Robust Weight Perturbation for Adversarial Training

Chaojian Yu[1*], **Bo Han**[2], **Mingming Gong**[3], **Li Shen**[4], **Shiming Ge**[5], **Du Bo**[6], **Tongliang Liu**[1†]
[1]Trustworthy Machine Learning Lab, School of Computer Science, The University of Sydney, Australia
[2]Department of Computer Science, Hong Kong Baptist University, China
[3]School of Mathematics and Statistics, The University of Melbourne, Australia
[4]JD Explore Academy, China
[5]Institute of Information Engineering, Chinese Academy of Sciences, China
[6]School of Computer Science, Wuhan University, China
{chyu8051,tongliang.liu}@sydney.edu.au, bhanml@comp.hkbu.edu.hk,
mingming.gong@unimelb.edu.au, mathshenli@gmail.com, geshiming@iie.ac.cn, gunspace@163.com

Abstract

Overfitting widely exists in adversarial robust training of deep networks. An effective remedy is adversarial weight perturbation, which injects the worst-case weight perturbation during network training by maximizing the classification loss on adversarial examples. Adversarial weight perturbation helps reduce the robust generalization gap; however, it also undermines the robustness improvement. A criterion that regulates the weight perturbation is therefore crucial for adversarial training. In this paper, we propose such a criterion, namely Loss Stationary Condition (LSC) for constrained perturbation. With LSC, we find that it is essential to conduct weight perturbation on adversarial data with small classification loss to eliminate robust overfitting. Weight perturbation on adversarial data with large classification loss is not necessary and may even lead to poor robustness. Based on these observations, we propose a robust perturbation strategy to constrain the extent of weight perturbation. The perturbation strategy prevents deep networks from overfitting while avoiding the side effect of excessive weight perturbation, significantly improving the robustness of adversarial training. Extensive experiments demonstrate the superiority of the proposed method over the state-of-the-art adversarial training methods.

1 Introduction

Although deep neural networks (DNNs) have led to impressive breakthroughs in a number of fields such as computer vision [He *et al.*, 2016], speech recognition [Wang *et al.*, 2017], and NLP [Devlin *et al.*, 2018], they are extremely vulnerable to adversarial examples that are crafted by adding small and human-imperceptible perturbation to normal examples [Szegedy *et al.*, 2013; Goodfellow *et al.*, 2014].

The vulnerability of DNNs has attracted extensive attention and led to a large number of defense techniques against adversarial examples. Across existing defenses, adversarial training (AT) is one of the strongest empirical defenses. AT directly incorporates adversarial examples into the training process to solve a min-max optimization problem [Madry *et al.*, 2017], which can obtain models with moderate adversarial robustness and has not been comprehensively attacked [Athalye *et al.*, 2018]. However, different from the natural training scenario, overfitting is a dominant phenomenon in adversarial robust training of deep networks [Rice *et al.*, 2020]. After a certain point in AT, the robust performance on test data will continue to degrade with further training, as shown in Figure 1(a). This phenomenon, termed as *robust overfitting*, breaches the common practice in deep learning that using over-parameterized networks and training for as long as possible [Belkin *et al.*, 2019]. Such anomaly in AT causes detrimental effects on the robust generalization performance and subsequent algorithm assessment [Rice *et al.*, 2020; Chen *et al.*, 2020]. Relief techniques that mitigate robust overfitting have thus become crucial for adversarial training.

An effective remedy for robust overfitting is Adversarial Weight Perturbation (AWP) [Wu *et al.*, 2020], which forms a double-perturbation mechanism that adversarially perturbs both inputs and weights:

$$\min_{w} \max_{v \in \mathcal{V}} \frac{1}{n} \sum_{i=1}^{n} \max_{||x_i' - x_i||_p \leq \epsilon} \ell(f_{w+v}(x_i'), y_i), \qquad (1)$$

where n is the number of training examples, x_i' is the adversarial example of x_i, f_w is the DNN with weight w, $\ell(\cdot)$ is the loss function, ϵ is the maximum perturbation constraint for inputs (*i.e.*, $||x_i' - x_i||_p \leq \epsilon$), and $\mathcal{V}$ is the feasible perturbation region for weights (*i.e.*, $\{v \in \mathcal{V} : ||v||_2 \leq \gamma||w||_2\}$, where γ is the constraint on weight perturbation size). The inner maximization is to find adversarial examples x_i' within the ϵ-ball centered at normal examples x_i that maximizes the classification loss ℓ. On the other hand, the outer maximization is to find weight perturbation v that maximizes the loss ℓ on adversarial examples to reduce robust generalization gap. This is the problem of training a weight-perturbed robust classifier on adversarial examples. Therefore, how well the weight

*This work is done during an internship at JD Explore Academy
†Corresponding author

perturbation is found directly affects the performance of the outer minimization, *i.e.*, the robustness of the classifier.

Several attack methods have been used to solve the inner maximization problem in Eq.(1), such as Fast Gradient Sign Method (FGSM) [Goodfellow *et al.*, 2014] and Projected Gradient Descent (PGD) [Madry *et al.*, 2017]. For the outer maximization problem, AWP [Wu *et al.*, 2020] injects the worst-case weight perturbation to reduce robust generalization gap. However, the extent to which the weights should be perturbed has not been explored. Without an appropriate criterion to regulate the weight perturbation, the adversarial training procedure is difficult to unleash its full power, since worst-case weight perturbation will undermine the robustness improvement (in Section 3). In this paper, we propose such a criterion, namely Loss Stationary Condition (LSC) for constrained perturbation (in Section 3), which helps to better understand robust overfitting, and this in turn motivates us to propose an improved weight perturbation strategy for better adversarial robustness (in Section 4). Our main contributions as follows:

- We propose a principled criterion LSC to analyse the adversarial weight perturbation. It provides a better understanding of robust overfitting in adversarial training, and it is also a good indicator for efficient weight perturbation.
- With LSC, we find that better perturbation of model weights is associated with perturbing on adversarial data with small classification loss. For adversarial data with large classification loss, weight perturbation is not necessary and can even be harmful.
- We propose a robust perturbation strategy to constrain the extent of weight perturbation. Experiments show that the robust strategy significantly improves the robustness of adversarial training.

2 Related Work

2.1 Adversarial Attacks

Let $\mathcal{X}$ denote the input feature space and $\mathcal{B}_\epsilon^p(x) = \{x' \in \mathcal{X} : ||x' - x||_p \leq \epsilon\}$ be the ℓ_p-norm ball of radius ϵ centered at x in $\mathcal{X}$. Here we selectively introduce several commonly used adversarial attack methods.

Fast Gradient Sign Method (FGSM). FGSM [Goodfellow *et al.*, 2014] perturbs natural example x for one step with step size ϵ along the gradient direction:

$$x' = x + \epsilon \cdot \mathrm{sign}(\nabla_x \ell(f_w(x), y)). \tag{2}$$

Projected Gradient Descent (PGD). PGD [Madry *et al.*, 2017] is a stronger iterative variant of FGSM, which perturbs normal example x for multiple steps K with a smaller step size α:

$$x^0 \sim \mathcal{U}(\mathcal{B}_\epsilon^p(x)), \tag{3}$$

$$x^k = \Pi_{\mathcal{B}_\epsilon^p(x)}(x^{k-1} + \alpha \cdot \mathrm{sign}(\nabla_{x^{k-1}} \ell(f_w(x^{k-1}), y))), \tag{4}$$

where $\mathcal{U}$ denotes the uniform distribution, x^0 denotes the normal example disturbed by a small uniform random noise, x^k denotes the adversarial example at step k, and $\Pi_{\mathcal{B}_\epsilon^p(x)}$ denotes the projection function that projects the adversarial example back into the set $\mathcal{B}_\epsilon^p(x)$ if necessary.

AutoAttack (AA). AA [Croce and Hein, 2020] is an ensemble of complementary attacks, which consists of three white-box attacks and a black-box attack. AA regards models to be robust only if the models correctly classify all types of adversarial examples, which is among the most reliable evaluation of adversarial robustness to date.

2.2 Adversarial Defense

Since the discovery of adversarial examples, a large number of works have emerged for defending against adversarial attacks, such as input denoising [Wu *et al.*, 2021], modeling adversarial noise [Zhou *et al.*, 2021], and adversarial training [Goodfellow *et al.*, 2014; Madry *et al.*, 2017]. Among them, adversarial training has been demonstrated to be one of the most effective method [Athalye *et al.*, 2018]. Based on adversarial training, a wide range of subsequent works are then proposed to further improve the model robustness. Here, we introduce two currently state-of-the-art AT frameworks.

TRADES. TRADES [Zhang *et al.*, 2019] optimizes a regularized surrogate loss that is a trade-off between the natural accuracy and adversarial robustness:

$$\ell^{\mathrm{TRADES}}(w; x, y) = \frac{1}{n}\sum_{i=1}^{n} \big\{ \mathrm{CE}(f_w(x_i), y_i) + \beta \cdot \max_{x' \in \mathcal{B}_\epsilon^p(x)} \mathrm{KL}(f_w(x_i) || f_w(x_i')) \big\}, \tag{5}$$

where CE is the cross-entropy loss that encourages the network to maximize the natural accuracy, KL is the Kullback-Leibler divergence that encourages to improve the robust accuracy, and β is the hyperparameter to control the trade-off between natural accuracy and adversarial robustness.

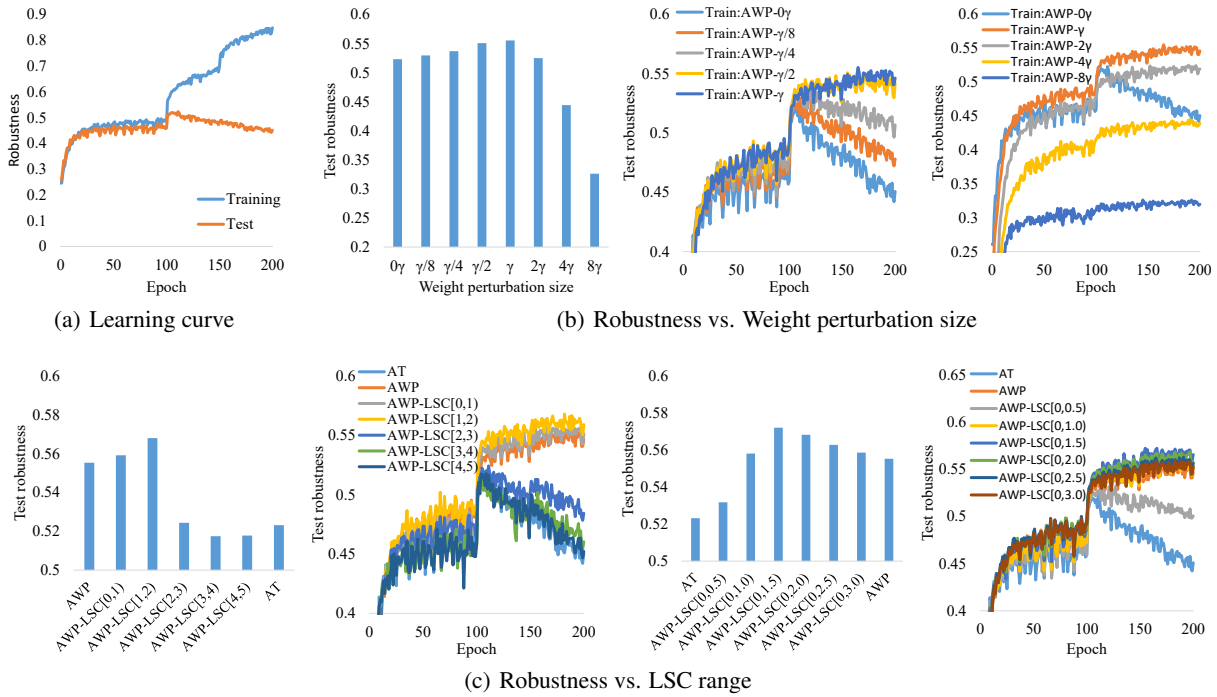
Robust Self-Training (RST). RST [Carmon *et al.*, 2019] utilize additional 500K unlabeled data extracted from the 80 Million Tiny Images dataset. RST first leverages the surrogate natural model to generate pseudo-labels for these unlabeled data, and then adversarially trains the network with both additional pseudo-labeled unlabeled data $(\tilde{x}, \tilde{y})$ and original labeled data (x, y) in a supervised setting:

$$\ell^{\mathrm{RST}}(w; x, y, \tilde{x}, \tilde{y}) = \ell^{\mathrm{TRADES}}(w; x, y) + \lambda \cdot \ell^{\mathrm{TRADES}}(w; \tilde{x}, \tilde{y}), \tag{6}$$

where λ is the weight on unlabeled data.

2.3 Robust Overfitting

Nowadays, there are effective countermeasures to alleviate the overfitting in natural training. But in adversarial training, robust overfitting widely exists and those common countermeasures used in natural training help little [Rice *et al.*, 2020]. [Schmidt *et al.*, 2018] explains robust overfitting partially from the perspective of sample complexity, and is supported by empirical results in derivative works, such as adversarial training with semi-supervised learning [Carmon *et al.*, 2019; Uesato *et al.*, 2019; Zhai *et al.*, 2019], robust local feature [Song *et al.*, 2020] and data interpolation [Zhang and Xu, 2019; Lee *et al.*, 2020; Chen *et al.*, 2021]. Separate works have also attempt to mitigate robust overfitting by the unequal treatment of data [Zhang *et al.*, 2020] and weight smoothing [Chen *et al.*, 2020]. Recent study [Wu *et al.*, 2020] reveals the



(a) Learning curve

(b) Robustness vs. Weight perturbation size

(c) Robustness vs. LSC range

Figure 1: (a): The learning curve of vanilla AT; (b): Test robustness of AWP with varying weight perturbation size; (c): Test robustness of AWP with varying LSC range.

connection between the flatness of weight loss landscape and robust generalization gap, and proposes to incorporate adversarial weight perturbation mechanism in the adversarial training framework. Despite the efficacy of adversarial weight perturbation in suppressing the robust overfitting, a deeper understanding of robust overfitting and a clear direction for valid weight perturbation is largely missing. The outer maximization in Eq.(1) lacks an effective criterion to regulate and constrain the extent of weight perturbation, which in turn influences the optimization of the outer minimization. In this paper, we propose such a criterion and provide new understanding of robust overfitting. Following this, we design a robust weight perturbation strategy that significantly improves the robustness of adversarial training.

3 Loss Stationary Condition

In this section, we first empirically investigate the relationship between weight perturbation robustness and adversarial robustness, and then propose a criterion to analyse the adversarial weight perturbation, which leads to a new perspective of robust overfitting. To this end, some discussions about robust overfitting and weight perturbation are provided.

Does Weight Perturbation Robustness Certainly Lead to Better Adversarial Robustness? First, we investigate whether the robustness against weight perturbation is beneficial to the adversarial robustness. In particular, we train PreAct ResNet-18 with AWP on CIFAR-10 using varying weight perturbation size from 0γ, $\gamma/8$, $\gamma/4$, $\gamma/2$, γ, 2γ, 4γ to 8γ. In each setting, we evaluate the robustness of the model against 20-step PGD (PGD-20) attacks on CIFAR-10 test images. As shown in Figure 1(b), when weight perturbation size is small, the best adversarial robustness has a certain improvement. However, when weight perturbation size is large, the best adversarial robustness begins to decrease significantly as the size of the perturbation increases. It can be explained by the fact that the network has to sacrifice adversarial robustness to allocate more capacity to defend against weight perturbation when weight perturbation size is large, which indicates that weight perturbation robustness and adversarial robustness are not mutually beneficial. As shown in Figure 1(b), the performance gain of AWP is mainly due to suppressing robust overfitting.

Loss Stationary Condition. In order to further analyse the weight perturbation, we propose a criterion that divides the training adversarial examples into different groups according to their classification loss:

$$\mathrm{LSC}[p, q] = \{x' \in \mathcal{X} \mid p \leq \ell(f_w(x'), y) \leq q\}, \qquad (7)$$

where $p \leq q$. The adversarial data in the group all satisfy their adversarial loss within a certain range, which is termed Loss Stationary Condition (LSC). The proposed criterion LSC allows the analysis of grouped adversarial data independently, and provides more insights into the robust overfitting.

LSC view of Adversarial Weight Perturbation. To provide more insight into how AWP suppresses robust overfitting, we train PreAct ResNet-18 on CIFAR-10 by varying the LSC group that performs adversarial weight perturbation. In each setting, we evaluate the robustness of the model against

Algorithm 1 Robust Weight Perturbation (RWP)

Input: Network f_w, training data S, mini-batch $\mathcal{B}$, batch size n, learning rate η, PGD step size α, PGD steps K_1, PGD constraint ϵ, RWP steps K_2, RWP constraint γ, minimum loss value c_{min}.
Output: Adversarially robust model f_w.
repeat
  Read mini-batch $x_{\mathcal{B}}$ from training set S.
  $x'_{\mathcal{B}} \leftarrow x_{\mathcal{B}} + \delta$, where $\delta \sim \text{Uniform}(-\epsilon, \epsilon)$
  for $k = 1$ **to** K_1 **do**
    $x'_{\mathcal{B}} \leftarrow \Pi_\epsilon(x'_{\mathcal{B}} + \alpha \cdot \text{sign}(\nabla_{x'_{\mathcal{B}}} \ell(f_w(x'_{\mathcal{B}}), y)))$
  end for
  Initialize $v = \mathbf{0}$
  for $k = 1$ **to** K_2 **do**
    $V = \mathbb{I}_{\mathcal{B}}(\ell(f_{w+v}(x'_{\mathcal{B}}), y) \leq c_{min})$
    if $\sum V = 0$ **then**
      break
    else
      $v \leftarrow v + \nabla_v(V \cdot \ell(f_{w+v}(x'_{\mathcal{B}}), y))$
      $v \leftarrow \gamma \frac{v}{||v||}||w||$
    end if
  end for
  $w \leftarrow (w+v) - \eta \nabla_{w+v} \frac{1}{n} \sum_{i=1}^{n} \ell(f_{w+v}(x'^{(i)}_{\mathcal{B}}), y^{(i)}) - v$
until training converged

PGD-20 attacks on CIFAR-10 test images. As shown in Figure 1(c), when varying the LSC range, we can observe that conducting adversarial weight perturbation on adversarial examples with small classification loss is sufficient to eliminate robust overfitting. However, conducting adversarial weight perturbation on adversarial examples with large classification loss fails to suppress robust overfitting. The results indicate that to eliminate robust overfitting, it is essential to prevent the model from memorizing these easy-to-learn adversarial examples. Besides, it is observed that conducting adversarial weight perturbation on adversarial examples with large classification loss leads to worse adversarial robustness, which again verifies that the robustness against weight perturbation will not bring adversarial robustness gain, or even on the contrary, it undermines the adversarial robustness enhancement.

Do We Really Need the Worst-case Weight Perturbation? As aforementioned, the robustness against weight perturbation is not beneficial to the adversarial robustness improvement. Therefore, to purely eliminate robust overfitting, conducting worst-case weight perturbation on these adversarial examples is not necessary. In the next section, we will propose a robust perturbation strategy to address this issue.

4 Robust Weight Perturbation

As mentioned in Section 3, conducting adversarial weight perturbation on adversarial examples with small classification loss is enough to prevent robust overfitting and leads to higher robustness. However, conducting adversarial weight perturbation on adversarial examples with large classification loss may not be helpful. Recalling the criterion LSC proposed in Section 3, we have seen that the loss is closely correlated with the tendency of adversarial example to be overfitted. Thus, it can be used to constrain the extent of weight perturbation at a fine-grained level. Therefore, we propose to conduct weight perturbation on adversarial examples that are below a minimum loss value, so as to ensure that no robust overfitting occurs while avoiding the side effect of excessive weight perturbation. Let c_{min} be the minimum loss value. Instead of generating weight perturbation v via outer maximization in Eq.(1), we generate v as follows:

$$v^{k+1} = v^k + \nabla_{v^k} \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(x'_i, y_i) \ell(f_{w+v^k}(x'_i), y_i),$$
$$\text{where} \quad \mathbb{I}(x'_i, y_i) = \begin{cases} 0 & \text{if } \ell(f_{w+v^k}(x'_i), y_i) > c_{min} \\ 1 & \text{if } \ell(f_{w+v^k}(x'_i), y_i) \leq c_{min} \end{cases} \tag{8}$$

The proposed Robust Weight Perturbation (RWP) algorithm is shown in Algorithm 1. We use PGD attack [Madry *et al.*, 2017] to generate the training adversarial examples, which can be also extended to other variants such as TRADES [Zhang *et al.*, 2019] and RST [Carmon *et al.*, 2019]. The mimimum loss value c_{min} controls the extent of weight perturbation during network training. For example, in the early stages of training, the classification loss of adversarial example is generally larger than c_{min} corresponding to no weight perturbation process. The classification loss of adversarial examples then decreases as training progresses. At each optimization step, we monitor the classification loss of the adversarial example and conduct the weight perturbation process for adversarial examples whose classification loss is smaller than c_{min}, enabled by an indicator control vector V. At each perturbation step, the weight perturbation v will be updated to increase the classification loss of the corresponding adversarial example. When the classification loss of training adversarial examples is all higher than c_{min} or the number of perturbation step reaches the defined value, we stop the weight perturbation process and inject the generated weight perturbation v for adversarial training.

5 Experiments

In this section, we conduct comprehensive experiments to evaluate the effectiveness of RWP including its experimental settings, robustness evaluation and ablation studies.

5.1 Experimental Setup

Baselines and Implementation Details. Our implementation is based on PyTorch and the code is publicly available[1]. We conduct extensive experiments across three benchmark datasets (CIFAR-10, CIFAR-100 and SVHN) and two threat models (L_∞ and L_2). We use PreAct ResNet-18 [He *et al.*, 2016] and Wide ResNet (WRN-28-10 and WRN-34-10) [Zagoruyko and Komodakis, 2016] as the network structure following [Wu *et al.*, 2020]. We compare the performance of the proposed method on a number of baseline methods: 1) standard adversarial training without weight perturbation, including vanilla AT [Madry *et al.*, 2017], TRADES [Zhang *et al.*, 2019] and RST [Carmon *et al.*, 2019]; 2) adversarial training with AWP [Wu *et al.*, 2020], including AT-

[1]https://github.com/ChaojianYu/Robust-Weight-Perturbation

Threat Model	Method	SVHN		CIFAR-10		CIFAR-100	
		Best	Last	Best	Last	Best	Last
L_∞	AT	53.22 ± 0.20	45.13 ± 0.17	52.32 ± 0.31	45.08± 0.19	27.79± 0.45	20.95± 0.30
	AT-AWP	59.49± 0.15	55.16± 0.10	55.54± 0.20	54.64± 0.25	30.89± 0.21	30.48± 0.43
	AT-RWP	**61.15± 0.16**	**57.45± 0.23**	**58.55± 0.50**	**58.01± 0.33**	**31.17± 0.18**	**30.64± 0.24**
L_2	AT	66.71± 0.24	65.25± 0.19	69.40± 0.38	66.02± 0.15	40.95± 0.13	36.24± 0.26
	AT-AWP	72.80± 0.30	68.40± 0.20	72.72± 0.21	72.48± 0.45	45.63± 0.48	44.98± 0.30
	AT-RWP	**73.35± 0.20**	**69.48± 0.32**	**74.47± 0.14**	**73.84± 0.27**	**45.71± 0.17**	**45.05± 0.30**

Table 1: Test robustness (%) of AT, AT-AWP and AT-RWP using PreAct ResNet-18.

Defense	Natural	FGSM	PGD-20	PGD-100	C&W$_\infty$	AA
AT	86.52± 0.57	61.91± 0.15	55.47± 0.10	55.15± 0.28	54.51± 0.19	52.18± 0.04
AT-AWP	85.67± 0.40	64.31± 0.23	58.57± 0.22	58.46± 0.17	55.78± 0.32	53.63± 0.09
AT-RWP	**86.86± 0.51**	**66.22± 0.31**	**62.87± 0.25**	**62.87± 0.34**	**56.62± 0.18**	**54.61± 0.11**
TRADES	84.42± 0.36	61.20± 0.09	56.05± 0.13	55.85± 0.20	53.67± 0.14	52.64± 0.07
TRADES-AWP	84.55± 0.30	62.99± 0.30	59.20± 0.24	59.05± 0.31	55.92± 0.20	55.32± 0.05
TRADES-RWP	**86.14± 0.43**	**64.70± 0.17**	**60.45± 0.19**	**60.30± 0.30**	**58.07± 0.33**	**57.20± 0.09**
RST	**89.88± 0.36**	**70.08± 0.62**	62.40± 0.51	62.08± 0.31	61.14± 0.46	59.71± 0.10
RST-AWP	88.01± 0.68	68.00± 0.23	63.67± 0.38	63.50± 0.11	60.55± 0.21	59.80± 0.08
RST-RWP	88.87± 0.55	69.71± 0.12	**64.11± 0.16**	**63.92± 0.26**	**62.03± 0.23**	**60.36± 0.06**

Table 2: Test robustness (%) on CIFAR-10 using Wide ResNet under L_∞ threat model.

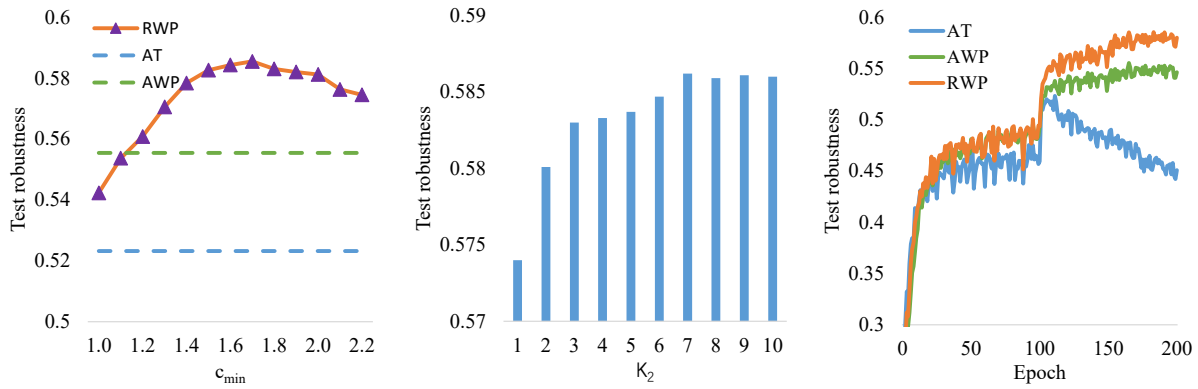
AWP, TRADES-AWP and RST-AWP. For training, the network is trained for 200 epochs using SGD with momentum 0.9, weight decay 5×10^{-4}, and an initial learning rate of 0.1. The learning rate is divided by 10 at the 100-th and 150-th epoch. Standard data augmentation including random crops with 4 pixels of padding and random horizontal flips are applied. For testing, model robustness is evaluated by measuring the accuracy of the model under different adversarial attacks. For hyper-parameters in RWP, we set perturbation step $K_2 = 10$ for all datasets. The minimum loss value $c_{min} = 1.7$ for CIFAR-10 and SVHN, and $c_{min} = 4.0$ for CIFAR-100. The weight perturbation budget of $\gamma = 0.01$ for AT-RWP, $\gamma = 0.005$ for TRADES-RWP and RST-RWP following literature [Wu *et al.*, 2020]. Other hyper-parameters of the baselines are configured as per their original papers.

Adversarial Setting. The training attack is 10-step PGD attack with random start. We follow the same settings in [Rice *et al.*, 2020] : for L_∞ threat model, $\epsilon = 8/255$, step size $\alpha = 1/255$ for SVHN, and $\alpha = 2/255$ for both CIFAR10 and CIFAR100; for L_2 threat model, $\epsilon = 128/255$, step size $\alpha = 15/255$ for all datasets, which is a standard setting for adversarial training [Madry *et al.*, 2017]. The test attacks used for robustness evaluation contains FGSM, PGD-20, PGD-100, C&W$_\infty$ and Auto Attack (AA).

5.2 Robustness Evaluation

Performance Evaluations. To validate the effectiveness of the proposed RWP, we conduct performance evaluation on vanilla AT, AT-AWP and AT-RWP across different benchmark datasets and threat models using PreAct ResNet-18. We report the accuracy on the test images under PGD-20 attack. The evaluation results are summarized in Table 1, where "Best" denotes the highest robustness that ever achieved at different checkpoints and "Last" denotes the robustness at the last epoch checkpoint. It is observed vanilla AT suffers from severe robust overfitting (the performance gap between "best" and "last" is very large). AT-AWP and AT-RWP method narrow the performance gap significantly over the vanilla AT model due to suppression of robust overfitting. Moreover, on CIFAR-10 dataset under the L_∞ attack, vanilla AT achieves 52.32% "best" test robustness. The AT-AWP approach boosts the performance to 55.54%. The proposed approach further outperforms both methods by a large margin, improving over vanilla AT by 6.23%, and is 3.01% better than AT-AWP, achieving 58.55% accuracy under the standard 20 steps PGD attack. Similar patten has been observed on other datasets and threat model. AT-RWP consistently improves the test robustness across a wide range of datasets and threat models, demonstrating the effectiveness of the proposed approach.

Benchmarking the state-of-the-art Robustness. To manifest the full power of our proposed perturbation strategy and also benchmark the state-of-the-art robustness on CIFAR-10 under L_∞ threat model, we conduct experiments on the large capacity network with different baseline methods. We train Wide ResNet-34-10 for AT and TRADES, and Wide ResNet-28-10 for RST following their original papers. We evaluate the adversarial robustness of trained model with various test attack and report the "best" test robustness, with the results shown in Table 2. "Natural" denotes the accuracy on natural test data. First, it is observed that the natural accuracy of RWP model consistently outperforms AWP by a large margin. It is due to the benefits that our RWP avoids the excessive weight perturbation. Moreover, RWP achieves the best adversarial robustness against almost all types of attack across a wide



(a) Importance of minimum loss value c_{min}

(b) Impacts of step number K_2

(c) Effect of RWP on adversarial robustness and robust overfitting

Figure 2: The ablation study experiments on CIFAR-10.

range of baseline methods, which verifies that RWP is effective in general and improves adversarial robustness reliably rather than improper tuning of hyper-parameters of attacks, gradient obfuscation or masking.

5.3 Ablation Studies

In this part, we investigate the impacts of algorithmic components using AT-RWP on PreAct ResNet-18 under L_∞ threat model following the same setting in section 5.1.

The Importance of Minimum Loss Value. We verify the effectiveness of minimum loss value c_{min}, by comparing the performance of models trained using different weight perturbation schemes: 1) AT: standard adversarial training without weight perturbation (equivalent to $c_{min} = 0$); 2) AWP: weight perturbation generated via outer maximization in Eq.(1) (equivalent to $c_{min} = \infty$); 3) RWP: weight perturbation generated using the proposed robust strategy with different c_{min} values. All other hyper-parameters are kept exactly the same other than the perturbation scheme used. The results are summarized in Figure 2(a). It is observed that the test robustness of RWP model first increases and then decreases as the minimum loss value increases, and the best test robustness is obtained at $c_{min} = 1.7$. It is evident that RWP with a wide range of c_{min} outperforms both AT and AWP methods, demonstrating its effectiveness. Furthermore, as it is the major component that is different from the AWP pipeline, this result suggests that the proposed LSC constraints is the main contributor to the improved adversarial robustness.

The Impact of Step Number. We further investigate the effect of step number K_2, by comparing the performances of model trained using different perturbation steps. The step number K_2 for RWP varies from 1 to 10. The results are shown in Figure 2(b). As expected, when K_2 is small, increasing K_2 leads higher test robustness. When K_2 increases from 7 to 10, the performance is flat, which suggests that the generated weight perturbation is sufficient to comprehensively avoid robust overfitting. Note that extra iterations will not bring computational overhead when the classification loss of adversarial examples exceeds minimum loss value c_{min}, as shown in Algorithm 1. Therefore, we uniformly use $K_2 = 10$ in our implementation.

Effect on Adversarial Robustness and Robust Overfitting. We then visualize the learning curves of AT, AWP and RWP, which are summarized in Figure 2(c). It is observed that the test robustness of RWP model continues to increase as the training progresses. In addition, RWP outperforms AWP with a clear margin in the later stage of training. Such observations exactly reflect the nature of our approach which aims to prevent robust overfitting as well as boost the robustness of adversarial training.

6 Conclusion

In this paper, we proposed a criterion, Loss Stationary Condition (LSC) for constrained weight perturbation. The proposed criterion provides a new understanding of robust overfitting. Based on LSC, we found that elimination of robust overfitting and higher robustness of adversarial training can be achieved by weight perturbation on adversarial examples with small classification loss, rather than adversarial examples with large classification loss. Following this, we proposed a Robust Weight Perturbation (RWP) strategy to regulate the extent of weight perturbation. Comprehensive experiments show that RWP is generic and can improve the state-of-the-art adversarial robustness across different adversarial training approaches, network architectures, threat models and benchmark datasets.

Acknowledgements

This work is supported in part by Beijing Natural Science Foundation (19L2040), NSFC Young Scientists Fund No. 62006202, Guangdong Basic and Applied Basic Research Foundation No. 2022A1515011652, and Science and Technology Innovation 2030 –"Brain Science and Brain-like Research" Major Project (No. 2021ZD0201402 and No. 2021ZD0201405).

References

[Athalye *et al.*, 2018] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283. PMLR, 2018.

[Belkin *et al.*, 2019] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[Carmon *et al.*, 2019] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, Percy Liang, and John C Duchi. Unlabeled data improves adversarial robustness. *arXiv preprint arXiv:1905.13736*, 2019.

[Chen *et al.*, 2020] 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.

[Chen *et al.*, 2021] Chen Chen, Jingfeng Zhang, Xilie Xu, Tianlei Hu, Gang Niu, Gang Chen, and Masashi Sugiyama. Guided interpolation for adversarial training. *arXiv preprint arXiv:2102.07327*, 2021.

[Croce and Hein, 2020] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Goodfellow *et al.*, 2014] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Lee *et al.*, 2020] Saehyung Lee, Hyungyu Lee, and Sungroh Yoon. Adversarial vertex mixup: Toward better adversarially robust generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 272–281, 2020.

[Madry *et al.*, 2017] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[Rice *et al.*, 2020] Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pages 8093–8104. PMLR, 2020.

[Schmidt *et al.*, 2018] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. *Advances in Neural Information Processing Systems*, 31:5014–5026, 2018.

[Song *et al.*, 2020] Chubiao Song, Kun He, Jiadong Lin, Liwei Wang, and John E Hopcroft. Robust local features for improving the generalization of adversarial training. In *International Conference on Learning Representations*, 2020.

[Szegedy *et al.*, 2013] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[Uesato *et al.*, 2019] Jonathan Uesato, Jean-Baptiste Alayrac, Po-Sen Huang, Robert Stanforth, Alhussein Fawzi, and Pushmeet Kohli. Are labels required for improving adversarial robustness? *arXiv preprint arXiv:1905.13725*, 2019.

[Wang *et al.*, 2017] Yisen Wang, Xuejiao Deng, Songbai Pu, and Zhiheng Huang. Residual convolutional ctc networks for automatic speech recognition. *arXiv preprint arXiv:1702.07793*, 2017.

[Wu *et al.*, 2020] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33, 2020.

[Wu *et al.*, 2021] Boxi Wu, Heng Pan, Li Shen, Jindong Gu, Shuai Zhao, Zhifeng Li, Deng Cai, Xiaofei He, and Wei Liu. Attacking adversarial attacks as a defense. *arXiv preprint arXiv:2106.04938*, 2021.

[Zagoruyko and Komodakis, 2016] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[Zhai *et al.*, 2019] Runtian Zhai, Tianle Cai, Di He, Chen Dan, Kun He, John Hopcroft, and Liwei Wang. Adversarially robust generalization just requires more unlabeled data. *arXiv preprint arXiv:1906.00555*, 2019.

[Zhang and Xu, 2019] Haichao Zhang and Wei Xu. Adversarial interpolation training: A simple approach for improving model robustness. 2019.

[Zhang *et al.*, 2019] 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*, pages 7472–7482. PMLR, 2019.

[Zhang *et al.*, 2020] Jingfeng Zhang, Jianing Zhu, Gang Niu, Bo Han, Masashi Sugiyama, and Mohan Kankanhalli. Geometry-aware instance-reweighted adversarial training. *arXiv preprint arXiv:2010.01736*, 2020.

[Zhou *et al.*, 2021] Dawei Zhou, Nannan Wang, Bo Han, and Tongliang Liu. Modeling adversarial noise for adversarial defense. *arXiv preprint arXiv:2109.09901*, 2021.