Understanding and Increasing Efficiency of Frank-Wolfe Adversarial Training

Theodoros Tsiligkaridis*, Jay Roberts*
Massachusetts Institute of Technology Lincoln Laboratory
Lexington, MA 02421
ttsili@mit.edu, jay.roberts@ll.mit.edu

2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) | 978-1-6654-6946-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/CVPR52688.2022.00015

Abstract

Deep neural networks are easily fooled by small perturbations known as adversarial attacks. Adversarial Training (AT) is a technique that approximately solves a robust optimization problem to minimize the worst-case loss and is widely regarded as the most effective defense against such attacks. Due to the high computation time for generating strong adversarial examples in the AT process, single-step approaches have been proposed to reduce training time. However, these methods suffer from catastrophic overfitting where adversarial accuracy drops during training, and although improvements have been proposed, they increase training time and robustness is far from that of multi-step AT. We develop a theoretical framework for adversarial training with FW optimization (FW-AT) that reveals a geometric connection between the loss landscape and the distortion of l-inf FW attacks (the attack's l-2 norm). Specifically, we analytically show that high distortion of FW attacks is equivalent to small gradient variation along the attack path. It is then experimentally demonstrated on various deep neural network architectures that l-inf attacks against robust models achieve near maximal l-2 distortion, while standard networks have lower distortion. Furthermore, it is experimentally shown that catastrophic overfitting is strongly correlated with low distortion of FW attacks. This mathematical transparency differentiates FW from the more popular Projected Gradient Descent (PGD) optimization. To demonstrate the utility of our theoretical framework we develop FW-AT-Adapt, a novel adversarial training algorithm which uses a simple distortion measure to adapt the number of attack steps during training to increase efficiency without compromising robustness. FW-AT-Adapt provides training time on par with single-step fast AT methods and improves closing the gap between fast AT methods and multi-step PGD-AT with minimal loss in adversarial accuracy in white-box and black-box settings.

*Equal contributions.

1. Introduction

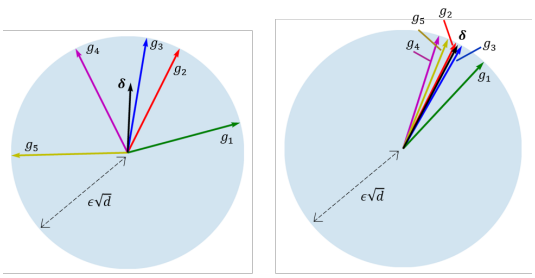
Deep neural networks (DNN) achieve excellent performance across various domains [18]. As these models are deployed across industries (e.g., healthcare or autonomous driving), concerns of robustness and reliability become increasingly important. Several organizations have identified important principles of artificial intelligence (AI) that include the notions of reliability and transparency [19, 21, 24].

One issue of large capacity models such as DNNs is that small, carefully chosen input perturbations, known as adversarial perturbations, can lead to incorrect predictions [12]. Various enhancement methods have been proposed to defend against adversarial perturbations [16, 20, 22, 28]. One of the best performing algorithms is adversarial training (AT) [20], which is formulated as a robust optimization problem [31]. Computation of optimal adversarial perturbations is NP-hard [36] and approximate methods are used to solve the inner maximization. The most popular approximate method that has been proven to be successful is projected gradient descent (PGD) [10]. Frank-Wolfe (FW) optimization has been recently proposed in [8] and was shown to effectively fool standard networks with less distortion, and can be efficiently used to generate sparse counterfactual perturbations to explain model predictions and visualize principal class features [27].

Since PGD has proven to be the main algorithm for adversarially robust deep learning, reducing its high computational cost without sacrificing performance, i.e. fast adversarial training, is a primary issue. Various methods have been proposed based on using a single PGD step, known as Fast Gradient Sign Method (FGSM) [37] but fail for large perturbations. [37] identified that FGSM-based training achieves some robustness initially during training but robustness drastically drops within an epoch, a phenomenon known as *catastrophic overfitting* (CO). While some methods have been proposed to ameliorate this problem [2, 14, 30], the training time suffers as a result and/or robustness is not on par with multi-step PGD-AT.

In this paper, we use the Frank-Wolfe optimization to derive a relationship between the ℓ_2 norm of ℓ_∞ adversarial

978-1-6654-6946-3/22/$31.00 ©2022 IEEE
DOI 10.1109/CVPR52688.2022.00015



(a) Low distortion (b) High distortion

Figure 1. Low (high) distortion of attacks δ (black) is equivalent to high (low) angular spread of signed gradients $g_l = \epsilon \cdot \text{sgn}(\nabla_\delta \mathcal{L}(x + \delta_l, y))$ (all with $\|g_l\|_2 = \epsilon\sqrt{d}$) computed over K steps along attack path ($K = 5$ here). Proposition 1 expresses FW adversarial perturbations δ as convex combinations of signed gradients along the attack path. This core concept is quantified in Theorem 1, and forms the basis for the development of adaptive adversarial training algorithm presented in Section 4.

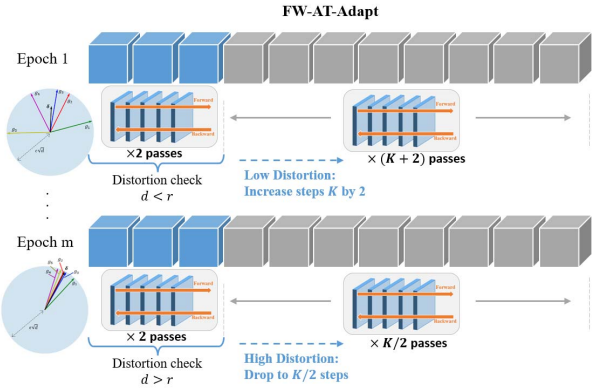


Figure 2. Illustration of concept behind FW-AT-ADAPT training algorithm. At each epoch, distortion d is monitored across the first B_m batches (here $B_m = 3$). If the average distortion is less than a threshold r, the current number of steps, K, is increased by 2, to $K + 2$, (and if it is higher than a threshold r, the current number of steps, K, is dropped by a factor of 2, to $K/2$) for the remaining batches in the epoch. This process is repeated until convergence and reduces the training time of adversarial training without sacrificing robustness. Theorems 2 and 3 provide stability guarantees for robust model weight updates in the high-distortion regime.

perturbations (distortion) and the geometry of the loss landscape (see Fig. 1). Using this theory and empirical studies we show that this distortion can be used as a signal for CO and propose a fast adversarial training algorithm based on an adaptive Frank-Wolfe adversarial training (FW-AT-ADAPT) method (see Fig. 2). This method yields training times on par with single step methods without suffering from CO, outperforms numerous single step methods, and begins to close the gap between fast adversarial training methods and multi-step PGD adversarial training.

Our main contributions are summarized below:

- We demonstrate empirically that FW attacks against robust models achieve near-maximal distortion across a variety of network architectures.
- We empirically show that distortion of FW attacks, even with only 2 steps, are strongly correlated with catastrophic overfitting.
- Theoretical guarantees are derived that relate distortion of FW attacks to the gradient variation along the attack path and which imply that high distortion attacks computed with several steps result in diminishing increases to the loss.
- Inspired by the connection between distortion and attack path gradient variation, we propose an adaptive step Frank-Wolfe adversarial training algorithm, FW-AT-ADAPT , which achieves superior robustness/training-time tradeoffs compared to single-step AT and closes the gap between such methods and multi-step AT variants when evaluated against strong white- and black-box attacks.

2. Background and Previous Work

Consider $(x_i, y_i) \sim \mathcal{D}$ pairs of data examples drawn from distribution $\mathcal{D}$. The labels span C classes. The neural network function $f_\theta(\cdot)$ maps input features into logits, where θ are the model parameters. The predicted class label is given by $\hat{y}(x) = \arg\max_c f_{\theta,c}(x)$.

Adversarial Training. The prevalent way of training classifiers is through empirical risk minimization (ERM):

$$\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathcal{L}(x, y; \theta)] \tag{1}$$

where $\mathcal{L}$ is the usual cross-entropy loss.Adversarial robustness for a classifier f_θ is defined with respect to a metric, here chosen as the ℓ_p metric associated with the ball $B_p(\epsilon) = \{\delta : \|\delta\|_p \leq \epsilon\}$, as follows. A network is said to be robust to adversarial perturbations of size (or strength) ϵ at a given input example x iff $\hat{y}(x) = \hat{y}(x + \delta)$ for all $\delta \in B_p(\epsilon)$, i.e., if the predicted label does not change for all perturbations of size up to ϵ. Training neural networks using the ERM principle (1) gives high accuracy on test sets but leaves the network vulnerable to adversarial attacks.

One of the most popular and effective defenses is adversarial training (AT) [20] which, rather than using the ERM principle, minimizes the adversarial risk

$$\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in B_p(\epsilon)} \mathcal{L}(x + \delta, y; \theta) \right]. \tag{2}$$

This framework was extended in the TRADES algo-

rithm [38] which proposes a modified loss function that captures the clean and adversarial accuracy tradeoff. Local Linearity Regularization (LLR) [25] uses an analogous approach where the adversary is chosen to maximally violate the local linearity based on a first order approximation to (2).

To construct their adversarial attacks at a given input x these defenses use Projected Gradient Descent (PGD) to approximate the constrained inner maximization using a fixed number of steps. PGD computes adversarial perturbations using the iterative updates:

$$\delta_{k+1} = P_{B_p(\epsilon)}\left(\delta_k + \alpha \nabla_\delta \mathcal{L}(x+\delta_k, y; \theta)\right) \tag{3}$$

where $P_{B_p(\epsilon)}(z) = \arg\min_{u \in B_p(\epsilon)} \|z-u\|_2^2$ is the orthogonal projection onto the constraint set. We refer to adversarial training using K step PGD as PGD(K)-AT.

The computational cost of this method is dominated by the number of steps used to approximate the inner maximization, since a K step PGD approximation to the maximization involves K forward-backward propagations through the network. While using fewer PGD steps can lower this cost, these amount to weaker attacks which can lead to gradient obfuscation [23, 34], a phenomenon where networks learn to defend against gradient-based attacks by making the loss landscape highly non-linear, and less robust models. Many defenses have been shown to be evaded by newer attacks, while adversarial training has been demonstrated to maintain state-of-the-art robustness [3, 10]. This performance has only been improved upon via semi-supervised methods [7, 33].

Fast Adversarial Training. Various fast adversarial training methods have been proposed that use fewer PGD steps. In [37] a single step of PGD is used, known as Fast Gradient Sign Method (FGSM), together with random initializations within the constraint ball, called FGSM-RAND, and achieves a good level of robustness at a lower computational cost. In [2] it was shown that the random initialization of FGSM-RAND can improve the linear approximation quality of the inner maximization, but still suffers from catastrophic overfitting (CO), a phenomenon whereby the model achieves strong robustness to the weaker training attack but is completely fooled by stronger multi-step attacks, which the authors overcome via a regularizer which penalizes gradient misalignment and we refer to as FGSM-GA. However, the double backpropagation needed for this method resulted in significantly higher training times than FGSM-RAND. In [14] the authors demonstrate that CO was the result of nonlinearities in the loss which resulted in higher losses on the interior of the ray connecting x and $x+\delta$, thereby making them more susceptible to multi-step attacks. To combat this, the authors adapted the size of the FGSM step by sampling along this ray which we refer to as FGSM-ADAPT. The free adversarial training method of [30], FREE-AT, recycles the gradient information computed when updating model parameters through minibatch replay. The robustness performance of all these single-step AT variants lag far behind that of the multi-step PGD-AT.

Additional methods adapt the number of steps used to approximate adversarial attacks. Curriculum learning [5] monitors adversarial performance during training and increases the number of attack steps as performance improves. Improving on this work the authors in [35] use a Frank-Wolfe convergence criterion to adapt the number of attack steps at a given input. Both of these methods use PGD to generate adversarial examples and do not report improved training times.

Frank-Wolfe Adversarial Attack. The Frank-Wolfe (FW) optimization algorithm has its origins in convex optimization though recently has been shown to perform well in more general settings [11, 13]. The method first optimizes a linear approximation to the original problem, called a Linear Maximization Oracle (LMO)

$$\text{LMO} = \bar{\delta}_k = \underset{\delta \in B_p(\epsilon)}{\text{argmax}} \left\langle \delta, \nabla_\delta \mathcal{L}(x+\delta_k, y) \right\rangle .$$

After calling the LMO, FW takes a step using a convex combination with the current iterate, $\delta_{k+1} = \delta_k + \gamma_k(\bar{\delta}_k - \delta_k)$ where $\gamma_k \in [0,1]$ is the step size. Optimizing step sizes can be found at additional computational cost; however, in practice an effective choice is is $\gamma_k = c/(c+k)$ for some $c \geq 1$.

The FW sub-problem can be solved exactly for any ℓ_p and the optimal $\bar{\delta}_k$ is given component-wise by $\bar{\delta}_{k,i} = \epsilon\, \phi_p(\nabla \mathcal{L}_{k,i})$, where

$$\phi_p(\nabla \mathcal{L}_{k,i}) = \text{sgn}(\nabla \mathcal{L}_{k,i}) \begin{cases} e_i^{i_k^*}, & p = 1 \\ \frac{|\nabla \mathcal{L}_{k,i}|^{q/p}}{\|\nabla \mathcal{L}^k\|_q^{q/p}}, & 1 < p < \infty \\ 1, & p = \infty \end{cases}, \tag{4}$$

$\nabla \mathcal{L} = \nabla_\delta \mathcal{L}(x+\delta_k, y)$, and $1/p + 1/q = 1$. For $p = 1$, $i_k^* = \text{argmax}_i |\nabla \mathcal{L}_{k,i}|$ and $e^{i_k^*}$ is equal to 1 for the i_k^*-th component and zero otherwise. FW does not require a projection onto the ℓ_p ball which is non-trivial for p not in $\{2, \infty\}$. For the special case of ℓ_∞ attacks, the optimal solution becomes the the Fast Gradient Sign Method (FGSM) [12].

Algorithm 1 FW-Attack$(x, y; K, \gamma_k, p)$

Input: Model f_θ, input batch (x, y), max perturbation ϵ, step schedule γ_k, steps K.
$\delta = 0$
for $0 \leq k < K$ **do**
  $\delta = (1-\gamma_k)\delta + \gamma_k\, \epsilon\, \phi_p(\nabla_\delta \mathcal{L}(f_\theta(x+\delta)))$
end for
Return: δ

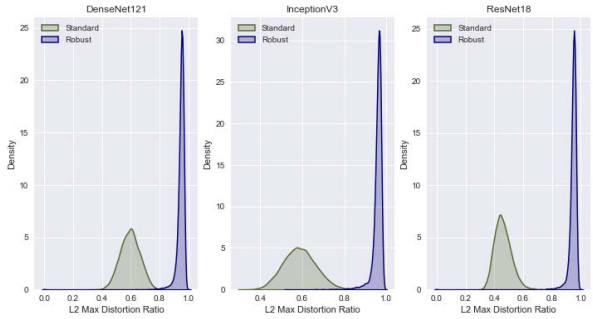


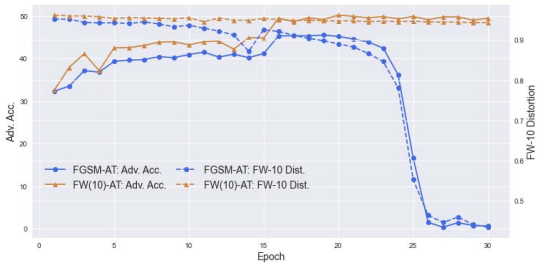
Figure 3. Kernel density estimates of the distribution of $\|\delta\|_2/(\epsilon\sqrt{d})$ attacks computed using FW(20) with $\epsilon = 8/255$ against standard and robust models across three architectures.

Our contributions. We present Frank Wolfe Adversarial Training (FW-AT) which replaces the PGD inner optimization with a Frank-Wolfe optimizer. FW-AT achieves similar robustness as its PGD counterpart. Using a closed form expression for the FW attack path, we derive a geometric relationship between distortion of the attack and loss gradient variation along the attack path. This key insight leads to a simple modification of FW-AT where the step size at each epoch is adapted based on the ℓ_2 distortion of the attacks and is shown to reduce training time while providing strong robustness without suffering from catastrophic overfitting.

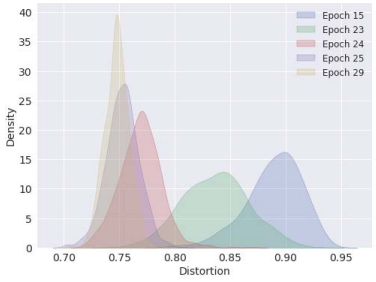
Although our work shares some aspects with FGSM-GA and FGSM-ADAPT it has several distinguishing features. Firstly, both methods are variants on FGSM which attempt to fix CO. The former by penalizing gradient misalignment and the latter by sampling steps along the FGSM direction. Our method takes multiple steps which allows it to reach points near the original FGSM direction and so avoids CO. Moreover, via our distortion analysis we show that our multi-step method can both monitor and regularize gradient variation **all while simultaneously using these attacks for adversarial training**. This efficient use of multi-step attacks allows us to obtain superior robustness training time tradeoffs than either of these prior methods.

3. Distortion of Frank-Wolfe Attacks

Though all ℓ_p attacks must remain in $B_p(\epsilon)$ their ℓ_q norms, for $q \neq p$, can be quite different. This is referred to as distortion and in particular for ℓ_∞ attacks we are interested in insights ℓ_2 distortion can give us into the behavior of the attack and FW-AT. In this setting, the maximal distortion possible of a d dimensional input is $\epsilon\sqrt{d}$, and we refer to $\|\delta\|_2/(\epsilon\sqrt{d})$ as the distortion ratio (or simply distortion) of the attack δ. In this section, we demonstrate empirically the connection between distortion and robustness and then derive theoretical guarantees on loss gradient variation based on distortion bounds.



(a) Adversarial accuracy and Distortion during training for FGSM and FW(10) adversarially trained models



(b) Kernel density estimate of distortion of FW(2) attacks for FGSM-trained model

Figure 4. (a) Adversarial accuracy against PGD(10) attacks (Blue) and average distortion (Tan) of FW(10) attacks on CIFAR-10 validation set for FGSM and FW(10) adversarial training at $\epsilon = 8/255$. Steep drops in adversarial accuracy signaling catastrophic overfitting (CO) are accompanied by drops in distortion. (b) Kernel density estimates of distortion of a FW(2) attack. As CO occurs the distribution of distortion shifts and tightens on low values.

3.1. FW Attacks Against Robust Models are Highly Distorted

Due to its exploitation of constraint convexity one may expect ℓ_∞ FW-attacks to remain near the interior and thus have low distortion. This was observed for standard models in [8] but robust models were not considered. Here we analyze the distortion ratio of FW(20) for ℓ_∞ constrained attacks with radius $\epsilon = 8/255$ on three architectures trained with ERM (Eq. 1) andPGD(10)-AT (Eq. 2) on CIFAR-10.

Figure 3 shows that, while the adversarial perturbations of standard models have small distortion, robust models produce attacks that are nearly maximally distorted. In both cases attacks are near maximal in ℓ_∞ norm. This phenomenon occurs across three different architectures and is further supported by our theory below. We note that for PGD attacks, the distortion ratio can be trivially maximized for large step size α, and thus this connection between distortion and robustness does not exist for PGD optimization.

3.2. Catastrophic Overfitting is Signaled by Distortion Drops

Many fast AT methods rely on a single gradient step which can lead to catastrophic overfitting (CO), a phe-

nomenon where the model's performance against multi-step attacks converges to a high value but then suddenly plummets. This indicates that the model has overfit its weights to the single step training attack.We demonstrate this by training with FGSM at strength $\epsilon = 8/255$ for 30 epochs and plotting its validation accuracy against PGD(10) attacks and average distortion computed with FW(10) attacks. Figure 4a demonstrates the FGSM's descent into CO, where we observe that the drop in adversarial accuracy is mirrored by a drop in the distortion of the multi-step attack. In the case of FW(10)-AT the distortion remains high throughout training.

In Figure 4b we show this behavior is present even when evaluated against the weaker FW(2) attack. Here we plot the kernel density estimate of distortion for a random sample of 1K validation CIFAR-10 images. At the model's peak robustness (Epoch 15) distortion is high, and as the model begins to suffer from CO ($\sim$ Epoch 23) the distribution shifts towards lower values until it strongly accumulates at low values when CO has fully taken effect.

Most interestingly about this result is that FW(2) is able to **detect CO through its distortion without needing to fool the model** (it had a success rate of only 16%). This points to a strong connection between distortion of FW attacks and robustness which make rigorous below.

3.3. Multi-step High Distortion Attacks are Inefficient

Our main tool in analyzing the distortion of FW attacks, and a prime reason FW-AT is more mathematically transparent than PGD-AT, is a representation of the FW attack as a convex combination of the LMO iterates. We refer to the steps taken during the optimization as the attack path. Proofs are included in the Appendix.

Proposition 1. *The* FW *attack with step sizes γ_k yields the following adversarial perturbation after K steps*

$$\delta_K = \epsilon \sum_{l=0}^{K-1} \alpha_l \phi_p(\nabla_\delta \mathcal{L}(x+\delta_l, y)) \tag{5}$$

where $\alpha_l = \gamma_l \prod_{i=l+1}^{K-1}(1-\gamma_i) \in [0,1]$ are non-decreasing in l, and sum to unity.

Proposition 1 shows that the FW adversarial perturbation may be expressed as a convex combination of the signed loss gradients for $p = \infty$, and scaled loss gradients for $p \in [1, \infty)$. Using this representation we can deduce connections between the distortion of the attack and the geometric properties of the attack path.

Theorem 1. *Consider a K step ℓ_∞* FW *Attack. Let $\cos\beta_{lj}$ be the directional cosine between $sgn(\nabla_\delta \mathcal{L}(x+\delta_l, y))$ and $sgn(\nabla_\delta \mathcal{L}(x+\delta_j, y))$. The maximal ℓ_2 distortion ratio of the adversarial perturbation δ_K is:*

$$\frac{\|\delta_K\|_2}{\epsilon\sqrt{d}} = \sqrt{1 - 2\sum_{l<j} \alpha_l \alpha_j (1 - \cos\beta_{lj})} \tag{6}$$

We can summarize the spirit of Theorem 1 as:

Higher distortion is equivalent to lower gradient variation throughout the attack path.

Concretely, the accumulation of sign changes between every step of the attack decreases distortion. In the extreme case of maximally distorted attacks, this implies that the attack is at the corner of the ℓ_∞ ball which could have had **no changes** to the sign of its gradient between any step on the attack path. Therefore each step was constant and the attack is equivalent to a FW(1) attack or FGSM. This is graphically illustrated in Figure 1. Following this logic further, we are able to quantify the distance between different step attacks in terms of the final distortion.

Theorem 2. *Let the same conditions as Theorem 1 hold and $K > 1$. Assume the maximal ℓ_2 distortion ratio of the adversarial perturbation satisfies:*

$$\frac{\|\delta_K\|_2}{\epsilon\sqrt{d}} \geq \sqrt{1-\eta}$$

for some $\eta \in (0,1)$. Then for all intermediate perturbations δ_{k_0}, with $k_0 = 1, \ldots, K$:

$$\frac{\|\delta_K - \delta_{k_0}\|_2}{\epsilon\sqrt{d}} \leq C_{k_0,K}\sqrt{\eta} \tag{7}$$

where $C_{1,K} = \sqrt{K-1}$ and $C_{k_0,K} = \sqrt{\frac{2\sum_{l=0}^{K-1}(\alpha^l)^2 \sum_{j=0}^{k_0-1}(\tilde{\alpha}^j)^2}{\alpha^0\alpha^1}}$ for $k_0 > 1$.

We can summarize the spirit of Theorem 2 as:

Multi-step attacks with high distortion are inefficient.

This suggests that during FW-AT using a large number of steps to approximate the adversarial risk results in diminishing returns once high distortion of the attacks is attained since the final step will be close to the early steps. The other direction is true as well,

Models attacked with low distortion perturbations can benefit from training with more steps.

Intuitively, adversarial attacks with low distortion imply the loss can be maximized at a lower ℓ_2 radius $\epsilon'\sqrt{d} < \epsilon\sqrt{d}$ than the target radius. These loss landscape irregularities are associated with CO as discussed in Section 3.2. Inspired by these two insights we design a FW-AT algorithm which adapts the number of attack steps used in the optimization based on the distortion of a FW(2) attack.

4. Frank-Wolfe Adversarial Training Algorithm

Pseudocode for the Adaptive Frank-Wolfe adversarial training method (FW-AT-ADAPT) is provided in Algorithm 2. The algorithm is graphically depicted in Figure 2 and makes the following modifications to PGD-AT:

(i) Adversarial attacks are computed using a FW optimization scheme (Alg. 1)

(ii) For the first B_m batches of each epoch, the distortion of a FW(2) attack is monitored. If the mean distortion across these batches is above a threshold r then the number of attack steps K is dropped to $K/2$ for the remainder of the epoch. Alternatively if it is lower than r then K is incremented by 2.

Algorithm 2 Epoch of FW-AT-ADAPT

Input: Model f_θ, data $\mathcal{D}$, epoch t, max batch size $|B|$, max perturbation ϵ, step schedule γ_k, learning rate η_t, previous epoch steps K_0, max steps K_1, max distortion ratio r, number of monitoring batches B_m.
Result: Robust model weights θ, current steps K.
$N_b,\ d_m = 0$
$K = 2$ ▷ Check FW(2) Distortion
for each batch $(x, y) \sim \mathcal{D}$ **do**
  δ = FW-Attack$(x, y; K, \gamma_k, p = \infty)$
  $d_m = d_m + \|\delta\|_2/(\epsilon\sqrt{d})$
  $N_b = N_b + 1$
  if $N_b = B_m$ **then**
    if $d_m/B_m > r$ **then** ▷ Check distortion
      $K = \max(1, \lfloor K_0/2 \rfloor)$
    else
      $K = \min(K_1, K_0 + 2)$
    end if
  end if
  $\theta = \theta - \eta_t \frac{1}{|B|} \sum_{i \in B} \nabla_\theta \mathcal{L}(f_\theta(x_i + \delta_i), y_i)$
end for

Next we analyze the effect of using fewer steps on adversarial training weight updates in the high distortion setting. Our analysis shows that in this setting, AT weight updates are minimally affected and thus our method does not sacrifice robustness. While loss functions $\mathcal{L}(f_\theta(x + \delta), y)$ in deep neural networks are non-convex in general, we make the following assumption.

Assumption 1. *The function $\mathcal{L}$ has L-Lipschitz continuous gradients on $B_p(\epsilon)$, i.e., $\|\nabla_\theta \mathcal{L}(f_\theta(x+u), y) - \nabla_\theta \mathcal{L}(f_\theta(x+v), y)\| \leq L\|u - v\|, \forall u, v \in B_p(\epsilon)$.*

Assumption 1 is a standard assumption that has been made in several prior works [32, 35]. Recent works have shown that the loss is semi-smooth in over-parameterized [1, 6, 39] deep neural networks, and batch normalization provides favorable Lipschitz continuity properties [29]. This helps justify Assumption 1.

Theorem 3. *Consider a batch update of* FW-AT *Algorithm 2 where the high distortion condition of Thm. 2 holds on average on examples in a batch B, i.e. for some small $\eta \in (0, 1)$:*

$$\frac{1}{|B|} \sum_{i \in B} \frac{\|\delta_i(K)\|_2}{\epsilon\sqrt{d}} \geq \sqrt{1 - \eta} \tag{8}$$

where $\delta_i(K)$ denotes the K-step FW *adversarial perturbation for the i-th example in the batch B. Let the SGD model weight gradient be given by:*

$$g(\theta, \delta(K)) = \frac{1}{|B|} \sum_{i \in B} \nabla_\theta \mathcal{L}(f_\theta(x_i + \delta_i(K)), y_i)$$

Given Assumption 1 holds, the model weights SGD update using adversarial perturbations δ_K and δ_{k_0} are bounded as:

$$\|g(\theta, \delta(K)) - g(\theta, \delta(k_0))\|_2 \leq LC_{k_0,K}\sqrt{\eta} \cdot \epsilon\sqrt{d}. \tag{9}$$

Bound (9) asserts that in the high distortion setting, the gradients, and thus the weight updates, obtained by a high-step FW attack are near those of a low-step FW attack. Therefore it is expected to achieve a similar level of adversarial robustness using the proposed adaptive algorithm. The proof is included in the Appendix.

4.1. Choosing the Target Distortion Ratio

To provide intuition for the distortion ratio signal hyper-parameter, r, we present the following corollary.

Corollary 1. *(FW(2) Distortion Check) Let s_0 and s_1 be the* LMO*s for the first two steps of a FW adversarial attack. Then if s_1 has k sign changes from s_0 the maximal distortion of δ_1 is*

$$\frac{\|\delta_1\|_2}{\epsilon\sqrt{d}} = \sqrt{1 - \frac{4k}{d}\gamma_1(1 - \gamma_1)} \tag{10}$$

Corollary 1 tells us that the distortion of FW(2) is a function of the number of sign change ratio from the loss gradient at x and at the FGSM attack. For example Figure 4b shows that CO models always have more than 30% sign changes between iterates, whereas more robust can have as few as 10% sign changes.

5. Experimental Results

We evaluate our models on the CIFAR-10 and CIFAR-100 datasets [15] for $\epsilon = 8/255$ and $16/255$. All networks were initialized with the weights of a pretrained standard model

then fine-tuned for 30 epochs via SGD optimization. The learning rate was 0.1 (aside from FGSM-GA) and was then decreased to 0.01 after 15 epochs. We record the time to train the full 30 epochs (aside from FREE-AT). For FW-ADAPT we choose 15 evenly spaced sign change ratios between 15 and 30% then set the distortion check based on Corollary 1.

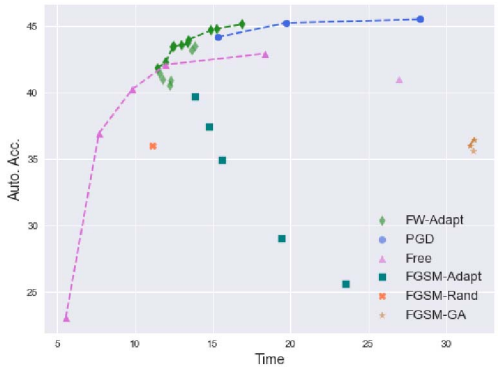
Baselines. We compare against multi-step PGD(K)-AT using a step size of $2.5\epsilon/K$ and $K = 2, 3, 5, 7, 10$. Additionally, we compare against methods which use a single gradient step in their defense. This includes FGSM-RAND, and FGSM-ADAPT with a step size of ϵ and a sweep of checkpoints $c = 2, 3, 4, 8$. Although it arguably uses multiple steps due to minibatch replays and warm starting of attacks we also include FREE-AT in this category, where we sweep the number of minibatch replays $m = 2, 3, 4, 8, 12$ since it obtains comparable training times to other single step methods. We place FGSM-GA in this category, even though it requires additional gradient information to compute its alignment regularization, since it still attacks with one step. Our FW-ADAPT algorithm belongs to a separate category as it aims to efficiently use multiple steps through adaptation, thereby bridging the gap between fixed multi-step and single-step methods.

FREE-AT training for 30 epochs would unfairly increase its training time since the minibatch replay effectively multiplies the number of steps the mode takes. To address this we tuned the number of epochs and minibatches to be near the clean accuracy of competitor methods. Further, FGSM-GA was unable to achieve high accuracy with a learning rate starting at 0.1 and so had its learning rate set to 0.01.

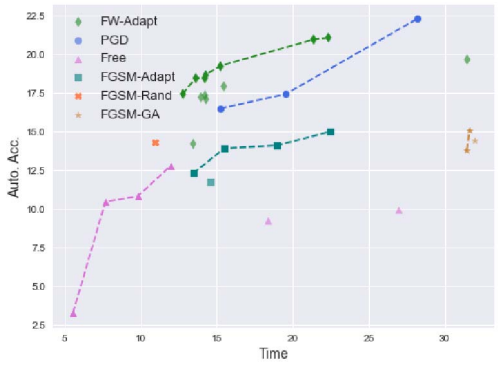
Evaluation Metrics. Robustness is evaluated using a strong white box attack, PGD(50) with step size of $2.5\epsilon/50$. To ensure that we detect gradient masking we also evaluate against AutoAttack (AA) [10], a hyperparameter-free attack suite composed of multiple strong white and black box attacks, making it a strong evaluation metric against gradient masking.

Results. Figure 5 shows the results for parameter sweep on CIFAR-10 of single-step and multi-step methods which trained in less than 35 minutes. Each point represents a different parameter, and the curves show the optimal performance curve which is defined as parameters for which there are none with higher AutoAttack accuracy that trained faster. In general FW-ADAPT obtains superior robustness vs. training time tradeoffs and in particular in the more difficult $\epsilon = 16/255$ we get substantial improvement over other methods.

Comparisons at the endpoints of the performance curves for each method are given in Table 1. We see FW-ADAPT is able to close the gap between single and multi-step methods in terms of robustness without sacrificing speed. In particular we see in the $\epsilon = 16/255$ case single step methods struggle in both clean and adversarial accuracy; whereas FW-ADAPT



(a) $\epsilon = 8/255$



(b) $\epsilon = 16/255$

Figure 5. AutoAttack performance vs. training times tradeoffs on CIFAR-10 for various AT methods. Points represent individual parameters of a method and curves are the optimal performance tradeoffs. FW-ADAPT achieves superior tradeoffs for similar time complexity methods.

is able to achieve similar training times with much higher performance. This suggests that larger attack sizes present a fundamental impediment to using single step methods. Similar performance benefits are observed on the CIFAR-100 dataset in Table 2.

6. Limitations

Our work focuses on obtaining a deeper understanding of the theory behind FW-AT and establishing whether an adaptive version of FW-AT, FW-AT-ADAPT , can offer superior robustness / training time tradeoffs compared to single-step and multi-step AT variants. We showed indeed such superior tradeoffs exist. Future work may focus on developing alternative adaptation strategies and criteria.

7. Conclusion

Adversarial training (AT) provide robustness against ℓ_p-norm adversarial perturbations computed using projected gradient descent (PGD). Through the use of Frank-Wolfe (FW) optimization for the inner maximization an interest-

Method $\epsilon = 8/255$	Clean	PGD(50)	AA	Time (min)
PGD(10)	82.31	50.11	45.38	49.8
PGD(5)	82.46	49.88	45.46	28.3
PGD(2)	83.42	48.50	44.12	15.3
FREE-AT ($m = 2$)	90.10	26.74	23.01	5.5
FREE-AT ($m = 12$)	76.50	45.09	40.97	27.0
FGSM-GA ($\lambda = 0.2$)	77.99	41.44	36.42	31.8
FGSM-ADAPT ($c = 2$)	83.97	43.77	39.65	13.9
FGSM-RAND	78.38	40.64	35.97	11.1
FW-ADAPT ($r = 0.865$)	83.31	45.81	41.80	11.5
FW-ADAPT ($r = 0.900$)	82.34	49.67	45.09	16.9
Method $\epsilon = 16/255$	Clean	PGD(50)	AA	Time (min)
PGD(10)	63.17	31.29	22.62	49.8
PGD(5)	61.72	30.58	22.27	28.2
PGD(2)	63.21	24.76	16.49	15.2
FREE-AT ($m = 2$)	57.69	5.00	3.27	5.5
FREE-AT ($m = 5$)	35.05	15.94	12.77	12.0
FGSM-GA ($\lambda = 0.5$)	55.57	22.72	15.09	31.6
FGSM-ADAPT ($c = 12$)	33.18	18.82	14.99	22.5
FGSM-ADAPT ($c = 2$)	40.44	17.32	12.29	13.5
FGSM-RAND	56.82	22.03	14.27	11.0
FW-ADAPT ($r = 0.830$)	57.78	25.54	17.41	12.8
FW-ADAPT ($r = 0.887$)	58.46	29.57	21.06	22.4

Table 1. Adversarial accuracy computed with PGD(50), AutoAttack (AA) and training time of baseline multi-step PGD (first block), single-step (second block) and FW-ADAPT (third block). Results for parameters at end points of performance curves, including PGD(10), for CIFAR-10 dataset.

Method $\epsilon = 8/255$	Clean	PGD(50)	AA	Time (min)
PGD(10)	59.07	27.37	23.10	49.8
PGD(2)	60.65	26.20	21.99	15.3
FREE-AT ($m = 4$)	60.16	23.20	19.27	9.9
FGSM-GA ($\lambda = 0.2$)	56.53	20.02	16.15	31.5
FGSM-ADAPT ($c = 2$)	49.28	20.19	15.97	13.7
FGSM-RAND	50.20	20.63	16.47	11.0
FW-ADAPT ($r = 0.830$)	61.12	23.20	19.97	11.9
FW-ADAPT ($r = 0.899$)	60.60	25.41	21.64	15.1
Method $\epsilon = 16/255$	Clean	PGD(50)	AA	Time (min)
PGD(10)	40.49	16.46	11.30	49.8
PGD(5)	41.99	15.71	10.88	28.3
PGD(2)	33.17	10.54	7.14	15.3
FREE-AT ($m = 4$)	47.06	8.83	6.34	9.8
FGSM-GA ($\lambda = 0.2$)	37.04	7.88	4.67	32.2
FGSM-ADAPT ($c = 2$)	8.38	4.85	3.55	13.5
FGSM-RAND	24.78	6.96	3.92	11.0
FW-ADAPT ($r = 0.830$)	44.65	11.52	7.99	13.6
FW-ADAPT ($r = 0.887$)	40.91	15.33	10.47	24.0

Table 2. Adversarial accuracy computed with PGD(50), AutoAttack (AA) and training time of baseline multi-step PGD (first block), single-step (second block) and FW-ADAPT (third block). Results for select top-performing models, including PGD(10), for CIFAR-100 dataset.

ing phenomenon occurs: FW attacks against robust models result in higher ℓ_2 distortions than standard ones despite achieving nearly the same ℓ_∞ distortion. We derive a theoretical connection between loss gradient alignment along the attack path and the distortion of FW attacks which explains this phenomenon. We provide theoretical and empirical evidence that this distortion can signal catastrophic overfitting in single-step fast AT models. Inspired by this connection, we propose an adaptive Frank-Wolfe adversarial training (FW-AT-ADAPT) algorithm that achieves robustness above single-step baselines while maintaining competitive training times particularly in the strong ℓ_∞ attack regime. This work begins to close the gap between robustness training time trade-offs of single-step and multi-step methods and hope it will inspire future research on the connection between Frank-Wolfe optimization and adversarial robustness.

Societal Impact Statement

As DNNs are increasingly being deployed for safety-critical applications, such as healthcare, autonomous driving, and biometrics, robustness against adversarial attacks is a rising concern. Addressing this is critical to gain public trust and avoid denial of opportunity. One of the most popular and effective defenses is adversarial training (AT). However, the popular multi-step PGD optimization approach used in AT cannot be easily analyzed to obtain insights into what type of regularization AT induces, and it also requires multiple steps in the inner maximization leading to slow training. To reduce training time, single-step approaches have been proposed, but are prone to catastrophic overfitting, leading to a false sense of robustness. This can have severe consequences in security applications. Our work improves the understanding of AT via the lens of FW optimization and provides simple methods for efficient training of robust models without compromising robustness.

Acknowledgements

Research was sponsored by the United States Air Force Research Laboratory and the United States Air Force Artificial Intelligence Accelerator and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Air Force or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[1] Z. Allen-Zhu, Y. Li, and Z. Song. A convergence theory for deep learning via over-parameterization. In ICML, 2019. 6

[2] M. Andriushchenko and N. Flammarion. Understanding and improving fast adversarial training. In NeurIPS, 2020. 1, 3, 12

[3] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In ICML, 2018. 3

[4] D. P. Bertsekas. Nonlinear Programming. Athena Scientific, 1999. 16

[5] Q.-Z. Cai, C. Liu, and D. Song. Curriculum adversarial training. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, pages 3740–3747. International Joint Conferences on Artificial Intelligence Organization, 7 2018. 3

[6] Y. Cao and Q. Gu. Generalization error bounds of gradient descent for learning over-parameterized deep relu networks. In AAAI, 2020. 6

[7] Y. Carmon, A. Raghunathan, L. Schmidt, P. Liang, and J. C. Duchi. Unlabeled data improves adversarial robustness. In NeurIPS, 2019. 3

[8] J. Chen, D. Zhou, J. Yi, and Q. Gu. A frank-wolfe framework for efficient and effective adversarial attacks. In Thirty-Fourth AAAI Conference on Artificial Intelligence, 2020. 1, 4, 16

[9] N. Codella, V. Rotemberg, P. Tschandl, M. E. Celebi, S. Dusza, D. Gutman, B. Helba, A. Kalloo, K. Liopyris, M. Marchetti, H. Kittler, and A. Halpern. Isic2018: Skin lesion analysis towards melanoma detection. 11

[10] F. Croce and M. Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In ICML, 2020. 1, 3, 7

[11] M. Frank and P. Wolfe. An algorithm for quadratic programming. Naval research logistics quarterly, 3:95–110, 1956. 3, 16

[12] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In International Conference on Learning Representations, 2015. 1, 3

[13] M. Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In ICML, pages 427–435, 2013. 3

[14] H. Kim, W. Lee, and J. Lee. Understanding catastrophic overfitting in single-step adversarial training. In AAAI, 2021. 1, 3

[15] A. Krizhevsky. Cifar-10 and cifar-100 datasets. 6

[16] A. Kurakin, I. J. Goodfellow, and S. Bengio. Adversarial machine learning at scale. In International Conference on Learning Representations, 2017. 1

[17] S. Lacoste-Julien. Convergence rate of frank-wolfe for non-convex objectives. In arXiv: 1607.00345, 2016. 16

[18] Y. LeCun, Y. Bengio, and G. Hinton. Deep Learning. Nature, 521(7533):436–444, 2015. 1

[19] C. T. Lopez. DOD Adopts 5 Principles of Artificial Intelligence Ethics, 2020. 1

[20] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In International Conference on Learning Representations, 2018. 1, 2

[21] Microsoft. Microsoft AI principles, 2019. 1

[22] S.-M. Moosavi-Dezfooli, J. Uesato, A. Fawzi, and P. Frossard. Robustness via curvature regularization, and vice versa. In IEEE Conference on Computer Vision and Pattern Recognition, 2019. 1, 16

[23] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami. Practical black-box attacks against machine learning. In arXiv:1602.02697v4, 2017. 3

[24] S. Pichai. AI at Google: our principles, 2018. 1

[25] C. Qin, J. Martens, S. Gowal, D. Krishnan, K. Dvijotham, A. Fawzi, S. De, R. Stanforth, and P. Kohli. Adversarial robustness through local linearization. In NeurIPS, 2019. 3, 16

[26] J. Rector-Brooks, J.-K. Wang, and B. Mozafari. Revisiting projection-free optimization for strongly convex constraint sets. In AAAI, 2019. 16

[27] J. Roberts and T. Tsiligkaridis. Controllably sparse perturbations of robust classifiers for explaining predictions and probing learnt concepts. In MLVis International Workshop on Machine Learning in Visualisation for Big Data, 2021. 1

[28] A. S. Ros and F. Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In AAAI Conference on Artificial Intelligence, 2018. 1

[29] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry. How does batch normalization help optimization? In NeurIPS, 2018. 6, 16

[30] A. Shafahi, M. Najibi, A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein. Adversarial training for free! In NeurIPS, 2019. 1, 3

[31] U. Shaham, Y. Yamada, and S. Negahban. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. Neurocomputing, 2018. 1

[32] A. Sinha, H. Namkoong, and J. Duchi. Certifying some distributional robustness with principled adversarial training. In ICLR, 2018. 6

[33] J. Uesato, J.-B. Alayrac, P.-S. Huang, R. Stanforth, A. Fawzi, and P. Kohli. Are labels required for improvingadversarial robustness? In NeurIPS, 2019. 3

[34] J. Uesato, B. O'Donoghue, A. van den Oord, and P. Kohli. Adversarial risk and the dangers of evaluating against weak attacks. In ICML, 2018. 3

[35] Y. Wang, X. Ma, J. Bailey, J. Yi, B. Zhou, and Q. Gu. On the convergence and robustness of adversarial training. In ICML, 2019. 3, 6

[36] T.-W. Weng, H. Zhang, H. Chen, Z. Song, C.-J. Hsieh, D. Boning, I. S. Dhillon, and L. Daniel. Towards fast computation of certified robustness for relu networks. In ICML, 2018. 1

[37] E. Wong, L. Rice, and J. Z. Kolter. Fast is better than free: Revisiting adversarial training. In ICLR, 2020. 1, 3

[38] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan. Theoretically principled trade-off between robustness and accuracy. In ICML, 2019. 3

[39] D. Zou and Q. Gu. An improved analysis of training over-parameterized deep neural networks. In NeurIPS, 2019. 6