TOWARDS UNDERSTANDING CATASTROPHIC OVERFITTING IN FAST ADVERSARIAL TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

After adversarial training was proposed, a series of works focus on improving the computational efficiency of adversarial training for deep neural networks (DNNs). Recently, FGSM based single-step adversarial training has been found to be able to train a robust model with the robustness comparable to the one trained by multi-step PGD, but it is an order of magnitude faster. However, there exists a failure mode called **Catastrophic Overfitting (CO)** where the network suddenly loses its robustness against multi-step attacks and hardly recovers by itself during the training process. This paper identifies that CO is closely related to the high-order terms in Taylor expansion after rethinking and decomposing the min-max problem in adversarial training. The negative high-order terms lead to a Perturbation Loss Distortion phenomenon, which is the underlying cause of CO. Based on the observations, we propose a simple but effective regularization method named Fast Linear Adversarial Training (FLAT) to avoid CO in the single-step adversarial training by making the loss surface flat.

1 INTRODUCTION

In recent years, deep learning has achieved state-of-the-art performance in many fields, such as computer vision (He et al., 2016) and natural language processing (Devlin et al., 2018). However, those deep neural networks (DNNs) are proved to be highly vulnerable to adversarial examples (Szegedy et al., 2013; Biggio et al., 2013), which are crafted by adding human imperceptible perturbations to clean examples. These properties of DNNs raise security concerns when DNNs deployed into real-world applications. Thus, it is essential to train a robust model with high accuracy both on clean examples and perturbed adversarial examples.

There have emerged various types of defense techniques to improve the robustness of DNNs, such as pre/post-processing (Buckman et al., 2018; Song et al., 2018), regularization based methods (Ross & Doshi-Velez, 2018; Moosavi-Dezfooli et al., 2019; Qin et al., 2019) and adversarial training (Goodfellow et al., 2014; Madry et al., 2017). However, most of these defenses are found to give a false sense of robustness because of gradient obfuscation (Athalye et al., 2018), and could be broken by well-designed stronger adaptive attacks (Tramer et al., 2020). Croce & Hein (2020) evaluated about 50 defensive methods and found most of them either have lower robustness or can be broken with a stronger attack named AutoAttack. In the end, across these defenses, adversarial training using projected gradient descent (PGD) attack (Madry et al., 2017) and its variations (Zhang et al., 2019b; Wang et al., 2019b) lead to the most stable model, which has the best empirical robustness when facing different attacks. Thus, in this paper, we mainly focus on adversarial training methods.

Although we can obtain a robust model with PGD adversarial training (PGD-AT) (Madry et al., 2017), the main drawback lies in the heavy computational overhead. It needs multi-step gradient descents to generate adversarial examples before each mini-batch weight update. PGD-AT is an order of magnitude slower than the standard training, limiting its scalability to large datasets such as ImageNet (Deng et al., 2009). Though many works (Shafahi et al., 2019; Zhang et al., 2019a) are trying to accelerate PGD without sacrificing its performance, they are still much slower than standard training.

[†]Corresponding author: Yisen Wang (yisen.wang@pku.edu.cn).

In addition to accelerating the multi-step methods, fast gradient sign method (FGSM) (Goodfellow et al., 2014) only needs single gradient descent step to generate adversarial examples, significantly improving the computation efficiency. But, FGSM-AT was found to be easily broken by stronger multi-step attacks (Tramèr et al., 2018; Kurakin et al., 2016). However, recently Wong et al. (2019) claimed that by simply adding random uniform initialization before FGSM, the network can achieve comparable robustness as the one trained by PGD. Nevertheless, this approach cannot always defend against PGD due to the **catastrophic overfitting (CO)**, where the network suddenly loses robustness against PGD after a few training epochs and hardly recovers by itself. Although Wong et al. (2019) suggested to use a small validation set to evaluate the robustness and stop training before catastrophic overfitting is detected, the trained model is sub-optimal because of insufficient training.

In this regard, follow-up works (Andriushchenko & Flammarion, 2020; Vivek & Babu, 2020; Kim et al., 2021; Li et al., 2020) attempted to discover the underlying cause of catastrophic overfitting and prevent this failure. However, these approaches are either unintentional underfitting or computationally inefficient and can not provide a fundamental reason for this phenomenon. In this paper, we commit to understanding and preventing catastrophic overfitting in single-step based fast adversarial training by rethinking and decomposing the min-max problem. Most of the previous works are focused on improving linearity in the perturbation norm ball because the maximum loss along this direction will be reached on the boundary of the norm ball. However, we argued that the sign of second and higher order terms in Taylor expansion of the loss function at the clean example plays a vital role in CO, and linearity is a particular case where high-order terms are all zero. After comparing the model before and after CO, we found that the negative high-order term will induce a Distorted Perturbation Loss Curve, a form of gradient obfuscation. According to this assumption, we think the positive term can help prevent CO and propose a method named FLAT using linear assumption to penalize negative ones. Our work makes the following contributions:

- After revisiting the min-max optimization problem, we analyze the limits of FGSM compared with PGD and find the plus-minus sign of high-order terms of Taylor series is the underlying cause of CO. The negative high-order terms lead to a Perturbation Loss Distortion (PLD) phenomenon.
- We propose an indicator **LPR** to measure the sign of high-order terms and a simple regularization method named Fast Linear Adversarial Training (FLAT), which can avoid Perturbation Loss Distortion by explicitly penalizing negative **LPR**. FLAT makes the loss surface flat and prevents CO effectively.
- We evaluate the robustness of the proposed method against various adversarial attacks (FGSM, PGD, and AutoAttack) on different models and datasets and demonstrate that the proposed method can provide sufficient robustness for single-step adversarial training without catastrophic overfitting.

2 Background and Related Work

2.1 Adversarial Training

Adversarial training is the most effective defensive method to improve robustness. Given an i.i.d. example x from the underlying distribution $\mathcal{D}$, the adversarial example x' is generated by adding some imperceptible perturbations δ to x. Let $\ell(f_\theta(x), y)$ denotes the loss function of a deep neural network f with parameters θ. The core idea of adversarial training is to minimize empirical risk on the adversarial example x' instead of clean one x. More concretely, adversarial training can be formulated as the following optimization problem (Madry et al., 2017):

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \Big[\max_{\delta\in\Delta} \ell(f_\theta(x+\delta), y) \Big] \tag{1}$$

The threat model or perturbation set $\Delta = \{\delta : \|\delta\|_p \leq \epsilon, \epsilon > 0\}$ denotes the ϵ-ball around the clean example x with a specific distance metric. The most used metric are L_0, L_1 and L_∞. In this paper, we focus on l_∞ norm bounded threat model. There are numerous methods to solve the outer empirical risk minimum problem, such as SGD (Robbins & Monro, 1951) and Adam (Kingma & Ba, 2015). So the primary goal is to find $\delta^* \in \Delta$ of a given example x that maximizes the inner loss function, i.e. $\delta^* = \arg\max_{\delta\in\Delta} \ell(x+\delta)$. However, the above optimization is considered an NP-hard

problem (Weng et al., 2018) because it contains a non-convex min-max problem. Thus, different adversarial training methods use various strategies to find a sub-optimal approximate solution to the inner maximization problem.

Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2014) is the simplest adversarial attack method, which uses the sign of local gradient to find an adversarial examples x'

$$x' = x + \epsilon \cdot \text{sgn}(\nabla_x \ell(x, y; \theta)) \tag{2}$$

Projected Gradient Descent (PGD) (Madry et al., 2017) uses multiple gradients to construct stronger adversarial examples. With a step size α, PGD can be formalized as follows:

$$x'_{t+1} = \Pi(x'_t + \alpha \cdot \text{sgn}(\nabla_x \ell(x'_t, y; \theta))) \tag{3}$$

where Π refers the projection to the ϵ-ball, and x'_t is the adversarial example after t-th steps.

After multi-step methods were proposed, the single-step FGSM was believed to be a non-robust method for its failure defense against multi-step attacks. So the follow-up studies mainly focused on improving and accelerating multi-step adversarial training. FreeAT proposed by Shafahi et al. (2019) has achieved remarkable robustness with accumulative perturbations constructed by mini-batch replay. Zhang et al. (2019a) considered that adversarial perturbation is only related to the first layer of the network according to Pontryagin's Maximum Principle (PMP). So they proposed YOPO, which fixed and shared the deep layer gradient, and computes the gradient w.r.t. the first layer several times to update the perturbation. However, contrary to the perception that single-step methods are not strong enough, the FastAT proposed by Wong et al. (2019) can obtain almost equivalent robustness to FreeAT with uniform random initialization before FGSM.

2.2 Catastrophic Overfitting

Although single-step adversarial training can achieve certain robustness in the initial stages of training, the accuracy of the network against multi-step attacks may decrease suddenly and sharply after a few epochs. This phenomenon is named catastrophic overfitting. FastAT uses early stopping to temporarily alleviate CO, which needs to track robustness against PGD on a small validation set. BS & Babu (2020) argued that CO arises with early overfitting to FGSM perturbations and empirically shows that adding a dropout layer after all non-linear layers can attain comparable robustness as the one trained by stronger attacks. Li et al. (2020) claimed the success factor of FastAT is the ability to recover from CO and proposed a simple strategy: switch to PGD once CO is detected and switch back to FGSM after recovery. This method is simple but does not explain why multi-step can help the model recover from CO. Andriushchenko & Flammarion (2020) thought FGSM is equivalent to PGD if gradients in the norm space around clean examples are constant. Based on this assumption, they proposed a regularization method named GradAlign, which prevents CO by explicitly maximizing the gradient alignment $\cos(\nabla_x \ell(x), \nabla_x \ell(x + \eta))$ inside the perturbation set. Kim et al. (2021) found that the decision boundary is highly curved when CO happens and hypothesized that the fixed magnitude of the perturbation is the cause of CO. They set c checkpoints in the local gradient direction to search appropriate step size and select the smallest perturbation that fools the network. Most of the previous works understand and avoid CO based on the first-order linear approximation of the loss function. However, we focus on higher order terms of approximation, e.g. second-order quadratic approximation, and find that the plus-minus sign of the high-order terms is the underlying cause of CO.

3 Fast Linear Adversarial Training

3.1 Rethink min-max Problem

The essence of adversarial training is to find the optimal solution or saddle point for the robust optimization problem shown in Equation 1. However, the closed-form solution of this min-max problem can not be directly calculated as it is NP-hard. So, iteration methods like PGD are applied to find a sub-optimal numerical solution. In the previous studies, FGSM-AT and PGD-AT are considered only different in solving the inner maximization problem. Here, we provide another understanding of PGD-AT according to its iterative nature. For PGD-K adversarial training, let us assume the

perturbation constructed after the i-th iteration is δ_i. To simplify the expression, we ignore the projection operation after each iteration, i.e. $\delta_{i+1} = \delta_i + \alpha \cdot \text{sign}(\nabla_x \ell(x + \delta_i))$. Also, we suppose the initial perturbation is 0 and the perturbation found at the K-th iteration is exectly the optimal solution for the inner maximization, i.e. $\delta_0 = 0, \delta_K = \delta^*$. Then, the Equation 1 can be rewritten as

$$\begin{aligned} \min_\theta \ell(x + \delta^*) &= \min_\theta \ell(x + \delta_K) \\ &= \min_\theta \{[\ell(x + \delta_K) - \ell(x + \delta_{K-1})] + \cdots + [\ell(x + \delta_1) - \ell(x + \delta_0)] + \ell(x)\} \\ &= \min_\theta \ell(x) + \min_\theta \sum_{i=0}^{K-1} [\ell(x + \delta_{i+1}) - \ell(x + \delta_i)] \end{aligned} \tag{4}$$

where we abbreviate $\ell(f_\theta(x), y)$ with $\ell(x)$. Previously, adversarial training was usually viewed as minimizing empirical risk on adversarial examples directly. However, after decomposing the adversarial loss $\ell(x + \delta^*)$ according to the iterative construction of adversarial examples, the min-max problem can be expressed as standard training plus K penalty terms as shown in Equation 4. The penalty term is composed of the loss difference in each gradient ascent iteration. When K equals 0, adversarial training degenerates to standard training as there is no penalty term. FGSM-AT is another particular case of PGD-AT with $k = 1$ and $\alpha = \epsilon$. Thus, the in-depth analysis of penalty terms can help understand the limits of single-step adversarial training and the cause of Catastrophic Overfitting.

3.2 Understanding of Catastrophic Overfitting

In practice, the classifier networks are trained with differentiable surrogate loss function, such as cross-entropy, rather than non-differentiable 0-1 loss. The loss function around the clean example is continuous and relatively smooth (Simon-Gabriel et al., 2019; Qin et al., 2019). So, the penalty term $\ell(x + \delta) - \ell(x)$ of FGSM-AT can be represented with Taylor expansion as shown in the first line of Equation 5, where $\langle \cdot \rangle$ is the inner product, $\nabla_x \ell(x)$ is the Jacobian matrix, $\nabla^2_{xx} \ell(x)$ is the Hessian matrix, and $o(\delta^3)$ denotes the approximation error consisting of three and higher order infinitesimal of δ. Unless mentioned otherwise, we refer to the second and higher order as high-order. For single-step perturbation, we use d to represent the local gradient direction of a clean example, i.e. $d = \text{sgn}(\nabla_x \ell(x))$. Then, the perturbation constructed by FGSM is $\delta = \alpha d$, where $\alpha = \epsilon$. For the n-th term of the Taylor series, it can be expressed as $\alpha^n h^{(n)}(x; d)$ by separating α from the term, where $h^{(n)}(x; d)$ consists of the n-order derivative and d. Then, Taylor expansion can be decoupled as the inner product of a coefficient vector and a derivative vector. The former vector consists of a geometric sequence of α, while the latter is only relevant to clean example and network parameters. The derivative vector will not change during the construction of the adversarial example, so these two vectors are independent and do not influence each other.

$$\begin{aligned} \ell(x + \delta) - \ell(x) &= \langle \nabla_x \ell(x), \delta \rangle + \langle \delta, \nabla^2_{xx} \ell(x) \delta \rangle + o(\delta^3) \\ &= \alpha \cdot \langle \nabla_x \ell(x), d \rangle + \alpha^2 \cdot \langle d, \nabla^2_{xx} \ell(x) d \rangle + o(\delta^3) \\ &= \alpha \cdot \|\nabla_x \ell(x)\|_1 + \alpha^2 \cdot d^\top \nabla^2_{xx} \ell(x) d + \sum_{n=3}^{\infty} \alpha^n h^{(n)}(x; d) \end{aligned} \tag{5}$$

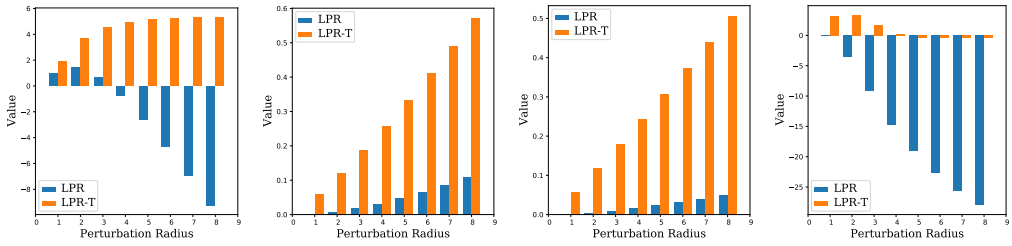
Because the coefficients of high-order terms decrease exponentially with the order, these terms can be ignored for the small step size such as $8/255$. Hence, the loss surface in the small norm ball is approximately linear, which implies that the maximum loss along this direction is reached on the boundary of the norm ball. However, as α increases, the penalty term may be dominated by high-order of Taylor series because the influence of higher-order terms also increases exponentially, while that of the first-order increases linearly. Especially for single-step perturbation, the first-order term is always positive for the local gradient direction is the same as that of perturbation, but high-order terms are not guaranteed. Considered that the goal is to minimize the $\ell(x + \delta) - \ell(x)$, which equals to minimize the sum of all terms in Taylor expansion, we suppose that this unknown plus-minus of high-order terms is precisely the crux of CO. For FGSM-AT, the step size α is relatively large, so high-order terms dominate in Taylor expansion and become the primary objective of the optimization. Intuitively, when high-order terms are less than 0, the optimizer may allow the first-order term increase to minimize the overall sum. During the training, if the norm of the first-order derivative gradually increases, FGSM will continue to select this direction because FGSM

determines the perturbation direction based on first-order derivative. This vicious circle induces a more distorted loss surface, causing the model to fall into CO and hardly recover from it.

$$\textbf{LPR-T} \stackrel{\text{def}}{=} \ell(x+\delta) - \ell(x) - \delta^\top \nabla_x \ell(x) = \sum_{n=2}^{\infty} \alpha^n h^{(n)}(x; d) \tag{6}$$

To prove our argument, we show that there are negative high-order terms in the Taylor series after CO happened. However, direct calculation of all terms requires corresponding high-order partial derivatives, which will consume heavy computation, so we use the surrogate indicator defined in Equation 6 to calculate the sum of high-order terms. We call this indicator Taylor-based Linear Perturbation Rate (LPR-T) because the loss surface is approximately linear when the indicator is close to 0. The LPR-T can indicate whether negative terms exist through the curve of LPR-T value w.r.t. α. As α gradually increases from 0 to ϵ, the corresponding LPR-T value will increase if all Taylor series terms are positive. On the contrary, when the value is not monotonically increasing, it indicates at least a negative term in series. The changing trend of LPR-T w.r.t. α under different models is shown in Figure 1, where only the LPR-T value of FGSM-AT with CO is not monotonically increasing. In fact, the LPR-T value of the model with CO is directly negative for large radius, and this proves our hypothesis that the negative high-order terms are closely related to CO. Kim et al. (2021) also found that the L_2 norm of the local gradient increases after CO happens, which can further validate our argument that the optimizer minimizes the sum of all terms at the cost of first-order terms.

Furthermore, the influence of step size and steps of PGD on catastrophic overfitting are well studied. We found that in the model with CO, even if the step size was large, the LPR-T curve still had monotonically increasing properties as long as the steps were enough. The complementary experiment also reveals that CO will not occur in PGD-10 adversarial training with $\alpha = \epsilon$. This result suggests the viewpoint of "fixed magnitude of the perturbation cause CO" proposed by Kim et al. (2021) is debatable. More details are discussed in the Appendix A.2.

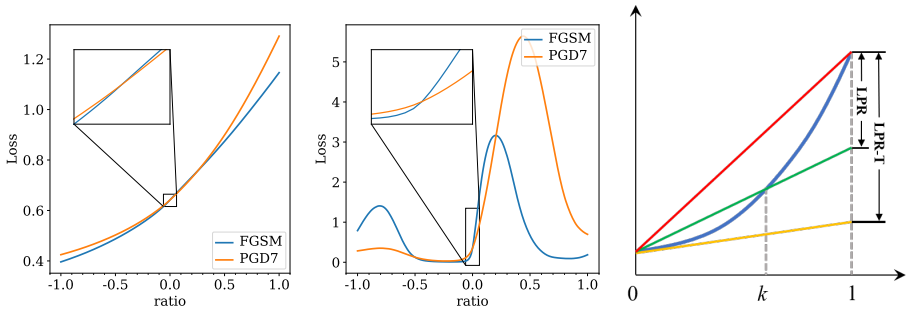


(a) Standard (b) PGD (c) FGSM w/o CO (d) FGSM w/ CO

Figure 1: LPR & LPR-T value in four different models. Step size α is equal to perturbation radius divided by 255 for CIFAT10. LPR value is measured with $k = 0.1$. PGD and FGSM without CO have a similar LPR-T curve, which is almost linearly increasing. LPR-T value of standard model is also monotone increasing but the curve is concave. The FGSM with CO has a completely different LPR-T curve, and the value for a large radius is even negative.

3.3 Proposed Method

The analysis above shows that negative high-order terms exist in the Taylor series after CO occurs. The consequence of negative terms is that the adversarial loss is not monotonically increasing as the perturbation radius increases along the FGSM direction. This argument can be further proved in Figure 2, where we use FGSM and PGD-7 respectively to compute perturbation direction and plot the curve of loss w.r.t. radius along that direction. A distortion in the perturbation-loss curve can be found when CO happens. Kim et al. (2021) also discovered such phenomenon and called it decision boundary distortion. Their proposed StableAT searches the smallest perturbation radius that can fool the network no matter whether CO occurs or not. Unlike StableAT, which decreases step size as long as the distortion is detected, we attempt to avoid this distortion directly.



(a) w/o CO (b) w/ CO (c) Geometric Interpretation

Figure 2: Distorted Perturbation Loss Curve and Geometric Interpretation of LPR & LPR-T. In (a) and (b), loss w.r.t. perturbation are plotted along direction found by FGSM and PGD7 respectively. FGSM uses the sign of first-order derivative as perturbation direction, so the loss of FGSM is greater than that of PGD around the clean example, but PGD can find the maximum loss.

$$\begin{aligned} \textbf{LPR} &\stackrel{\text{def}}{=} \ell(x+\delta) - [\ell(x) + \frac{\ell(x+k\cdot\delta)-\ell(x)}{k}] \\ &= \frac{1}{k}[(1-k)\ell(x) + k\ell(x+\delta) - \ell(x+k\cdot\delta)] \end{aligned} \tag{7}$$

Intuitively, the LPR-T metric mentioned in Section 3.1 can be used as a regularizer directly. However, it is the tendency of LPR-T w.r.t. α that reflects the sign of high-order terms and single LPR-T value is meaningless. Furthermore, the first-order derivatives need to be optimized with this regularizer, and only local gradient information is used. To address the limitations of LPR-T, we propose another alternative indicator LPR as shown in Equation 7, where k is in the range of 0 to 1. The difference between LPR and LPR-T can be easily clarified with Geometric Interpretation in Figure 2, where the LPR metric uses secant line to approximate tangent line in LPR. k control the approximation effect of LPR, and the closer k approaches 0, the smaller the difference between LPR and LPR-T. Compared with LPR-T, the neighborhood loss information is used without optimizing on the first-order derivative. Another reason for using this indicator is that it reflects the concavity of the surface. For $\forall\ k \in (0,1)$, if LPR is greater than 0, then the perturbation-loss curve in this direction is convex, which means the second-order derivative is also greater than 0 and this is in line with our previous analysis. Figure 1 illustrates that the LPR value is negative as long as the LPR-T curve is not convex.

According to the assumption above, we propose the Fast Linear Adversarial Training(FLAT) method to avoid catastrophic overfitting in single-step adversarial training. We believe that CO will not occur when the LPR value is greater than 0, and the network will be trained with adversarial loss directly. However, when the LPR value is negative, the perturbation-loss curve is distorted and CO may happen, so we use LPR as a regularizer to penalize the negative high-order terms. The complete FLAT loss is shown in Equation 8, and the Algorithm 1 shows a summary of the proposed method.

$$\mathcal{L}_{\text{FLAT}} = \ell(x+\delta) - \lambda \cdot \min(\textbf{LPR}, 0) \tag{8}$$

The core concept of FLAT is to utilize the LPR as an indicator and regularizer to penalize the negative high-order terms. According to previous analysis, LPR is reasonable as an indicator, and we will also show that LPR is also interpretable as a regularizer. By simple equation deformation, the FLAT loss under negative LPR can be transformed into Equation 9. It shows that the LPR regularizer will balance between standard loss and adversarial loss and constrain the Lipsitz constant in the perturbation direction. As k approaches 0, the FLAT loss can be approximated by the following Equation 10. However, as mentioned before, when k is infinitely approaching 0, LPR is almost as same as LPR-T. In practice, k will not close 0 to much and is usually between 0.1 and 0.2.

$$\mathcal{L}_{\text{FLAT}} = (1-\lambda)\ell(x+\delta) + \lambda[\ell(x) + \frac{\ell(x+k\cdot\delta)-\ell(x)}{k}] \tag{9}$$

$$\begin{aligned} \lim_{k\to 0}\mathcal{L}_{\text{FLAT}} &= (1-\lambda)\ell(x+\delta) + \lambda[\ell(x) + \lim_{k\to 0}\frac{\ell(x+k\cdot\delta)-\ell(x)}{k}] \\ &\approx (1-\lambda)\ell(x+\delta) + \lambda[\ell(x) + \epsilon\cdot\|\nabla_x\ell(x)\|_1] \end{aligned} \tag{10}$$

Although not intended to avoid CO, Qin et al. (2019) also proposed Local Linearity Regularization(LLR) based on Taylor expansion to improve local linearity. LLR in Equation 11 forces both first-order and high-order terms to approach 0, while our method concentrates on high-order terms and encourages them to become positive rather than close to 0. In fact, we found that if we force LPR close to 0, LPR will converge to 0 from the negative side and CO cannot be effectively avoided.

$$\text{LLR} = \lambda \left| \ell(x+\delta) - \ell(x) - \delta^\top \nabla_x \ell(x) \right| + \mu \left| \delta^\top \nabla_x \ell(x) \right| \tag{11}$$

4 EXPERIMENT

This section provides an empirical understanding of CO and FLAT through two metrics: LPR and FOSC(Wang et al., 2019a). Then we show the effectiveness of FLAT through some ablation studies. At last, we conduct comprehensive experiments to evaluate the robustness of FLAT and compare FLAT with other adversarial training accelerating methods. The following experiments and evaluations are conducted on CIFAR10(Krizhevsky et al., 2009) with ResNet18 (He et al., 2016) under the threat model $\|\delta\|_\infty \leq 8/255$. More experiments on other datasets, models and large perturbations can be found in Appendix B.

4.1 EMPIRICAL UNDERSTANDING OF FLAT

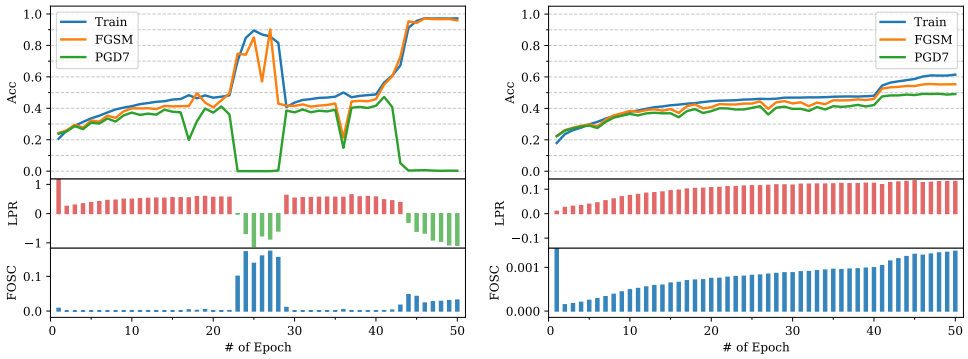
Because of a highly curved loss surface in the network with CO, the single-step adversarial attack hardly finds the perturbation that maximizes the inner loss. We think such distorted surface is a form of gradient obfuscation for the single-step method and hypothesize that there will be severe gradient obfuscation in the CO model. However, gradient obfuscation is a phenomenon rather than a metric that can be calculated directly, so we use First-Order Stationary Condition(FOSC) proposed by Wang et al. (2019a) to measure the severity of gradient obfuscation. The FOSC is originally proposed as the quantitative convergence criterion for the inner maximization problem of Equation 1. A smaller value of $\text{FOSC}(x')$ indicates a better solution of the inner maximization problem or equivalently the stronger the adversarial example x' is. If gradient obfuscation exists in the network, then the adversarial example constructed by a single step is not strong enough. Therefore, the larger the FOSC is, the more serious the gradient obfuscation is. The FOSC criterion for the single-step adversarial example have the following closed-form solution:

$$\text{FOSC}(x') = \epsilon \left\| \nabla_x \ell(x', y; \theta) \right\|_1 - \langle x' - x, \nabla_x \ell(x', y; \theta) \rangle \tag{12}$$

Experimental Settings. Following the baseline setting suggested by Pang et al. (2020), We adversarially train ResNet18 on CIFAR10 for 50 epochs using SGD optimizer with batch size 128, momentum 0.9, weight decay 5×10^{-4}, an initial learning rate of 0.1 that is divided by 10 at the 40-th and 45-th epoch, i.e. stepwise learning rate scheduling. Simple data augmentations such as 32×32 random crop with 4-pixel padding and random horizontal flip are applied. Both FGSM-AT and FLAT use the same setting and are initialized with the same random parameters. We track the robustness against FGSM and PGD-7 on the test set as well as the LPR and FOSC value on the train set during training.

Effectiveness of the FLAT. Figure 3 illustrates the relationship between CO and LPR metrics. In the initial stage of FGSM-AT, the LPR value was positive and FOSC was relatively small, so the robustness against FGSM and PGD-7 improved simultaneously. However, the FOSC value became relatively large after CO, which means there exists gradient obfuscation in the network and the constructed adversarial examples were not strong enough. At the same time, the LPR value also became negative. Before learning rate decay, the network has the probability of recovering from CO, but hardly recover after decay. In contrast, under the effect of FLAT regularizer, the LPR value kept positive and CO never occurred. Though the FOSC value increased slowly, it is still extremely small in the order of compared with that of FGSM-AT. Furthermore, we draw the loss surface to show the effectiveness of FLAT in Appendix A.3, which illustrates that model with CO has a distorted surface while the FLAT makes the surface flat.

Ablation studies on λ. With the same experiment setting mentioned before, we fixed $k = 0.1$ and varied λ to verify the effectiveness of FLAT. Compared with FGSM-AT, the clean example x and weak adversarial example $x+k\cdot\delta$ will go through BatchNorm layers, so we also check the robustness when $\lambda = 0$ to exclude the influence of BatchNorm particularly. According to the Figure 4(a), there

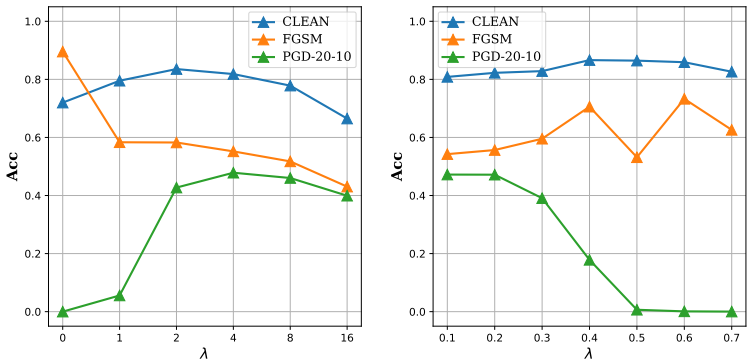


(a) FGSM-AT (b) FLAT

Figure 3: Underlying connection between CO and LPR, FOSC metric during training. LPR value are measured under $k = 0.1$. For FGSM-AT, the network fell in CO at epoch 23 and recovered at epoch 29, and fell in CO again at epoch 43.

still exists CO in the small λ, but the model can obtain robustness as λ increase exponentially. For larger λ, we observe a decrease in both clean and adversarial accuracy since the model becomes overregularized.

Ablation studies on k. Hyper-parameter k determines the linear approximation quality of the LPR. The LPR metric is closer to linear approximation as k approaches 0. The Figure 4(b) shows that for the fixed $\lambda = 5.0$, the catastrophic overfitting phenomenon still occurred when $k \geq 0.5$. This failure of FLAT can be easily explained according to the Figure 2(b). When the ratio is greater than 0.5, the loss of the corresponding perturbation is close to that of clean examples, which means the LPR value is almost equivalent to 0 though it is negative. In this case, the effect of the regularizer is inadequate, and we need to increase λ to find the appropriate value.



(a) Influence of λ (b) Influence of k

Figure 4: Ablation studies on λ and k.

4.2 Robustness Evaluation

In this part, we evaluated the robustness of FLAT and compared it with other single-step methods: 1) FastAT(Wong et al., 2019), 2) GradAlign(Andriushchenko & Flammarion, 2020), 3) StableAT(Kim et al., 2021), as well as multi-step acceleration methods: 1) YOPO(Zhang et al., 2019a), 2) FreeAT(Shafahi et al., 2019).

Experimental Settings. The adversarial training settings on CIFAR-10 with ResNet18 are the same as Section 4.1. For our FLAT, we use $\lambda = 4.0$ and k random selected from set $\{0.1, 0.2, 0.4, 0.8\}$

with probability 0.4, 0.3, 0.2, 0.1 respectively in each mini-batch. For other contrast methods, we use hyper-parameters that achieve the best robustness according to their paper. We evaluate the robustness of all methods against three types of adversarial attacks: FGSM, PGD-20-10, i.e. PGD with 20 steps and 10 random restarts with step size $\alpha = 1/255$, and latest strongest adaptive AutoAttack. All experiments are conducted on a single NVIDIA Tesla P100 over five different seeds range from 0 to 4.

Table 1: Test accuracy (%) and training time (sec/epoch) on CIFAR10 with ResNet18. The results are averaged over 5 random seeds and reported with the standard deviation.

	Method	**CLEAN**	**FGSM**	**PGD-20-10**	**AA**	**Time**
	Standard	94.04±0.19	17.72±1.36	0.0±0.0	0.0±0.0	24.1
Multi-step	PGD10	82.58±0.19	56.90±0.11	51.45±0.12	47.52±0.17	244.9
	YOPO-3-5	86.36±0.13	53.20±0.20	42.78±0.28	39.91±0.27	85.4
	YOPO-5-3	84.83±0.10	53.02±0.32	44.86±0.20	41.83±0.17	128.1
	FreeAT	81.67±0.10	52.14±0.21	46.21±0.19	42.44±0.07	95.3
Single-step	FGSM	71.80±5.32	97.04±3.49	0.04±0.04	0.01±0.01	47.8
	FastAT	85.78±1.01	59.05±7.30	37.03±18.00	33.99±16.93	48.1
	StableAT	87.15±0.16	49.34±0.09	36.72±0.49	33.97±0.41	64.5
	GradAlign	82.11±0.10	55.40±0.19	48.54±0.27	44.39±0.27	191.4
	FLAT	82.27±0.15	55.93±0.23	47.81±0.17	43.45±0.21	87.2

Result on CIFAR10. Table 1 summarizes the adversarial robustness of methods achieved after the last training epoch. We also reported the time each epoch consumed as some methods may use cyclic learning rate scheduling to speed up the convergence, and it is not fair to compare total training time. According to Table 1, yields the most robust model but requires heavy computational time. Other multi-step acceleration methods consume much less time but sacrifice some robustness. Among the single-step methods, the FGSM-AT is most computationally efficient and achieves almost 100% accuracy against FGSM but shows non-robust against multi-step attack because the occurrence of CO. FastAT can achieve some degree of robustness but may suffer from CO. In fact, by tracking the training accuracy, we found that CO occurred both in FGSM-AT and FastAT during training. However, compared with FGSM-AT, FastAT has a higher probability of recovering from CO and eventually achieving some robustness. This result shows shows that random start is helpful to alleviate CO. StableAT can completely avoid CO in the process of training. Nevertheless, the model trained with StableAT may be underfitting according to the achieved robustness. GradAlign is another effective method, but the consumed time is almost the same as PGD-10 because of double backpropagation and optimization on the first-order derivative. The FLAT can effectively and efficiently prevent catastrophic overfitting. Compared with multi-step accelerating methods and single-step methods except for GradAlign, our method can achieve the best robustness without too much additional computational cost. Even for GradAlign, our method only has a slight robustness drop, but with a significant computation time improving because FLAT is a derivative-free method.

5 Conclusion

In this paper, we theoretically analyzed and empirically showed the underlying connection between high-order terms in Taylor expansion and catastrophic overfitting through the decomposition of the min-max problem. When there exists negative high-order term, single-step adversarial training prioritizes optimizing these negative terms at the cost of first-order term growth. This optimization leads to a distortion of the perturbation loss curve, a form of gradient obfuscation that prevents the FGSM from finding the perturbation that maximizes the inner loss function. Based on these observations, we proposed a derivative-free indicator **LPR** to measure the plus-minus sign of high-order terms and a new simple regularization method named `FLAT`, which explicitly penalizes negative **LPR** value to prevent CO in single-step adversarial training. `FLAT` makes the loss surface flat, and the perturbation found by FGSM is equivalent to that found by PGD. Furthermore, we evaluated the robustness of the proposed method against various adversarial attacks, and `FLAT` showed sufficient robustness using single-step adversarial training without the occurrence of catastrophic overfitting.

REFERENCES

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

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*, pp. 274–283. PMLR, 2018.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 387–402. Springer, 2013.

Vivek BS and R Venkatesh Babu. Single-step adversarial training with dropout scheduling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 950–959, 2020.

Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *International Conference on Learning Representations*, 2018.

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

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

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.

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

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*, pp. 770–778, 2016.

Hoki Kim, Woojin Lee, and Jaewook Lee. Understanding catastrophic overfitting in single-step adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8119–8127, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

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

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

Bai Li, Shiqi Wang, Suman Jana, and Lawrence Carin. Towards understanding fast adversarial training. *arXiv preprint arXiv:2006.03089*, 2020.

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.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness via curvature regularization, and vice versa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9078–9086, 2019.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. In *International Conference on Learning Representations*, 2020.

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*, 32:13847–13856, 2019.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.

Andrew Slavin Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Thirty-second AAAI conference on artificial intelligence*, 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*, 32:3358–3369, 2019.

Carl-Johann Simon-Gabriel, Yann Ollivier, Leon Bottou, Bernhard Schölkopf, and David Lopez-Paz. First-order adversarial vulnerability of neural networks and input dimension. In *International Conference on Machine Learning*, pp. 5809–5817. PMLR, 2019.

Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *International Conference on Learning Representations*, 2018.

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.

F Tramèr, D Boneh, A Kurakin, I Goodfellow, N Papernot, and P McDaniel. Ensemble adversarial training: Attacks and defenses. In *6th International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings*, 2018.

Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020.

BS Vivek and R Venkatesh Babu. Single-step adversarial training with dropout scheduling. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 947–956. IEEE, 2020.

Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. On the convergence and robustness of adversarial training. In *International Conference on Machine Learning*, pp. 6586–6595. PMLR, 2019a.

Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*, 2019b.

Lily Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane Boning, and Inderjit Dhillon. Towards fast computation of certified robustness for relu networks. In *International Conference on Machine Learning*, pp. 5276–5285. PMLR, 2018.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*, 2019.

Dinghuai Zhang, Tianyuan Zhang, Yiping Lu, Zhanxing Zhu, and Bin Dong. You only propagate once: Accelerating adversarial training via maximal principle. *Advances in Neural Information Processing Systems*, 32:227–238, 2019a.

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, 2019b.

A SUPPORTING EXPERIMENTS AND VISUALIZATIONS

In this part, we describe some supporting experiments and visualizations mentioned in Section 3.

A.1 FLAT ALOGRITHM

Algorithm 1: Fast Linear Adversarial Training(FLAT)

Parameter: T epochs, B mini-batches, perturbation radius ϵ, and a network f_θ. Hyper parameter k and λ for FLAT.

Initialize model weights θ;

for $t = 1, \cdots, T$ **do**

  for $i = 1, \cdots, B$ **do**

    // Perform FGSM adversarial attack

    $\delta = \epsilon \cdot \text{sign}(\nabla_x \ell(x_i, y_i; \theta))$;

    $\delta = \text{CLAMP}(x_i + \delta, 0, 1) - x_i$;

    $\textbf{LPR} = (1-k) \cdot \ell(x_i, y_i; \theta) + k \cdot \ell(x_i + \delta, y_i; \theta) - \ell(x_i + k \cdot \delta, y_i; \theta)$;

    $\mathcal{L} = \ell(x_i + \delta, y_i; \theta) - \lambda \cdot \min(\textbf{LPR}, 0)$;

    // Update model weights with some optimizer, e.g. SGD ;

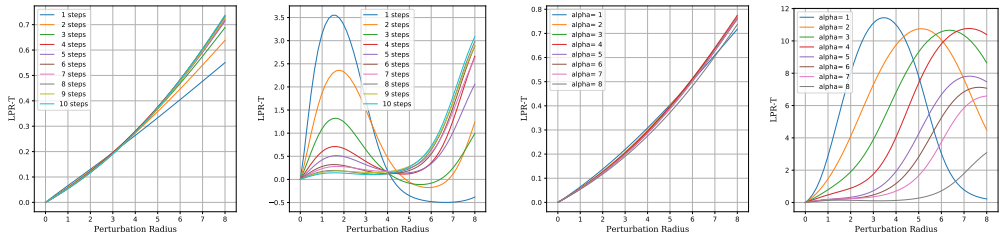
    $\theta = \theta - \nabla_\theta \mathcal{L}$;

  end

end

A.2 CATASTROPHIC OVERFITTING IN VARIANT PGD

From Equation 4, we have found two factors, step k and step size α, determine the quality of the adversarial example. To investigate the influence of these factors, we study the tendency of LPR-T w.r.t. perturbation radius along the direction found by PGD with various k and α. For the impact of steps, we fixed step size $\alpha = 8/255$, and vary steps from 1 to 10. For the influence of step size, we fixed steps $k = 10$, and vary step size from $1/255$ to $8/255$. We compare the difference between the model before and after CO. From Figure 5, we can find that no matter how k and α change, the LPR-T curve are always monotone increasing for the model without CO. For the CO model, there is distortion in the LPR-T curve when $k = 1$. However, as the step increases, the curve becomes monotone increasing. We can have the first hypothesis that when the step is enough, CO will not occur even if step size is relatively large. We can also find that when α is small, PGD will find a perturbation that achieves maximum inner loss but with a highly distorted LPR-T curve along this direction. Then, we have the second hypothesis that if we project perturbation found by PGD to norm ball boundary, CO may still happen.

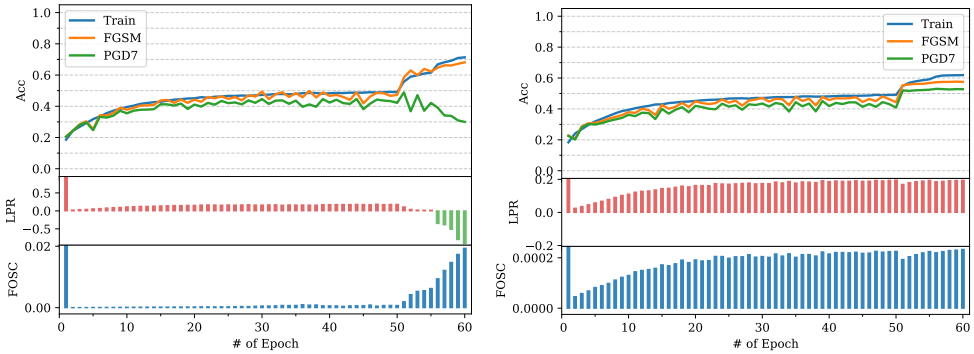


(a) various steps k (b) various step size α

Figure 5: The LPR value w.r.t. perturbation radius along direction found by PGD with various k and α. The left part of each subplot is model without CO, while the right part is model with CO.

We use two variant versions of PGD to prove our two hypothesises. It is obviously that perturbation found by PGD-K span the whole l_∞-ball $[-\epsilon, \epsilon]^{C \times H \times W}$, while that of FGSM only located at corner $\{-\epsilon, \epsilon\}^{C \times H \times W}$. By modifying the vanilla PGD, we can have two PGD variant. One is PGD-corner, which projects perturbation found by PGD-K onto $\{-\epsilon, \epsilon\}^{C \times H \times W}$. The other is PGD-large,

which is the same as PGD-k but with large step size, e.g. $\alpha = \epsilon$, and the perturbation located at $\{-\epsilon, 0, \epsilon\}^{C\times H\times W}$. Figure 6 illustrates robustness and metrics during the training progress of variant PGD-AT. The PGD-corner has the robustness drop after the learning rate decay, which means there is a slight CO phenomenon, while PGD-large can maintain robustness after decay. This result not only further proves the relationship between CO and LPR, but also proves our hypothesis.



(a) PGD-corner (b) PGD-large

Figure 6: Training process of PGD variant version. PGD-corner suffer from CO after learning rate decay, while PGD-large still maintain robustness.

A.3 Loss surface

The loss surface can intuitively reflect the cause of CO and the effectiveness of FLAT. Since the loss function of the neural network is in a high-dimensional space and can not be drawn directly, we use directions found by FGSM and PGD as u-axis and v-axis to plot the loss surface. According to Figure 7(a) and Figure 7(b), the loss surface is relatively flat before CO occurrs while distorted after CO. This distortion lead to the FGSM failing to find the maximum loss. The Figure 7(c) and Figure 7(d) is the surface of the model under $\lambda = 1.0$ and $\lambda = 0$ respectively. The FLAT prevents CO by making the distorted loss surface flat, and the perturbation found by FGSM is equivalent to that found by PGD.

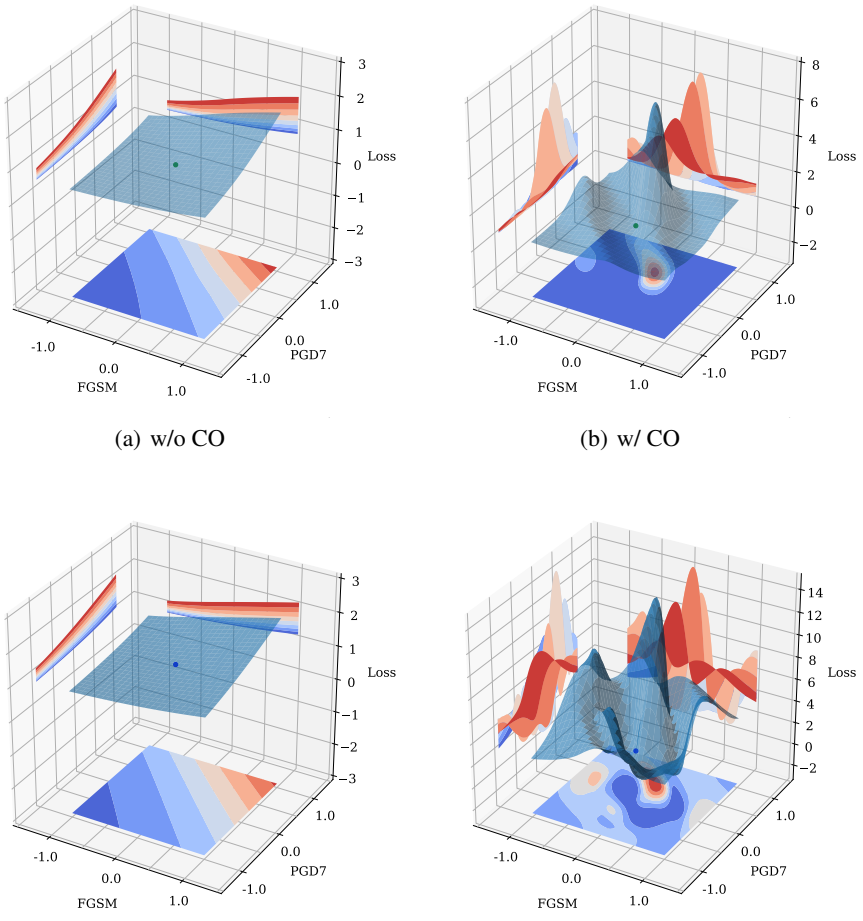
B Additional experiments

B.1 Different models

In this part, we show the adaptability of FLAT on more different residual-based architectures, including PreActResNet-18 and WideResNet-28-10. The setting of WideResNet-28-10 is the same as ResNet18. For PreActResNet-18, the experiment settings are almost the same as ResNet, except the initial learning rate is 0.05, because 0.1 is too large and hard to convergence. The Table B.1 shows that FLAT can performs well in other architectures, and the results are consistent with the hypothesis that the larger the network capacity, the better the robustness.

B.2 Different datasets

In this part, we show the adaptability of FLAT on different datasets, including CIFAR100 and SVHN. The setting of CIFAR100 is the same as CIFAR10, while that of SVHN has a slight difference. For SVHN, there is no random horizontal flip in data augmentation and the perturbation radius is increased from 0 to ϵ linearly in the first 5 epochs to prevent convergence to a constant classifier. The initial learning rate for SVHN is 0.01.



(a) w/o CO

(b) w/ CO

(c) FLAT($\lambda = 1.0$)

(d) FLAT($\lambda = 0.0$)

Figure 7: Loss surface along FGSM direction u and PGD-7 direction v for different models. (a) and (b) are the model before and after the CO occurrs. (c) and (d) are the model 1 epoch after the model (b) trained with FLAT under different λ.

Table 2: Test accuracy (%) on CIFAR10 with Different model.

	Method	**CLEAN**	**FGSM**	**PGD-20-10**	**Time**
PreActResNet-18	Standard	93.15	10.52	0.0	24.0
	PGD10	80.08	54.36	48.08	244.6
	FastAT	86.01	55.80	44.46	45.1
	FLAT	83.23	54.53	46.35	86.5
WideResNet-28-10	Standard	94.70	23.39	0.0	112.4
	PGD10	84.96	59.63	53.27	1197.8
	FastAT	78.43	77.30	0.0	225.4
	FLAT	85.49	58.89	49.51	425.7

B.3 Larger Perturbation

In this part, we show the ability of FLAT on larger perturbations. For single-step adversarial training, the larger the disturbance, the more likely CO occurs. So the robustness of FLAT under larger perturbation can further verify the effectiveness of FLAT. We found the failure of GradAlign at

Table 3: Test accuracy (%) on Different datasets with ResNet18.

	Method	**CLEAN**	**FGSM**	**PGD-20-10**	**Time**
CIFAR100	Standard	76.05	6.84	0.0	24.2
	PGD10	58.67	31.48	27.51	243.3
	FastAT	61.83	59.89	0.07	48.3
	FLAT	56.92	31.35	26.46	89.8
SVHN	Standard	95.55	18.61	1.36	34.7
	PGD10	93.32	64.96	52.81	306.8
	FastAT	92.69	90.32	0.0	68.5
	GradAlign	93.39	64.23	48.08	277.2
	FLAT	94.02	64.05	47.85	129.6

extremely large perturbations such as $\epsilon \geq 15/255$, while the FLAT still maintains some robustness. However, as the perturbation radius increase, the robustness between PGD-10 and FLAT is also getting bigger, which requires further research.

Table 4: Robustness (%) on larger radius.

radius	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
PGD-10	47.52	44.62	42.53	39.78	38.64	36.78	34.92	33.36
GradAlign	44.35	42.24	38.73	37.58	34.69	33.22	10.03	10.03
FLAT	43.86	41.05	37.94	36.24	32.76	30.18	28.15	26.02