K} f(x^k) \\
=&\frac{1}{K}\sum_{k=1}^{K} f(x^*) - f(x^k) && (130)\\
\leq&\frac{1}{2K}\sum_{k=1}^{K}\left(\|\lambda G(\nabla f(x^k))\|^2 + \|x^k - x^*\|^2 - \|x^{k+1} - x^*\|^2\right) + \frac{d^*(\eta^2+\eta+1)}{2} + \frac{\lambda^2 C}{2} && (131)\\
\leq&\frac{\lambda^2 C}{2} + \frac{\|x^1 - x^*\|^2 - \|x^{K+1} - x^*\|^2}{2K} + \frac{d^*(\eta^2+\eta+1)}{2} + \frac{\lambda^2 C}{2} && (132)\\
\leq&\lambda^2 C + \frac{\|x^1 - x^*\|^2}{2K} + \frac{d^*(\eta^2+\eta+1)}{2} && (133)\\
\leq&\lambda^2 C + \frac{d^*}{2K} + \frac{d^*(\eta^2+\eta+1)}{2} && (134)
\end{aligned}
$$

Taking $\lambda = \frac{1}{\sqrt{K}}$, we have

$$
f(x^*) - \frac{1}{K}\sum_{k=1}^{K} f(x^k) \leq \frac{(2C + d^*)}{2K} + \frac{d^*(\eta^2+\eta+1)}{2} \tag{135}
$$

This completes the proof. □