

Rethinking Fast Adversarial Training: A Splitting Technique to Overcome Catastrophic Overfitting

Masoumeh Zareapoor[✉] and Pourya Shamsolmoali^{*✉}

East China Normal University, Queen’s University Belfast

Abstract. Catastrophic overfitting (CO) poses a significant challenge to fast adversarial training (FastAT), particularly at large perturbation scales, leading to dramatic reductions in adversarial test accuracy. Our analysis of existing FastAT methods shows that CO is accompanied by abrupt and irregular fluctuations in loss convergence, indicating that a stable training dynamic is key to preventing CO. Therefore, we propose a training model that uses the Douglas-Rachford (DR) splitting technique to ensure a balanced and consistent training progression, effectively counteracting CO. The DR splitting technique, known for its ability to solve complex optimization problems, offering a distinct advantage over classical FastAT methods by providing a smoother loss convergence. This is achieved without resorting to complex regularization or incurring the computational costs associated with double backpropagation, presenting an efficient solution to enhance adversarial robustness. Our comprehensive evaluation conducted across standard datasets, demonstrates that our DR splitting-based model not only improves adversarial robustness but also achieves this with remarkable efficiency compared to various FastAT methods. This efficiency is particularly observed under conditions involving long training schedules and large adversarial perturbations.

Keywords: Adversarial training · Adversarial robustness · Convergence

1 Introduction

Deep neural networks (DNNs), despite their successes, are vulnerable to adversarial attacks, where small, undetectable, input perturbations can mislead models into incorrect predictions [11]. Given the security risks of such vulnerabilities, improving neural network robustness against these attacks has become a critical area of research [28, 38]. Adversarial training (AT) [18, 30] and its derivative, PGD-AT [22, 31], have emerged as effective strategies for enhancing the robustness of neural networks. However, the complex inner optimizations required by these methods at each training step significantly elevate computational costs, limiting their practicality for large-scale applications [28]. To address these computational challenges, 'fast' versions of AT (FastAT) [25, 36] have been proposed, which employ a single-step gradient propagation [11] to produce adversarial training examples. While FastAT offers a more efficient training process

* Correspondence to pshams55@gmail.com. Code: <https://github.com/mzarea2/RAT>

compared to PGD-AT, it introduces the problem of **Catastrophic Overfitting** (CO) [16,36]. CO occurs when training with strong adversaries, leading to a significant drop in the robustness of the resulting model. Several FastAT methods have been developed to reduce CO [42]. Wong et al. [36] enhance adversarial example diversity through the use of initial perturbations that are randomly initialized. Following this approach, [3] proposed FAST-GA, an efficient regularizer that aims to explicitly align gradients between benign and adversarial inputs. Likewise, MEP-PGI [15] and NuAT [33] employ nuclear or p-norm regularization within adversarial training to improve consistency in predictions on both benign and adversarial examples. These approaches are effective only within a limited perturbation $\mathcal{E} \leq 8/255$, where $\mathcal{E}$ denotes the degree of perturbations in the adversarial training [42]. Moreover, models trained with relatively small perturbation scales show vulnerability to larger $\mathcal{E}$, with noted declines in robustness.

For example, models trained with NuAT and MEP-PGI at a perturbation size of $\mathcal{E} = 8/255$ demonstrated robustness levels of 56% and 55% against PGD-50 attacks, but their robustness significantly dropped to 24% and 22%, respectively, with $\mathcal{E} = 16/255$. Our investigation into adversarial training processes (illustrated in Fig. 2) reveals that CO is characterized by minor fluctuations in benign classification loss and significant decreases in adversarial example loss. This observation raises a question: How can we design fast adversarial training that ensures improved stability and mitigates CO, while also optimizing the trade-off between robustness accuracy (RA) and standard accuracy (SA)?

In this context, the Douglas-Rachford (DR) splitting technique emerges as a promising solution. The DR splitting is known for its linear convergence across both strongly convex and non-convex settings [4, 20, 23, 26]. Several studies demonstrate its applicability in a variety of optimization tasks. For example, Liang et al. [19] and Applegate et al. [4] have highlighted its capability for local and global linear convergence, underscoring the DR technique’s effectiveness in complex optimization scenarios. We also capitalize on the inherent stability and efficiency of the DR splitting, incorporating these properties into adversarial training to address CO. This integration allows for a more stable and consistent adversarial learning process, significantly lowering the risk of CO. Indeed, our implementation demonstrates that DR splitting not only stabilizes the training process but also offers a cost-effective solution to enhance model robustness against adversarial attacks, as shown in Fig. 1. Our **contributions** are three-fold: ① We provide a comprehensive analysis showing that existing FastAT methods are still prone to CO with larger $\mathcal{E}$. Our investigation emphasizes the critical role of loss function convergence stability in addressing this vulnerability. ② Introducing a novel solution with the Douglas-Rachford splitting technique, specifically designed to mitigate CO without incurring computational complexities. ③ Conduct extensive experiments on standard datasets to validate the effectiveness of our model- **R**ethinking fast **A**dversarial **T**raining (RAT)- particularly against large adversarial perturbations. RAT, leveraging the DR-splitting technique, shows enhanced adversarial robustness when compared to classical adversarial training methods (find the training code in the supplementary file).

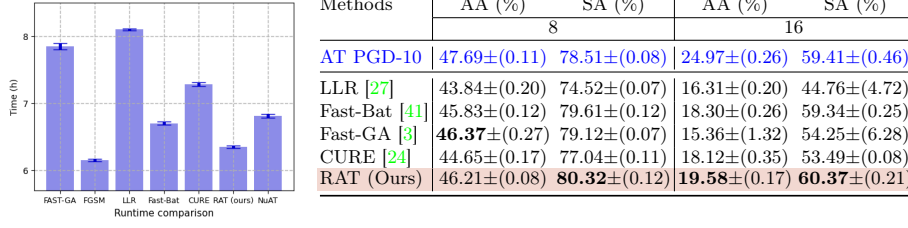


Fig. 1: Performance comparison of adversarial robustness with WideResNet34-10 on CIFAR10 for $\mathcal{E} = (8, 16)/255$. (left) Illustrates the average runtime for each method. (right) We compare AA (AutoAttack) and SA accuracies. Lower AA indicates higher vulnerability. Higher SA (standard accuracy) indicates better general performance.

2 Related Work

2.1 Adversarial Attacks

Adversarial attacks aim to fool deep learning models by introducing perturbed inputs. Goodfellow et al. [11] introduced the concept of the fast gradient sign method (FGSM) for generating adversarial examples x' within classification tasks. They demonstrated that adversarial examples, created through a single-step gradient update, can lead a classification model to incorrect predictions with significant confidence. The adversarial examples x' are generated by

$$x' = x + \mathcal{E} \cdot \text{sgn}(\nabla_x L(f(x; \theta), y)), \quad (1)$$

where (x, y) denotes an input sample and its corresponding label, $\mathcal{E}$ is the perturbation magnitude, $\nabla_x L$ is the loss gradient with respect to x , and θ signifies the fixed model weights. Subsequent research has developed more sophisticated attack methods, such as I-FGSM [17], MIM [9], and PGD [22], which leverage iterative gradient descent to enhance the efficacy of the attacks. The PGD attack, in particular, generates adversarial examples by applying small perturbations to the input data, ensuring these adjustments stay within a pre-defined boundary $\mathcal{E}$. This process is described as follows

$$x'_{t+1} = \Pi_{\mathcal{E}}(x'_t + \alpha \cdot \text{sgn}(\nabla_x L(f(x'_t; \theta), y))), \quad (2)$$

in which x' is the adversarial example at iteration t , and $\Pi_{\mathcal{E}}$ denotes the projection operation that ensures the adversarial example remains within the $\mathcal{E}$ -ball. The PGD attack is one of the most effective attack methods for evaluating the robustness of deep learning against adversarial examples.

2.2 Adversarial Training (AT)

Adversarial Training (AT) [22] and TRADES [40] have become prominent training methods for training robust architectures, which can be written as

$$\min_{\theta} E_{(x,y) \sim D} \left[\max_{\delta \in [-\mathcal{E}, \mathcal{E}]} L(f(x + \delta; \theta), y) \right], \quad (3)$$

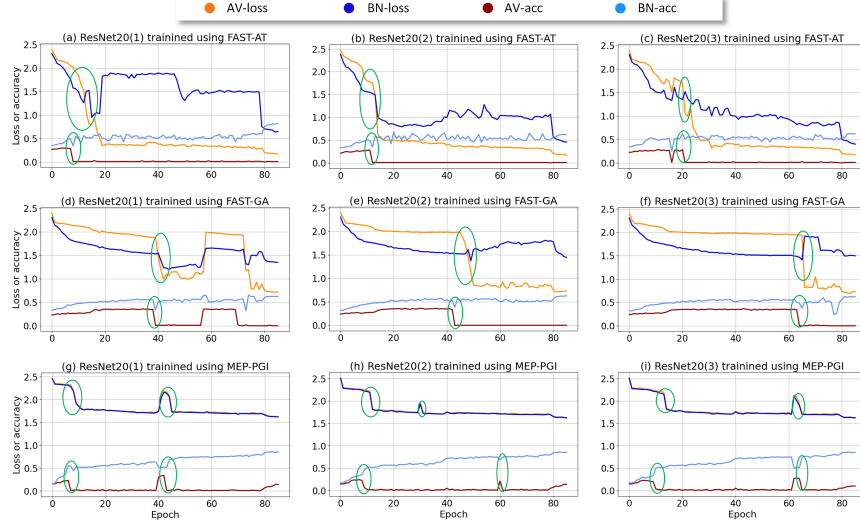


Fig. 2: Performance of fast adversarial training (FastAT) methods on CIFAR10 using ResNet20, for $\mathcal{E} = 16/255$. The figure illustrates the training process across three runs. The terms AV-loss and BN-loss are the training loss for adversarial and benign inputs, respectively, while AV-acc and BN-acc quantify testing accuracy against both adversarial and benign samples. The occurrence of catastrophic overfitting is marked (in green), indicating critical periods where adversarial robustness drops.

where δ is the adversarial perturbations and $x' = x + \delta$ is the adversarial examples generated by various attacks such as PGD and FGSM. This optimization first, generates adversarial examples within a defined perturbation limit to maximize classification loss (inner maximization) and then adjusts the model parameters θ to minimize this loss over the adversarial examples (outer minimization). However, a notable trade-off emerges in AT between computational efficiency and adversarial robustness, due to the complexity of solving the inner maximization problem. To enhance the computational efficiency of AT, several methods have been developed. Free AT [30], simultaneously updating the network parameters θ and the adversarial perturbation δ multiple times within a single batch. On the other hand, Fast-AT [36], simplifies the process further by employing a single-step FGSM attack [11] but the attack initiates from a randomly chosen point $x + \mu$ with $\mu \in [-\delta, \delta]$. Fast-Bat These approaches reduce the cost of solving the inner maximization problem, significantly accelerating the training process.

2.3 Catastrophic Overfitting (CO)

The acceleration of AT methods comes with significant drawbacks. Wong et al. [36] first reported this with FGSM training, and was later supported by Andriushchenko et al. [3] in their analysis of Free AT and Fast-AT methods. In these cases, the model’s accuracy on adversarial tests can fall sharply, sometimes

to 0%, in just a few training epochs. This decline in performance has been attributed to CO, a critical challenge that has motivated significant research efforts aimed at understanding and mitigating its impact [14, 18, 41]. Below, we provide an overview of various strategies developed to avoid CO.

2.4 Regularization-based Approaches

Recent efforts focus on regularization and data augmentation to prevent CO. [3] introduced FAST-GA to mitigate CO by penalizing the misalignment of gradients between the input and their adversarial counterparts. This method ensures that the direction of the gradient remains consistent when slight perturbations are introduced to the input. Qin et al. [27] introduced local linear regularization (LLR) to improve model robustness against adversarial attacks. LLR smooths the loss landscape using a first-order Taylor approximation, addressing non-linear sensitivities and promoting a more linear loss response to perturbations. ELLE [28] introduced a regularization term designed to mitigate CO both effectively and efficiently within classical AT frameworks, by enforcing local linearity. Sriramanan et al. [32] introduced GAT, a novel regularization strategy aimed at reducing the difference between the logits of a training point and a random sample. NuAT [33], proposed a regularization term that penalizes the nuclear norm of the logits difference between benign and adversarial examples. However, GAT and NuAT still face challenges with CO under larger perturbation size $\mathcal{E}$ [16].

2.5 Other Approaches

Park et al. [25] introduced SLAT which performs adversarial attacks within the latent space, focusing on the internal representations rather than the input space directly. Another study [35] employs the Frank-Wolfe optimization technique for AT, connecting the distortion observed in the loss landscape to the norm of adversarial perturbations, providing a deeper understanding of adversarial robustness. Li et al. [18] addressed CO by limiting network weights to a specific subspace based on the main components of AT-trained weights. Meanwhile, Zhang et al. [41] introduce Fast-Bat, which redefines the AT challenge as a Bilevel optimization problem and introduces a one-step solution to address it. However, [28] shows that Fast-BAT is affected by CO for $\mathcal{E} > 10/255$. Zero-Grad [10] introduces a strategy to mitigate adversarial perturbations by setting to zero the gradient components that fall below a threshold, however, [16] demonstrates, this approach results in decreased performance for large $\mathcal{E}$. Different from these methods, we aim to minimize the loss fluctuations during training through the Douglas-Rachford splitting technique, known for its effectiveness in complex optimization [23]. It has been successfully applied to optimal transport [20] and its discretizations [6], demonstrating its flexibility and robustness in achieving linear convergence rates under both convex and non-convex settings [4, 34]. A notable advantage of this splitting technique is to leverage sparsity in solutions, which can significantly accelerate the convergence process [26].

3 Method

This section evaluates FastAT methods under large perturbation budgets, $\mathcal{E}$, explores the causes of CO, and finally introduces our proposed solutions.

3.1 Performance of Fast Adversarial Training on a Large $\mathcal{E}$

Fast AT methods have shown effectiveness in mitigating CO at lower perturbation ($\mathcal{E} \leq 8/255$) [28, 36]. To extend this evaluation, we analyze performance at a higher perturbation level ($\mathcal{E} = 16/255$), in line with [42]. **FAST-AT** [36] introduces an initial step for Eq. (3), where samples are perturbed with uniform random noise $\delta_0 \sim U(-\mathcal{E}, \mathcal{E})$ before applying the FGSM attack,

$$\min_{\theta} E_{(x,y) \sim D} [\max_{\delta_0 + \delta \in [-\mathcal{E}, \mathcal{E}]} L(f(x + \delta_0 + \delta; \theta), y)]. \quad (4)$$

Here, $L(\cdot)$ is the loss function, δ represents adversarial perturbations, and θ is the model weights. **FAST-GA**, as introduced by [3] focuses on aligning the gradients of the loss with respect to both benign samples x and their perturbed samples $x + \delta_0$, which is described as,

$$E_{(x,y) \sim D} [1 - \cos(\nabla_x L(x, \theta), \nabla_{x+\delta_0} L(x + \delta_0, \theta))], \quad (5)$$

in which $L(x, \theta) = L(f(x; \theta), y)$. The cosine similarity $\cos(\cdot)$ measures the cosine similarity between the gradients of the loss function for the benign and perturbed inputs. This strategy aims to ensure that the model’s behavior does not change drastically for adversarial examples, thereby reducing CO. Different from other attack methods, **MEP-PGI** [15] extends the FGSM attack by incorporating a Prior-Guided Initialization (PGI) mechanism. This technique initializes the adversarial perturbations δ_0 , by using the information from adversarial examples generated in previous training epochs,

$$E_{(x,y) \sim D} [\|f(x'; \theta) - f(x + \delta_0; \theta)\|_2^2], \quad (6)$$

here, $\|\cdot\|$ is the squared (L2) norm, measuring the distance between the model’s outputs for the two types of examples. FAST-PGI aims to enrich the adversarial examples with historical insight, thereby enhancing the adversarial training process. Fig. 2 illustrates the adversarial training of ResNet20 on CIFAR10, highlighting times when CO occurs, between epochs 5th-20th, 40th-65th, and 10th-15th. The key observations from the results are i) transition from smooth/stable to erratic loss fluctuations; ii) discrepancies in the loss for benign and adversarial examples, i.e., minor fluctuations in benign loss $L(x, \theta)$ followed with a rapid decline in adversarial loss $L(x', \theta)$; and iii) a sharp decrease in adversarial accuracy. The result is consistent with the observations in [1, 28, 42]. From these patterns, we derive several insights for understanding CO: it is obvious that $L(x, \theta)$ has greater stability than the loss $L(x', \theta)$, indicating a tendency of models to overfit to adversarial samples. CO directly correlates with the instability in adversarial training convergence. Anomalies in both benign and adversarial loss occur together, signifying an issue in the learning process that affects its performance

on both data. We also find that Fast-GA and MEP-PGI enhance the stability of adversarial training through the constraints in Eq. (4) and Eq. (5), compared to the Fast-AT. We are now in shape to give a formal definition of our model.

3.2 Douglas-Rachford (DR) Splitting for Adversarial Training

We use the DR splitting technique to ensure a consistent and stable optimization to prevent sudden changes in adversarial robustness. This technique divides the optimization process into a two-step process (reflection and enforcement), which together ensure a stable and consistent optimization path. Let's denote the model at iteration k by $f(\cdot, \theta^{(k)})$, where $\theta^{(k)}$ represents the model parameters. The loss function for adversarial examples is $L(f(x'^{(k)}, \theta^k), y)$, and for benign examples, $L(f(x^k, \theta^k), y)$. An important property of our model is its ability to alternate between reducing adversarial loss and applying careful regularization terms. This ensures that any adjustments to the model parameters effectively impact both benign and adversarial loss functions $L(x, \theta)$ and $L(x', \theta)$. Specifically, to address CO, we introduce epoch-based constraints on the loss differences for both adversarial and benign instances, represented as $(\Delta L_{\theta^{(k)}}, \Delta L_{\theta^{(k)}}^{benign})$,

$$\begin{aligned}\Delta L_{\theta^{(k)}} &= |L(f(x'^{(k)}, \theta^{(k)}), y) - L(f(x'^{(k-1)}, \theta^{(k-1)}), y)| \\ \Delta L_{\theta^{(k)}}^{benign} &= |L(f(x, \theta^{(k)}), y) - L(f(x, \theta^{(k-1)}), y)|,\end{aligned}\quad (7)$$

where $|\cdot|$ measures the absolute difference in loss for adversarial (x') and benign (x) inputs, $x'^{(k)} = x + \delta_{0,k} + \delta_k$. Here, $\delta_{0,k}$ can be replaced by various attack initialization perturbations and δ_k is generated by $[\max_{\delta_k \in [-\varepsilon, \varepsilon]} L(x'^{(k)}, \theta^{(k-1)})]$. This formulation guides the model to effectively manage perturbations and maintain a stable training process. Additionally, these constraints help in mitigating the simultaneous anomalies in both $L(x, \theta)$ and $L(x', \theta)$ facilitating a more controlled update of the model parameters that considers both adversarial and benign inputs. In practice, this involves updating the model's intermediate parameters, C^k , to reduce the adversarial loss, according to the gradient of the loss function ∇_{θ} with respect to the adversarial examples $x'^{(k)}$,

$$C^k = \theta^{(k)} - \eta \nabla_{\theta} L(f(x'^{(k)}, \theta^{(k)}), y), \quad (8)$$

where η is a learning rate. This update, termed as the 'reflection step' (θ_{ref}), is pivotal in aligning the model's parameters towards reducing adversarial vulnerability. We then combine these updates C^k with regularization adjustments to stabilize the training dynamics and enhance the adversarial robustness, effectively preventing overfitting to adversarial examples while maintaining performance on benign inputs. The updated model parameters, $\theta^{(k+1)}$, allow to handle both adversarial and benign inputs effectively without compromising training stability or performance,

$$\theta^{(k+1)} = C^k + \lambda(\text{Reg}(\theta^{(k)}) - C^k), \quad (9)$$

where $\lambda \in [0, 1]$ controls the strength of the regularization, and Reg involves components in Eq. (7) such that

$$\text{Reg}(\theta^{(k)}) = \gamma_1(\Delta L_{\theta^{(k)}}) + \gamma_2(\Delta L_{\theta^{(k)}}^{benign}). \quad (10)$$

The important reason behind Eq. (9) is that the regularization term $Reg(\theta^{(k)})$ incorporates changes in loss for both adversarial and benign examples, weighted by γ_1 and γ_2 , respectively. This helps to balance the model’s performance across adversarial and benign inputs, addressing CO by preventing the model from becoming overly specialized on the adversarial examples. Because CO occurs when a model overfits to specific adversarial examples during training, compromising its ability to generalize [21]. Our formulation ensures that the model’s updates consider the impact on both adversarial robustness and benign accuracy. This regularization can be seen as part of the ‘enforcement step’ (θ_{enf}). The DR technique involves iteratively updating the model parameters by combining the adversarial loss minimization and the stability enforcement components, as $\theta = 0.5 \times (\theta_{ref} + \theta_{enf})$. θ_{ref} targets reducing the loss on adversarially perturbed inputs Eq. (8), conversely, θ_{enf} represents the parameters adjusted after applying the regularization term, focusing on stability and CO prevention Eq. (9). This approach under the DR-splitting technique provides a comprehensive strategy for addressing CO in the adversarial training process. Indeed, this allows the model to effectively adapt to adversarial perturbations while ensuring that such adaptations do not compromise the model’s ability to accurately classify benign inputs or lead to overfitting on the adversarial examples during training. Our proposed RAT is given in Algorithm 1, discussion in the supplementary file.

Efficient Computation of Loss Differences In practice, directly computing the losses $L(f(x'^{(k-1)}, \theta^{(k-1)}), y)$ and $L(f(x, \theta^{(k-1)}), y)$ in Eq. (7) can consume a significant amount of memory. To address this, we introduce u^{k-1} for benign inputs and u'^{k-1} for adversarial inputs, replacements for these terms, allowing for efficient computation and tracking of the average loss over benign and adversarial examples, respectively. Specially, $u^{k-1} = \frac{1}{N} \sum_i L(f(x'_i, \theta^{(k-1)}), y_i)$ represents the average loss for adversarial inputs, and $u'^{k-1} = \frac{1}{N} \sum_{i=1}^N L(f(x_i, \theta^{(k-1)}), y_i)$ denotes the average loss for benign inputs, where N is the batch size. This can reduce computational and memory overhead by focusing on the aggregate loss across batches, rather than calculating them individually for each sample.

4 Experimental Results

4.1 Experimental settings

In this section, we evaluate the proposed model across four benchmark datasets to determine its effectiveness and robustness against adversarial attacks. For both CIFAR10 [8] and CIFAR100 [8], we use the ResNet20 [12] architecture as the backbone, recognizing its strong performance in image classification tasks. For the more complex Tiny-ImageNet and ImageNet [8], we use the PreActResNet18 [13] to better handle its challenges. Additionally, the use of WideResNet34-10 [39] highlights our model’s capacity to maintain effectiveness on long training schedules. We use the SGD optimizer with a batch size of 128, a weight decay of 5.10^{-4} , and a momentum of 0.9. We initiate training with learning rates of 0.1 for CIFAR10 and CIFAR100. For Tiny ImageNet, due to its greater complexity,

Algorithm 1 Rethinking fast Adversarial Training (RAT) using DR Splitting

Require: Training dataset $(x_i, y_i)_{i=1}^N$, # epochs K, # batch B, perturbation budget $\mathcal{E}$, $\lambda, \gamma_1, \gamma_2$

- 1: Initialize model parameters θ_0
- 2: **for** epoch = 1 **to** K **do**
- 3: **for** $i = 1$ **to** B **do**
- 4: $x'_i = x_i + \delta \text{sign}(\nabla_{x_i} L(f(x_i; \theta_{\text{epoch}-1}), y_i))$ ▷ Adversarial example generation
- 5: $L_{\text{adv}} = L(f(x'_i; \theta_{\text{epoch}-1}), y_i)$
- 6: $L_{\text{benign}} = L(f(x_i; \theta_{\text{epoch}-1}), y_i)$
- 7: $\theta_{\text{reflect}} = \theta_{\text{epoch}-1} - \eta \cdot \nabla_{\theta} L_{\text{adv}}$ ▷ Reflection step
- 8: $\Delta L_{\text{adv}} = |L(f(x'_i; \theta_{\text{reflect}}), y_i) - L_{\text{adv}}|$
- 9: $\Delta L_{\text{benign}} = |L(f(x_i; \theta_{\text{reflect}}), y_i) - L_{\text{benign}}|$
- 10: $\text{Reg} = \gamma_1 \cdot \Delta L_{\text{adv}} + \gamma_2 \cdot \Delta L_{\text{benign}}$ ▷ Enforcement step
- 11: $\theta_{\text{enforce}} = \theta_{\text{reflect}} + \lambda \cdot (\text{Reg} - \theta_{\text{reflect}})$
- 12: $\theta_{\text{epoch}} = 0.5 \cdot (\theta_{\text{reflect}} + \theta_{\text{enforce}})$ ▷ DR splitting
- 13: **end for**
- 14: **end for**

we start at a lower rate of 0.01. The training spans 110 epochs, with a strategy to decrease the learning rate by a factor of 0.1 at both the 100th and 105th epochs for optimization. To prepare our model against adversarial attacks, we incorporate FAST-AT [36] and MEP-PGI [15] as our initial attack strategies for CIFAR datasets. Given the significant memory demands of MEP-PGI, particularly its need to store previous adversarial perturbations, we use its variation PGI-BP on Tiny ImageNet. We set the step-size parameter γ to 0.01 for CIFAR datasets, and 0.001 for Tiny ImageNet. The regularization term $\lambda=0.3$ for all datasets, and $\mathcal{E} = 16/255$. Experiments are run using a single GeForce RTX 3090 GPU. Find the training code and additional results in the *supplementary* file.

Method	SA (%)	RA-PGD-10 $\mathcal{E}=8/255$	RA-PGD-10 $\mathcal{E}=10/255$	RA-PGD-10 $\mathcal{E}=12/255$	RA-PGD-10 $\mathcal{E}=16/255$	SQ [2]	Time (h)
AWP [37]	77.94	36.26	33.18	30.25	27.43	19.76	2.4
OAAT [1]	70.51	45.32	39.47	34.10	28.91	21.43	3.2
ExAT [29]	79.33	49.56	43.28	36.49	32.58	23.15	1.2
RAT-MEP	69.86	52.64	46.83	44.18	41.26	27.13	1.7

Table 1: Comparative performance across different perturbation levels on CIFAR10 using ResNet20 backbone. Our evaluation focuses on two key metrics: robust accuracy (RA) against PGD-10 attacks, and the computational time (measured in hours).

Baselines and Evaluation Metrics: To thoroughly evaluate the adversarial robustness of our model, we use a comprehensive set of attack methods. Following [36], we use FGSM [11], and a series of PGD attacks [22] with varying iterations: PGD-10, PGD-20 and PGD-50. Additionally, we use the APGD-CE [7], C&W attack [5], and Autoattack (AA) [7]. AutoAttack is an ensemble method

Methods	SA	RA-FGSM	RA-PGD10	RA-PGD20	RA-PGD50	RA-C&W	RA-APGD-CE	RA-AA	Time (h)
AT PGD	<u>65.72</u>	<u>46.51</u>	<u>40.87</u>	<u>35.42</u>	<u>34.21</u>	<u>30.97</u>	<u>33.68</u>	<u>26.85</u>	<u>6.3</u>
Fast-GA [3]	58.49	39.88	33.31	<u>26.89</u>	<u>25.07</u>	22.81	<u>24.33</u>	17.56	2.4
ZeroGrad [10]	72.81	44.26	32.78	22.15	19.23	20.85	17.20	12.75	1.3
NuAT [33]	73.55	<u>44.90</u>	<u>35.52</u>	26.34	24.31	<u>24.95</u>	23.72	<u>18.93</u>	1.8
Fast-AT [36]	50.64	38.62	26.47	21.64	20.55	19.26	19.81	14.97	1.2
RAT-AT	70.62	46.28	38.12	30.83	29.35	26.54	27.79	19.64	1.3
MEP-PGI [15]	54.23	36.84	32.65	27.98	26.70	22.65	26.73	19.31	1.5
RAT-MEP	69.86	47.69	41.26	34.81	33.64	28.75	32.87	23.91	1.7

Table 2: CIFAR10 results: We report the RA (%) against various attacks at $\mathcal{E}=16/255$ with ResNet20 backbone. Each model undergoes 3 training sessions. The results reflect the average performance of the top-performing model across these iterations. The best and the runner-up models are bolded and underlined, respectively.

that integrates APGD-CE, APGD-T (targeted APGD), and Square Attack [2], to thoroughly evaluate a model’s robustness to adversarial attacks. Due to the inherent variability in the training process, which often leads to fluctuations in performance, relying on a single run may not accurately capture the effectiveness of different methods. Thus, for each method, we conduct the training process three times and report the average of the best results in these runs. The evaluation metrics include robust accuracy (RA) and standard accuracy (SA).

4.2 Results and Insights

In our preliminary analysis, we highlight the importance of our results on the adversarial robustness of trained models. Our findings indicate that adversarial attacks with $\mathcal{E} = 16/255$, rarely result in the misclassification of the input. This observation is achieved through the generation of adversarial examples across CIFAR10 and CIFAR100 using both targeted and non-targeted attack methods, including FGSM, APGD-CE, APGD-T, PGD, and C&W. Remarkably, most of these adversarial examples still were classified correctly according to their original labels. Our experiments in Tab. 1 involve generating adversarial training with perturbation size $\mathcal{E} = 16/255$ and testing the models’ robustness across different levels of perturbation. We also compared our model (RAT) with other techniques, such as Fast-Bat, AWP, OAAT, and ExAT. We have reimplemented these AT models within the FastAT framework to ensure a fair comparison. Our model achieves superior performance even as the perturbation magnitude increases to $\mathcal{E} = 16/255$. Furthermore, our model integrates into FAST-AT and MEP-PGI, which are specifically designed to address distributions of adversarial samples. Conversely, methods like OAAT are optimized for the distribution of benign examples and yield higher SA accuracy. Adopting the settings from OAAT, we apply both OAAT and our variation OAAT_{RAT} to FastAT tasks. The OAAT_{RAT} achieves +2.78% improvement on SA accuracy over OAAT (70.51%→73.29), and 24.66% on the SQ, as (21.43%→24.66%).

Methods	$\mathcal{E}$	SA	RA-FGSM	RA-PGD10	RA-PGD20	RA-PGD50	RA-C&W	RA-APGD-CE	RA-AA	Time (h)
Fast-GA [36]	12	67.39	43.57	32.19	28.02	26.24	27.65	26.21	21.75	2.3
NuAT [33]		<u>72.84</u>	52.33	41.95	<u>38.84</u>	<u>37.69</u>	<u>36.08</u>	<u>36.75</u>	<u>32.41</u>	1.8
MEP-PGI [15]		73.78	52.39	<u>38.67</u>	36.52	35.78	33.41	33.68	27.93	1.5
RAT-MEP		72.81	54.62	45.29	42.86	42.23	36.88	41.60	33.16	1.7
Fast-GA [3]	10	<u>81.70</u>	55.73	36.80	30.71	28.65	31.29	26.91	24.12	2.3
NuAT [33]		76.39	56.09	<u>46.23</u>	<u>43.95</u>	<u>43.50</u>	41.52	<u>43.36</u>	<u>38.99</u>	1.8
MEP-PGI [15]		81.94	59.48	43.18	39.76	37.83	37.46	36.21	32.72	1.5
RAT-MEP		76.52	<u>58.30</u>	47.51	45.74	45.56	<u>41.27</u>	45.36	39.70	1.7

Table 3: Quantitative comparison of FastAT methods on CIFAR10 using ResNet20 for different ($\mathcal{E}$), where both training and testing used the same $\mathcal{E}$. The best and the runner-up methods are bolded and underlined, respectively.

Methods	SA	RA-FGSM	RA-PGD10	RA-PGD20	RA-PGD50	RA-C&W	RA-APGD-CE	RA-AA	Time (h)
AT PGD	<u>40.83</u>	<u>25.19</u>	<u>21.38</u>	<u>17.59</u>	<u>17.23</u>	<u>15.06</u>	<u>17.10</u>	<u>12.54</u>	6.3
Fast-GA [3]	32.40	15.84	12.98	10.79	10.08	8.54	9.72	6.84	2.4
ZeroGrad [10]	47.81	<u>24.13</u>	<u>18.21</u>	12.93	12.24	<u>11.70</u>	11.21	7.90	1.2
NuAT [33]	32.49	21.27	16.75	<u>14.38</u>	<u>13.71</u>	11.52	<u>12.62</u>	<u>8.75</u>	1.8
Fast-AT [36]	31.22	15.69	13.41	10.78	10.25	8.73	9.88	7.21	1.2
RAT-AT (ours)	42.26	25.81	19.43	15.77	14.56	12.81	13.95	9.86	1.3
MEP-PGI [15]	22.41	14.09	12.25	9.86	9.61	8.15	9.53	6.86	1.5
RAT-MEP (ours)	43.77	25.26	21.35	17.42	17.03	14.22	16.76	11.49	1.7

Table 4: CIFAR100 results: We report the RA (%) against various attacks at $\mathcal{E}=16/255$ with ResNet20 backbone. Each method undergoes three training sessions. The results reflect the average performance of the top-performing model across these iterations. The best and the runner-up methods are bolded and underlined, respectively.

4.3 Faster DR Splitting Technique

Fig. 1-right provides a detailed comparison of various adversarial training on the CIFAR10 dataset, at $\mathcal{E} = (8, 16)/255$. At the lower perturbation level ($\mathcal{E} = 8/255$), all methods demonstrate similar AA and SA, with FAST-GA achieving the highest AA among them. However, increasing the perturbation to $\mathcal{E} = 16/255$ results in a significant reduction in AA across all methods, highlighting the challenges of maintaining robustness at higher perturbation levels. Interestingly, our model outperforms Fast-Bat at $\mathcal{E} = 16/255$, achieving 60.37% vs. 59.34% in SA accuracy. This improvement is also evident at $\mathcal{E} = 8/255$, where RAT aligns closely with the performance of other leading methods, e.g., 46.21% AA accuracy, comparable to 46.37%, 45.83% for FAST-GA and Fast-Bat respectively. While FGSM is faster (as shown in Fig. 1-left), results in 0% AA accuracy. We also observe that our model adds very little overhead to FGSM training, i.e., 15.02 min with FGSM, but is considerably faster than other methods. The increased computation time in LLR and CURE is due to Double Backpropagation [28].

4.4 CIFAR10 Results

In this experiment, we use backbone ResNet20, to evaluate our model’s performance, setting different perturbation levels $\mathcal{E}$. The results, in Tab.2, show that

Methods	SA	RA-FGSM	RA-PGD10	RA-PGD20	RA-PGD50	RA-C&W	RA-APGD-CE	RA-AA	Time (h)
AT PGD	33.79	17.66	13.82	10.56	10.28	8.36	10.45	6.47	68.1
Fast-AT [36]	28.12	13.06	10.21	7.36	6.95	5.48	6.72	4.51	10.8
Fast-GA [3]	29.24	13.98	11.32	8.45	8.16	6.23	7.40	4.96	21.4
ZeroGrad [10]	34.58	12.85	9.41	6.73	5.71	4.22	5.26	2.81	10.9
NuAT [33]	35.20	<u>16.71</u>	<u>12.48</u>	<u>9.52</u>	<u>9.37</u>	<u>7.62</u>	<u>8.21</u>	<u>5.39</u>	25.2
PGI-BP [15]	21.76	11.32	9.48	7.56	7.20	5.49	7.21	4.18	13.8
RAT-BP (ours)	34.61	17.62	13.23	9.91	9.45	8.20	9.87	6.25	14.2

Table 5: Tiny-ImageNet results: We report the RA (%) against various attack methods accuracy at $\mathcal{E} = 16/255$ using PreActResNet18 as our backbone. The results represent the average performance of the best models across three training iterations.

our model outperforms existing Fast AT techniques in adversarial robustness across various attacks. Specifically, our RAT-AT surpasses all FastAT methods, and our RAT-MEP is more effective than other methods while maintaining comparable SA accuracy. In comparison to PGD-AT, RAT-MEP achieves nearly equivalent adversarial robustness, with a marginal of 0.57 difference in performance against PGD-50 (33.64% *vs.* 34.21%). Regarding computational efficiency, RAT-AT is faster, completing in 75 minutes, compared to 144 minutes for Fast-GA and 108 minutes for NuAT. RAT-MEP also presents a good balance between computational time (102 minutes) and performance, taking only slightly longer than MEP-PGI (96 minutes) but significantly less than the 380 minutes required by PGD-AT. In Tab. 3, we conduct additional experiments on different perturbation scales $\mathcal{E} = (10, 12)/55$, further validating the model’s effectiveness. The results clearly demonstrate that our method consistently achieves superior adversarial robustness across different perturbation budgets.

4.5 CIFAR100 Results

Tab. 4 details the performance on CIFAR100 with a ResNet20 backbone, highlighting that the training times for FastAT methods across CIFAR10 and CIFAR100 are similar. As can be seen, our model demonstrates the ability to maintain stable adversarial training, and among the FastAT methods, RAT-AT is the most robust against different attacks. RAT-PGI surpasses the performance of all FastAT methods in terms of adversarial robustness. While RAT-MEP is slightly lower than AT-PGD, it boasts a training speed that is approximately 3.5 times faster, marking a significant improvement in efficiency.

4.6 ImageNet Results

Our experiments with the PreActResNet18 on Tiny-ImageNet, demonstrate the effectiveness of our method in handling larger and more complex dataset, as presented in Tab. 5. The RAT-BP shows comparable robustness to PGD-AT and surpasses other FastAT methods. In terms of training efficiency, RAT-BP is marginally higher, requiring 14.2 hours, compared to PGI-BP (13.8 hours), yet it is significantly faster than PGD-AT, which takes 68 hours. Further extending

Methods	SA	RA-AA	RA-FGSM	RA-PGD10	RA-PGD50	Time (h)
Fast-AT [36]	7.41	1.27	3.42	3.76	2.25	61.5
PGI-BP [15]	16.21	1.05	9.48	2.81	1.85	77.6
RAT-BP (ours)	25.60	1.88	14.61	8.70	3.29	79.2

Table 6: Quantitative results of the adversarial training methods on ImageNet at ($\mathcal{E} = 16/255$) with ResNet50 as the backbone. Adding our model (RAT) helps PGI-BP overcome CO and improve its performance.

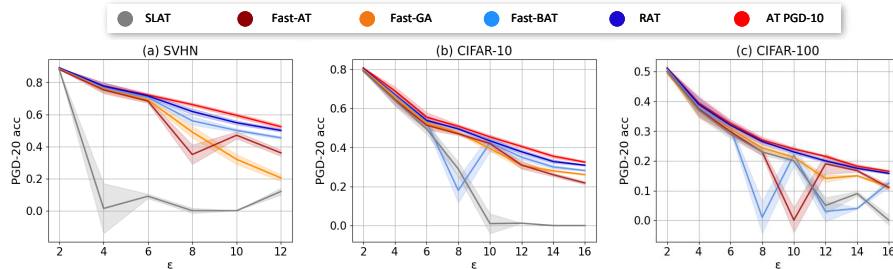


Fig. 3: Catastrophic Overfitting: Illustrates the robustness across perturbation strengths. The x-axis ($\mathcal{E}$) represents the strength of the perturbation applied to test images to create adversarial examples. The y-axis is the accuracy of the model under attack (PGD-20). As $\mathcal{E}$ increases, we expect accuracy to decrease, since stronger perturbations are harder to withstand. The AT PGD-10 is referenced for comparison, showing that our model achieves performance closely aligned with this benchmark.

our evaluation to the ImageNet [8], we use a ResNet50 following [36] training configuration with the only difference by the adoption of a maximum image resolution of 288×288 , similar to [16]. The initial learning rate is 0.1, with a reduction factor of 0.1 applied after 30 and 60 epochs, across a total of 90 epochs. The results in Tab. 6 show the stability of our training across this dataset.

4.7 Ablation

Our ablation study focuses on evaluating the effectiveness of our model against leading adversarial training methods, such as Fast-GA, Fast-AT, SLAT, and Fast-Bat. We also reference the robust but computationally intensive AT PGD-10. PreActResNet18 is a backbone model for these experiments, with each method’s recommended hyperparameters applied. We calculate the PGD-20 test accuracy, averaging results over three random seeds to ensure reliability. Fig. 3, illustrates that SLAT encounters CO at perturbation levels of $\mathcal{E} \geq 4/255$ for SVHN and $\mathcal{E} \geq 10/255$ for CIFAR10. Fast-GA shows less than optimal performance at $\mathcal{E} \geq 16/255$ for CIFAR10 and inconsistent results on CIFAR100. Notably, Fast-Bat also experiences CO for $\mathcal{E} > 10/255$ in CIFAR100. Overall, our RAT model significantly narrows the gap between single-step and multi-step AT methods, showing enhanced robustness and efficiency. **Sensitivity to λ :** To optimize our model, we focus on determining the optimal value for the parameter $\mathcal{E}$ in Eq.(9)

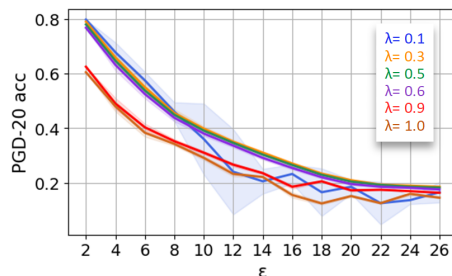


Fig. 4: Exploration of λ values for RAT on CIFAR10, using PRN backbone. This illustrates the relationship between the choice of λ and the occurrence of CO. Higher values successfully prevent CO but lead to a noticeable reduction in model performance.

across varying levels of adversarial perturbation. We evaluate the PGD-20 adversarial accuracy on a validation set of 1024 images from the CIFAR10 dataset. The experiments are conducted using the PRN architecture, exploring a range of $\lambda \in \{0.1, 0.3, 0.5, 0.6, 0.9, 1\}$. Except for λ , all other parameters are maintained as per the default configuration. The objective is to identify the λ that yields the highest adversarial accuracy for each $\mathcal{E}$, as shown in Fig. 4. Our results indicate a clear trend: lower λ and higher $\mathcal{E}$ values, fail to prevent CO. Increasing λ to 0.9 eliminates CO but degrades model performance. This effect is particularly ev-

ident at $\lambda = 1$, where, the PGD-20 accuracy decreases for lower and higher perturbation values. For all datasets, the model performance is almost consistent across $\lambda \in \{0.3, 0.5, 0.6\}$, leading us to select the lowest value within this range for efficiency. On the ImageNet dataset, however, a higher value (0.7) enhances performance. To further assess the impact of hyperparameters γ_1 and γ_2 on model performance and adversarial robustness, we conducted experiments by varying their values and analyzing the outcomes on CIFAR100. The results are presented in Table 2 (supplementary file). The values are chosen from a set $\{0.0, 0.3, 0.5, 0.7, 0.8, 1.0\}$, and the model’s performance was evaluated against several adversarial attacks, including FGSM, PGD-10, PGD-[20, 50], and AA.

5 Conclusion

We presented an efficient model to overcome catastrophic overfitting in fast adversarial training through the Douglas-Rachford (DR) splitting technique, emphasizing linearity and stability in the training process. By using this technique, we provide a novel perspective on managing adversarial robustness that balances between minimizing adversarial loss and enforcing effective regularization. Indeed, using DR splitting allows for subtle control over the training process, facilitating adjustments in model parameters to prevent sudden degradation in adversarial robustness. This methodology not only addresses CO with greater efficiency than existing solutions like Fast-GA, Fast-AT, and NuAT but also enhances the overall training stability across a wide range of adversarial perturbations and training durations. Through extensive evaluations, with significant adversarial challenges and in conjunction with various Fast AT methods, our model demonstrates superior performance and adaptability.

References

1. Addepalli, S., Jain, S., Sriramanan, G., Venkatesh Babu, R.: Scaling adversarial training to large perturbation bounds. In: European Conference on Computer Vision. pp. 301–316 (2022) [6](#), [9](#)
2. Andriushchenko, M., Croce, F., Flammarion, N., Hein, M.: Square attack: a query-efficient black-box adversarial attack via random search. In: European conference on computer vision. pp. 484–501 (2020) [9](#), [10](#)
3. Andriushchenko, M., Flammarion, N.: Understanding and improving fast adversarial training. *Advances in Neural Information Processing Systems* **33**, 16048–16059 (2020) [2](#), [3](#), [4](#), [5](#), [6](#), [10](#), [11](#), [12](#)
4. Applegate, D., Hinder, O., Lu, H., Lubin, M.: Faster first-order primal-dual methods for linear programming using restarts and sharpness. *Mathematical Programming* **201**(1), 133–184 (2023) [2](#), [5](#)
5. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy (SP). pp. 39–57 (2017) [9](#)
6. Chambolle, A., Contreras, J.P.: Accelerated bregman primal-dual methods applied to optimal transport and wasserstein barycenter problems. *SIAM Journal on Mathematics of Data Science* **4**(4), 1369–1395 (2022) [5](#)
7. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: International conference on machine learning. pp. 2206–2216 (2020) [9](#)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009) [8](#), [13](#)
9. Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., Li, J.: Boosting adversarial attacks with momentum. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9185–9193 (2018) [3](#)
10. Golgooni, Z., Saberi, M., Eskandar, M., Rohban, M.H.: Zerograd: Mitigating and explaining catastrophic overfitting in fgsm adversarial training. *arXiv preprint arXiv:2103.15476* (2021) [5](#), [10](#), [11](#), [12](#)
11. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. *International Conference on Learning Representations* (2014) [1](#), [3](#), [4](#), [9](#)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016) [8](#)
13. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. pp. 630–645 (2016) [8](#)
14. He, Z., Li, T., Chen, S., Huang, X.: Investigating catastrophic overfitting in fast adversarial training: A self-fitting perspective. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2313–2320 (2023) [5](#)
15. Jia, X., Zhang, Y., Wei, X., Wu, B., Ma, K., Wang, J., Cao, X.: Prior-guided adversarial initialization for fast adversarial training. In: European Conference on Computer Vision. pp. 567–584 (2022) [2](#), [6](#), [9](#), [10](#), [11](#), [12](#), [13](#)
16. de Jorge Aranda, P., Bibi, A., Volpi, R., Sanyal, A., Torr, P., Rogez, G., Dokania, P.: Make some noise: Reliable and efficient single-step adversarial training. *Advances in Neural Information Processing Systems* **35**, 12881–12893 (2022) [2](#), [5](#), [13](#)
17. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236* (2016) [3](#)

18. Li, T., Wu, Y., Chen, S., Fang, K., Huang, X.: Subspace adversarial training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13409–13418 (2022) [1](#), [5](#)
19. Liang, J., Fadili, J., Peyré, G.: Local convergence properties of douglas–rachford and alternating direction method of multipliers. *Journal of Optimization Theory and Applications* **172**, 874–913 (2017) [2](#)
20. Lindbäck, J., Wang, Z., Johansson, M.: Bringing regularized optimal transport to lightspeed: a splitting method adapted for gpus. *Advances in Neural Information Processing Systems* **36** (2023) [2](#), [5](#)
21. Liu, X., Chakraborty, S., Sun, Y., Huang, F.: Rethinking adversarial policies: A generalized attack formulation and provable defense in rl. In: International Conference on Learning Representations (2024) [8](#)
22. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations* (2018) [1](#), [3](#), [9](#)
23. Mai, V.V., Lindbäck, J., Johansson, M.: A fast and accurate splitting method for optimal transport: Analysis and implementation. *International Conference on Learning Representations* (2022) [2](#), [5](#)
24. Moosavi-Dezfooli, S.M., Fawzi, A., Uesato, J., Frossard, P.: Robustness via curvature regularization, and vice versa. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9078–9086 (2019) [3](#)
25. Park, G.Y., Lee, S.W.: Reliably fast adversarial training via latent adversarial perturbation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7758–7767 (2021) [1](#), [5](#)
26. Poon, C., Liang, J.: Trajectory of alternating direction method of multipliers and adaptive acceleration. *Advances in Neural Information Processing Systems* **32** (2019) [2](#), [5](#)
27. Qin, C., Martens, J., Goyal, S., Krishnan, D., Dvijotham, K., Fawzi, A., De, S., Stanforth, R., Kohli, P.: Adversarial robustness through local linearization. *Advances in Neural Information Processing Systems* **32** (2019) [3](#), [5](#)
28. Rocamora, E.A., Liu, F., Chrysos, G.G., Olmos, P.M., Cevher, V.: Efficient local linearity regularization to overcome catastrophic overfitting. *International Conference on Learning Representations* (2024) [1](#), [5](#), [6](#), [11](#)
29. Shaeiri, A., Nobahari, R., Rohban, M.H.: Towards deep learning models resistant to large perturbations. *arXiv preprint arXiv:2003.13370* (2020) [9](#)
30. Shafahi, A., Najibi, M., Ghiasi, M.A., Xu, Z., Dickerson, J., Studer, C., Davis, L.S., Taylor, G., Goldstein, T.: Adversarial training for free! *Advances in Neural Information Processing Systems* **32** (2019) [1](#), [4](#)
31. Song, C., He, K., Wang, L., Hopcroft, J.E.: Improving the generalization of adversarial training with domain adaptation. *International Conference on Learning Representations* (2019) [1](#)
32. Sriramanan, G., Addepalli, S., Baburaj, A., et al.: Guided adversarial attack for evaluating and enhancing adversarial defenses. *Advances in Neural Information Processing Systems* **33**, 20297–20308 (2020) [5](#)
33. Sriramanan, G., Addepalli, S., Baburaj, A., et al.: Towards efficient and effective adversarial training. *Advances in Neural Information Processing Systems* **34**, 11821–11833 (2021) [2](#), [5](#), [10](#), [11](#), [12](#)
34. Themelis, A., Patrinos, P.: Douglas–rachford splitting and admm for nonconvex optimization: Tight convergence results. *SIAM Journal on Optimization* **30**(1), 149–181 (2020) [5](#)

35. Tsiligkaridis, T., Roberts, J.: Understanding and increasing efficiency of frank-wolfe adversarial training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 50–59 (2022) [5](#)
36. Wong, E., Rice, L., Kolter, J.Z.: Fast is better than free: Revisiting adversarial training. International Conference on Learning Representations (2020) [1](#), [2](#), [4](#), [6](#), [9](#), [10](#), [11](#), [12](#), [13](#)
37. Wu, D., Xia, S.T., Wang, Y.: Adversarial weight perturbation helps robust generalization. Advances in Neural Information Processing Systems (2020) [9](#)
38. Xie, Y., Li, Z., Shi, C., Liu, J., Chen, Y., Yuan, B.: Enabling fast and universal audio adversarial attack using generative model. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 14129–14137 (2021) [1](#)
39. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (2016) [8](#)
40. Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., Jordan, M.: Theoretically principled trade-off between robustness and accuracy. In: International conference on machine learning. pp. 7472–7482 (2019) [3](#)
41. Zhang, Y., Zhang, G., Khanduri, P., Hong, M., Chang, S., Liu, S.: Revisiting and advancing fast adversarial training through the lens of bi-level optimization. In: International Conference on Machine Learning. pp. 26693–26712 (2022) [3](#), [5](#)
42. Zhao, M., Zhang, L., Kong, Y., Yin, B.: Fast adversarial training with smooth convergence. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4720–4729 (2023) [2](#), [6](#)