Adaptive Certified Training:
Towards Better Accuracy-Robustness Tradeoffs

Zhakshylyk Nurlanov [1 2]   **Frank R. Schmidt** [2]   **Florian Bernard** [1]

Abstract

As deep learning models continue to advance and are increasingly utilized in real-world systems, the issue of robustness remains a major challenge. Existing certified training methods produce models that achieve high provable robustness guarantees at certain perturbation levels. However, the main problem of such models is a dramatically low standard accuracy, i.e. accuracy on clean unperturbed data, that makes them impractical. In this work, we consider a more realistic perspective of maximizing the robustness of a model at certain levels of (high) standard accuracy. To this end, we propose a novel certified training method based on a key insight that training with adaptive certified radii helps to improve both the accuracy and robustness of the model, advancing state-of-the-art accuracy-robustness tradeoffs. We demonstrate the effectiveness of the proposed method on MNIST, CIFAR-10, and TinyImageNet datasets. Particularly, on CIFAR-10 and TinyImageNet, our method yields models with up to two times higher robustness, measured as an average certified radius of a test set, at the same levels of standard accuracy compared to baseline approaches.

1. Introduction

Deep learning models are successfully deployed in real-world applications, related to speech and image recognition, natural language processing, drug discovery, recommendation systems, and bioinformatics (Balas et al., 2019). However, the issue of robustness poses safety concerns in mission-critical tasks, such as autonomous driving (Zhang et al., 2021a), since the discovery of adversarial examples (Szegedy et al., 2014). Methods for improving empirical robustness, such as adversarial training (Madry et al., 2018), are effective against a set of adversarial attacks but give no robustness guarantees. Formal verification techniques (Müller et al., 2022) can provide worst-case deterministic performance guarantees. Nonetheless, to improve the robustness certificates, specialized certified training methods (Wong et al., 2018; Zhang et al., 2020; Shi et al., 2021; Müller et al., 2023) are necessary. Existing certified training approaches improve worst-case certified accuracy at the cost of drastically reduced standard accuracy on clean inputs. This limits the adoption of certifiably robust models in practical applications.

In this work, we tackle the problem of reduced standard accuracy of certified training methods and propose a novel approach, **A**daptive **Cert**ified Training (ACERT). It is based on the idea that the input points may have different levels of vulnerability. Therefore, by adapting certification intensity to each input sample individually, ACERT optimizes robustness on all levels. As a result, it achieves increased standard accuracy and robustness, measured as an average certified radius of a test set, advancing the state-of-the-art. Moreover, we propose to evaluate models based on their performance across accuracy-robustness curves, therefore we introduce a new metric called Accuracy-Robustness Tradeoff (ART) score, which combines standard accuracy and average certified radius.

Main contributions   Our main contributions are summarized in the following:

1. An empirical study of accuracy-robustness tradeoffs of certified training methods with a new metric, ART score (short for Accuracy-Robustness Tradeoff), combining standard accuracy and average certified robustness of models;

2. A novel certified training method called ACERT that efficiently finds theoretically sound certification strength for each input sample and effectively maximizes it;

3. Experiments demonstrating improved accuracy-robustness curves of ACERT over previous state-of-the-art methods across different datasets, such as MNIST, CIFAR-10, and TinyImageNet.

[1]University of Bonn [2]Bosch Center for Artificial Intelligence. Correspondence to: Zhakshylyk Nurlanov <zh.nurlanov@uni-bonn.de>.

2. Background

In the following, we denote a deep neural network (DNN) as a parameterized function $f_\theta : \mathcal{X} \to \mathcal{Y}$ with parameters θ, an input space $\mathcal{X}$, and an output space $\mathcal{Y}$. Consider a classification problem, where $\mathcal{X} \subset \mathbb{R}^{d_{\text{in}}}$, $\mathcal{Y} = \mathbb{R}^c$ and $f_\theta(x) \in \mathbb{R}^c$ predict numerical scores (i.e., unnormalized *logits*) for each of the c classes. Hard-label classification $h(f_\theta(x))$ is performed by assigning an index of the highest score, that is $h(f_\theta(x)) := \arg\max_i f_\theta(x)_i$.

Adversarial Robustness A hard-label classifier h is *adversarially robust* on an ℓ_p-ball $\mathcal{B}_p(x, \varepsilon)$ around input x if, given a ground-truth target class y, it always predicts y for all inputs in $\mathcal{B}_p(x, \varepsilon)$, or formally,

$$h(f_\theta(x')) \equiv y, \qquad (1)$$
$$\forall x' \in \mathcal{B}_p(x, \varepsilon) := \{x' \in \mathcal{X} : ||x - x'||_p \leq \varepsilon\}.$$

Equivalently, a soft classifier f_θ is *adversarially robust* on an ℓ_p-ball $\mathcal{B}_p(x, \varepsilon)$ around input x, if the maximum value of a margin function $m(x', y; f_\theta) := \max_{i \neq y} f_\theta(x')_i - f_\theta(x')_y$ on $x' \in \mathcal{B}_p(x, \varepsilon)$ is less than 0, i.e.

$$R(x, y; \mathcal{B}_p(x, \varepsilon); f_\theta) := \max_{x' \in \mathcal{B}_p(x, \varepsilon)} m(x', y; f_\theta) < 0. \quad (2)$$

A solution to the maximization problem in (2) defines a *robust margin* function $R(x, y; \mathcal{B}_p(x, \varepsilon); f_\theta)$, which indicates the robustness of the underlying classifier f_θ.

In this work, we consider a common case of $p = \infty$ for ℓ_p balls, i.e. $\mathcal{B}(x, \varepsilon) = \mathcal{B}_\infty(x, \varepsilon)$. Furthermore, whenever it is clear from context, we omit extra arguments of the robust margin, i.e. we write $R(x, \varepsilon) = R(x, y; \mathcal{B}_p(x, \varepsilon); f_\theta)$.

Certified Robustness Calculating the robust margin function in (2) is a challenging task, as it is in general an NP-hard problem (Katz et al., 2017). Therefore, to guarantee robustness, it is preferable to calculate an upper bound of the robust margin, called the *certified robust margin* denoted as $R_{\text{cert}} \geq R$. The calculation of the certified robust margin may become significantly easier, depending on the tightness of the upper bounds. The certified robust margin is often computed with the help of the outer bounds of the neural network output, namely

$$R_{\text{cert}}(x, \varepsilon) := \max_{i \neq y} \overline{f_\theta}(x, \varepsilon)_i - \underline{f_\theta}(x, \varepsilon)_y, \qquad (3)$$

where $\underline{f_\theta}(x, \varepsilon) \leq f_\theta(x') \leq \overline{f_\theta}(x, \varepsilon), \forall x' \in \mathcal{B}(x, \varepsilon)$ represent the outer bounds.

With that, the classifier f_θ is *certifiably robust* on $\mathcal{B}(x, \varepsilon)$ if the upper bound of the robust margin R_{cert} is less than 0, i.e.

$$R(x, \varepsilon) \leq R_{\text{cert}}(x, \varepsilon) < 0. \qquad (4)$$

Certified Training It can be difficult to successfully verify the robustness of regularly trained neural networks, as they often have low certification rates. To address this issue, specialized training techniques called *certified training*, are used to improve the robustness guarantees of these networks.

A particularly effective certified training method is based on the interval bound propagation (IBP) technique (Gowal et al., 2019). It propagates the input interval bounds $\mathcal{B}(x, \varepsilon)$ through the layers of a DNN using interval arithmetic to get the output intervals $(\underline{f_\theta}(x, \varepsilon), \overline{f_\theta}(x, \varepsilon))$ (details of bound propagation in IBP are provided in the Appendix A.1). Calculating the IBP bounds is comparable to computing a forward pass through an extended network. Therefore, computational efficiency together with favorable optimization properties (Jovanović et al., 2022) allows one to train certifiably robust models by directly optimizing the loss function over the IBP bounds.

In practice, IBP-based certified training minimizes a differentiable approximation of the certified robust margin function, a cross-entropy loss

$$\mathcal{L}_{\text{rob}}^{\text{IBP}}(x, y, \varepsilon_t; \theta) := \mathcal{L}_{\text{CE}}(\hat{f}_\theta(x, y, \varepsilon_t), y), \qquad (5)$$

where the radius ε_t is shared across inputs and is heuristically scheduled to increase during training from 0 to $\varepsilon_{\max}$. The worst-case bounds of a DNN's output $\hat{f}_\theta(x, y, \varepsilon_t)$ in (5) are formed such that

$$\hat{f}_\theta(x, y, \varepsilon_t)_i := \begin{cases} \overline{f_\theta}(x, \varepsilon_t)_i, & \forall i \neq y, \\ \underline{f_\theta}(x, \varepsilon_t)_y, & i = y. \end{cases} \qquad (6)$$

Optimizing the robust loss outlined in (5) yields models that can achieve an impressive level of certified accuracies at certain (large) perturbation levels. However, this often leads to a decrease in accuracy on clean inputs (also known as *standard accuracy*) due to overregularization. Balancing robustness and accuracy remains a challenge for these methods.

3. Method

In the following, we propose the **A**daptive **Cert**ified training method, ACERT, which addresses the issue of low standard accuracy in existing certified training approaches. First, we define the concept of certified robust radius, which is an integral part of our method. Then, we formalize the adaptive certified training, and finally, present a new metric to capture progress in both accuracy and robustness.

3.1. Certified Robust Radius

Definition 3.1 (Certified robust radius). A bounded certified robust radius $\mathcal{E}(x) = \mathcal{E}(x, y; f_\theta; p) \in [0, \varepsilon_{\max}]$ of an input sample $(x, y) \sim \mathcal{D}$ with respect to the classifier f_θ is the

maximum radius of an ℓ_p ball $\mathcal{B}(x, \varepsilon)$, not exceeding the predefined (perceptual) limit $\varepsilon_{\max}$, where the classifier is certifiably robust. Formally,

$$\mathcal{E}_{\text{cert}}(x) = \begin{cases} 0, & \text{if } R_{\text{cert}}(x, 0) \geq 0, \\ \varepsilon_{\max}, & \text{if } R_{\text{cert}}(x, \varepsilon_{\max}) < 0, \\ \sup_{R_{\text{cert}}(x, \varepsilon) < 0} \varepsilon, & \text{otherwise.} \end{cases} \quad (7)$$

Theorem 3.2. *Let $R_{cert}(x, \cdot)$ be continuous and strictly monotonically increasing function. Then the certified robust radius, defined in Def. 3.1, exists and is uniquely defined as*

$$\mathcal{E}_{cert}(x) = \begin{cases} 0, & \text{if } R_{cert}(x, 0) > 0, \\ \varepsilon_{\max}, & \text{if } R_{cert}(x, \varepsilon_{\max}) < 0, \\ \varepsilon \text{ s.t. } R_{cert}(x, \varepsilon) = 0, & \text{otherwise.} \end{cases}$$
$$(8)$$

The existence and uniqueness of the certified robust radius follow from continuity and strict monotonicity of $R_{\text{cert}}(x, \cdot)$. Find detailed proofs in the Appendix A.2. Further, whenever it is clear, we omit the extra arguments in the certified robust radius function, i.e. $\mathcal{E}_{\text{cert}}(x) = \mathcal{E}_{\text{cert}}(x, y; f_\theta, p)$.

Remark 3.3. We observe that in the regularly initialized classifiers f_θ, interval bound propagation-based certified margin function $R_{\text{cert}}(x, \cdot)$ in practice satisfies the requirements of Theorem 3.2.

As a result, using IBP for computing the outer bounds of a DNN's output, we can find the certified robust radius of input x as a root of a scalar function $R_{\text{cert}}(x, \cdot)$, bounded by perceptual limits $[0, \varepsilon_{\max}]$.

3.2. Adaptive Certified Training

Employing the definition of certified robust radius from Theorem 3.2, we show that it is tractable to directly maximize the radius of each input sample with gradient-based techniques.

Theorem 3.4 (Implicit function). *Let $R_{cert}(\cdot, \cdot)$ be a continuously differentiable function of arguments (w, ε), and let (w_0, ε_0) be a point such that $R_{cert}(w_0, \varepsilon_0) = 0$. If $\partial_\varepsilon R_{cert}(w_0, \varepsilon_0) \neq 0$, by implicit function theorem*

$$\partial_w \mathcal{E}_{cert}(w) = -\frac{\partial_w R_{cert}(w, \mathcal{E}_{cert}(w))}{\partial_\varepsilon R_{cert}(w, \mathcal{E}_{cert}(w))}$$

Remark 3.5. Note that the derivative $\partial_\varepsilon R_{\text{cert}}(w, \mathcal{E}_{\text{cert}}(w))$ is always a positive scalar for a strictly increasing differentiable function, and that the first argument w can represent both input x and parameters θ. Hence, Theorem 3.4 implies that a gradient step in a direction of maximizing the certified robust radius $\partial_\theta \mathcal{E}_{\text{cert}}(\theta)$ is a scaled gradient step in a direction of minimizing the certified margin $\partial_\theta R_{\text{cert}}(\theta, \mathcal{E}_{\text{cert}}(\theta))$.

Therefore, by minimizing the certified margin at the point of certified robust radius, i.e. $\min_\theta R_{\text{cert}}(x, \mathcal{E}_{\text{cert}}(x); \theta)$, we equivalently maximize the corresponding certified robust radius $\max_\theta \mathcal{E}_{\text{cert}}(x; \theta)$. In practice, similar to IBP-based certified training (5), we minimize an approximation of the certified margin that is better suitable for gradient-based optimizers, a cross-entropy loss $\mathcal{L}_{\text{CE}}$.

We employ the worst-case output bounds based on IBP as in Eq. (6), with the key difference that the input bounds $\mathcal{B}(x, \varepsilon)$ have adaptive radii ε, which are equal to certified robust radii for each of the inputs $\mathcal{E}_{\text{cert}}(x)$, that is $\hat{f}_\theta(x, y, \varepsilon) \equiv \hat{f}_\theta(x, y, \mathcal{E}_{\text{cert}}(x))$. Overall, the robust loss of the Adaptive Certified training method is defined as

$$\mathcal{L}_{\text{rob}}^{\text{ACERT}}(x, y; \theta) := \mathcal{L}_{\text{CE}}(\hat{f}_\theta(x, y, \mathcal{E}_{\text{cert}}(x)), y), \quad (9)$$

To capture the tradeoff between standard accuracy and robustness, we construct a final loss as a convex combination of standard cross entropy loss and a robust loss, both for IBP and ACERT, i.e.

$$\mathcal{L}_{\text{stn}}(x, y; \theta) := \mathcal{L}_{\text{CE}}(f_\theta(x), y), \quad (10)$$
$$\mathcal{L} = \kappa \mathcal{L}_{\text{stn}} + (1 - \kappa) \mathcal{L}_{\text{rob}}. \quad (11)$$

By varying the proportion of standard loss $\kappa \in [0, 1]$, we can get the accuracy-robustness curves of certified training methods (e.g. see in Fig. 1).

3.3. Accuracy-Robustness Tradeoff Score

In this section, we present metrics that we use for measuring both the accuracy and robustness of models.

The standard accuracy on clean data is the proportion of correctly classified samples (Acc).

The average robustness of a model is characterized by an average certified robust radius, which is bounded in an interval $[0, \varepsilon_{\max}]$. To compare performance across different perceptual limits $\varepsilon_{\max}$, we normalize the average certified radius (ACR), that is

$$\text{ACR} := \frac{1}{\varepsilon_{\max}} \cdot \frac{\sum_i^N \mathcal{E}_{\text{cert}}(x_i, y_i; f_\theta)}{N}. \quad (12)$$

To measure the progress in both standard accuracy and robustness, we propose to use a new metric, ART (Accuracy-Robustness Tradeoff) that performs a geometric averaging of standard accuracy and average certified radius. Formally,

$$\text{ART}(\text{Acc}, \text{ACR}) := \sqrt{\text{Acc} \times \text{ACR}}. \quad (13)$$

The main advantage of the geometric mean over other averaging functions is that it is *homogeneous* as a function of each argument. Therefore, changing a scaling factor in the computation of an average certified radius (12) would
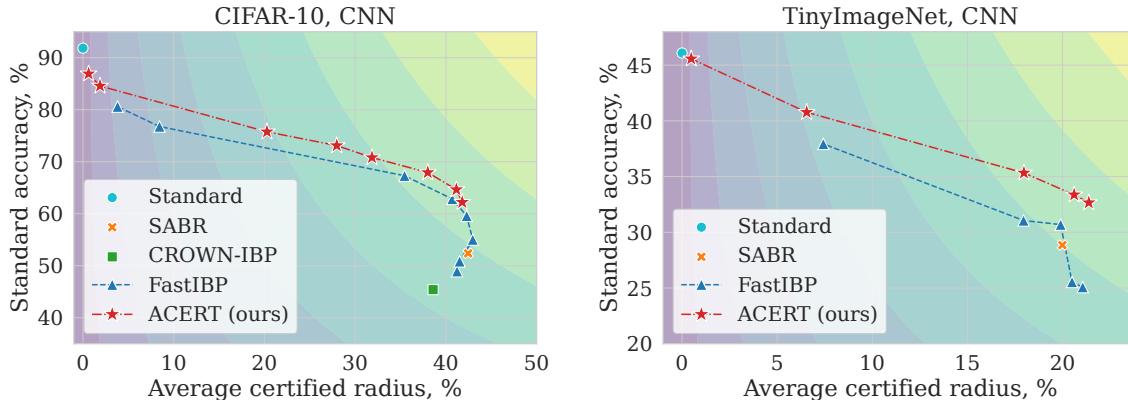
Figure 1. Accuracy vs robustness (given as an average certified robust radius of a test set) tradeoffs of certified training methods produced by varying the proportion of standard loss $\kappa \in [0, 1]$ in a total loss function (11). The ART score (13), or geometric mean of accuracy and average certified radius, is visualized with a colormap, where lighter colors represent higher values.

only result in a different multiplicative constant of ART, and would not alter the order of compared methods. Formally, $\text{ART}(z_1, c \cdot z_2) = \sqrt{c} \cdot \text{ART}(z_1, z_2)$. Introducing the ART score helps to select models from a Pareto front of the accuracy-robustness curve.

4. Experiments

In the following section, we demonstrate the performance of our proposed method, ACERT, and compare it with state-of-the-art certified training methods on various datasets. Furthermore, we investigate the effect of each component of our method in ablation studies.

Experimental Setup We conduct experiments on three image classification datasets of various difficulties and scales: MNIST (LeCun, 1998), CIFAR-10 (Krizhevsky, 2009) and TinyImageNet (Le & Yang, 2015).

The baseline method, FastIBP (Shi et al., 2021), gives strong IBP-based certified robustness requiring relatively short training schedules. We reimplement it based on the authors' code and run experiments in the same setups as our method and for the same values of $\kappa \in [0, 1)$. For CROWN-IBP (Zhang et al., 2020) and a more recent SABR (Müller et al., 2023), we calculate certified robust radii with IBP on the pre-trained checkpoints provided by authors.

The training procedures, such as IBP initialization, regularizers, and training schedules, follow the prior work, FastIBP (Shi et al., 2021). The CNN7 architecture is used across datasets, similar to (Shi et al., 2021; Müller et al., 2023). For our method, in the initial stages of training, we schedule the maximally allowed certified radius $\varepsilon_{\max,t}$ to increase from 0 to $\varepsilon_{\max}$. The verification of all methods is performed by the efficient IBP technique since considered models are trained using the IBP bounds. Find more

details on hyperparameters, hardware, training times, and reproducibility in the Appendix B. We will release the code together with the paper publication.

We implement an efficient vectorized GPU-accelerated version of the BrentQ (Brent, 1973) algorithm to find adaptive certified radii (8) as roots of a batch of scalar functions $(R_{\text{cert}}(x^1, \cdot), \ldots, R_{\text{cert}}(x^b, \cdot))$. The BrentQ algorithm guarantees convergence with any predefined level of error tolerance, and it is generally considered a powerful root finding routine (Press et al., 2007). In practice, it usually requires 2 to 4 iterations to achieve solutions with absolute error tolerance `xtol=1e-6` and relative tolerance `rtol=1e-4`. Therefore, this routine can be used during the training of a neural network, allowing one to efficiently compute certified radii with high precision. A more detailed analysis of the root finding algorithm is given in Section 4.2.2.

4.1. Main Results

We compare ACERT to state-of-the-art certified training methods in terms of standard accuracy, average certified robust radius on a test dataset, and a combination of those, ART score, in Fig. 1 and Table 1. Based on the accuracy-robustness curves presented in Fig. 1, it can be inferred that the proposed method, ACERT, demonstrates higher levels of robustness at the same levels of accuracy (Fig. 1) comparing to baseline. For instance, at a fixed level of standard accuracy of 75% on CIFAR-10, ACERT provides twice ($\times 2$) of the average certified radius given by FastIBP (18% vs 9%). Similarly, ACERT reaches around twice ($\times 2$) of the robustness level of the baseline method on TinyImageNet (14% vs 7%) for a standard accuracy of 38%. Additionally, we observe in Fig. 1 that the accuracy-robustness curves of ACERT consistently get close to the regularly trained network's point with high accuracy, while the baseline fails

to reach those levels. This indicates that adaptive training successfully eliminates unnecessary overregularization, which is present in certified training. Overall, the accuracy-robustness curves of ACERT dominate those of the baseline across datasets, resulting in superior Pareto fronts.

For its part, the baseline method, FastIBP, exhibits accuracy and robustness levels similar to those of SABR, provided that κ (the proportion of standard loss) is tuned and the same verification technique is utilized, and it outperforms CROWN-IBP. So analyzing the performance of our method in comparison to FastIBP gives a fair judgment.

To quantitatively evaluate the certified training methods, we compare the ART (Accuracy-Robustness Tradeoff) scores in Table 1. We provide results for $\kappa = 0$ for all methods since it is a common setup in previous works. Moreover, for the baseline FastIBP and our method, we tune the hyperparameter $\kappa \in [0, 1]$ to get the highest ART score (except for the MNIST dataset, where standard accuracies are already close to the highest possible one). Overall, ACERT outperforms previous state-of-the-art certified training methods with and without tuning of κ in terms of ART score. The effect of adaptive certified training is more prominent on a challenging TinyImageNet dataset, where it achieves both higher standard accuracy and average certified robust radius at the same time.

4.2. Ablation Studies

4.2.1. THE EFFECT OF THE ε SCHEDULER

Since certified training methods rely on a scheduler for certification radius, i.e. ε_t, during a warm-up stage of training, we investigate the effect of the scheduler on the final performance. The proposed method, ACERT, relies on a scheduler of individual maximally allowed perturbation radius, i.e. $\varepsilon_{\max,t}$.

Table 2 shows performances of models trained without ε schedulers on CIFAR-10 dataset, where $\varepsilon_{\max}$ is set to $^{8.8}/255$. We observe that all training methods have a drop in accuracy and robustness, reflected in the Δ ART column, compared to the default training with corresponding ε schedulers. Without the schedulers, ACERT has the highest ART score and the lowest drop in performance (Δ ART), compared to baseline models. It is also notable that the performance of baseline certified training, FastIBP, improves both in terms of accuracy and robustness when the standard loss with the weight of $\kappa = 0.8$ is introduced.

Even though the proposed adaptive certified training internally assigns certification strength ε_i to each input x_i (providing a theoretically defined automatic scheduler), in the early stages of training the model might get overregularized whenever it processes inputs with high certified radii, limiting the final performance. Therefore, it is reasonable
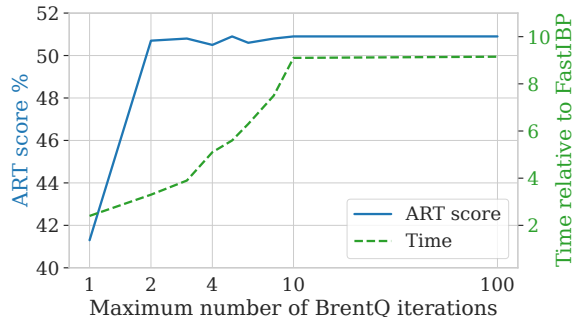

Figure 2. Effect of the root finding algorithm's precision (given as a maximum number of iterations) on computation time (relative to FastIBP) and final ART score of ACERT on CIFAR-10 dataset.

to cap the maximally allowed perturbation radius during a warm-up stage.

4.2.2. THE EFFECT OF THE ROOT FINDER

Since ACERT finds certified robust radii as roots of a scalar function $R_{\text{cert}}(x, \cdot)$, several additional certifications (forward passes for IBP) are required. In Fig. 2, we study how the precision of a root-finding algorithm, BrentQ, affects its computational efficiency and overall training performance. We train ACERT models with varying maximum numbers of BrentQ iterations, which limits the precision of the algorithm. Surprisingly, we notice that already 2 iterations of the BrentQ routine allow us to determine roots of $R_{\text{cert}}(x, \cdot)$ with sufficient precision to reach high ART scores. Since it runs 3 additional certifications (2 without gradients and 1 for loss computation), it is around 3.3 times slower than FastIBP's training step. Further optimization of the root finding algorithm's efficiency, such as saving and using history from previous epochs, can additionally reduce training times.

5. Related Work

Adversarial and Certified Defences In recent years, there has been significant research interest in developing robust training methods to enhance the resilience of machine learning models against adversarial attacks. Empirical methods (Szegedy et al., 2014; Madry et al., 2018) achieve strong robustness against various attack methods. However, adversarial training often suffers from high computational costs and can be limited in its ability to provide theoretical guarantees. To address the limitations of adversarial training, researchers have explored certified defenses that provide provable guarantees. These defenses encompass various approaches, including methods based on semidefinite relaxations (Raghunathan et al., 2018), linear relaxations (Zhang et al., 2018; Xu et al., 2020), dual problems (Wong & Kolter, 2018; Wong et al., 2018; Dvijotham et al., 2018), interval

Table 1. Comparison of standard accuracy (Acc), average certified radius (ACR) and accuracy-robustness tradeoff score (ART) for certified training methods on MNIST, CIFAR-10, TinyImageNet test sets. The `CNN7` architecture is used for all datasets. The certificates are obtained using IBP.

Dataset	$\varepsilon_{\max}$	Method	κ	Acc	ACR	ART ↑
MNIST	0.4	Standard	1.0	99.24	0.001	0.35
		CROWN-IBP	0.0	98.20	95.13	96.65
		SABR	0.0	98.75	87.27	92.83
		FastIBP	0.0	97.61	94.69	96.13
		ACERT (ours)	0.0	98.42	95.01	**96.69**
CIFAR-10	$\dfrac{8.8}{255}$	Standard	1.0	91.84	0.0002	0.14
		CROWN-IBP	0.0	45.40	38.59	41.84
		SABR	0.0	52.38	42.45	47.14
		FastIBP	0.0	48.94	41.23	44.91
		FastIBP[†]	0.8	62.84	40.70	50.49
		ACERT (ours)	0.0	62.21	41.80	50.96
		ACERT[†] (ours)	0.25	64.63	41.20	**51.58**
TinyImageNet	$\dfrac{1}{255}$	Standard	1.0	46.09	0.0004	0.13
		SABR	0.0	28.85	19.99	23.96
		FastIBP	0.0	25.08	21.06	22.96
		FastIBP[†]	0.85	30.67	19.90	24.63
		ACERT[†] (ours)	0.0	32.66	21.38	**26.34**

[†]denotes results with the highest ART score among different levels of $\kappa \in [0, 1]$ from Eq. (11).

Table 2. Performance of certified training methods without ε schedulers (denoted as ε_t) on CIFAR-10, $\varepsilon_{\max} = {}^{8.8}/_{255}$. ACERT has the lowest drop in ART score, given as Δ ART, compared to default training.

Method	κ	ε_t	Acc	ACR	ART	Δ ART
FastIBP	0.0	×	33.44	29.64	31.46	-13.45
FastIBP	0.8	×	52.10	37.10	43.97	-6.52
ACERT	0.0	×	52.65	38.95	45.32	**-5.64**
ACERT	0.25	×	54.41	37.96	**45.43**	-6.15

bound propagation (Mirman et al., 2018; Gowal et al., 2019; Shi et al., 2021) and combinations of approaches (Zhang et al., 2020; Balunović & Vechev, 2020; De Palma et al., 2022; Müller et al., 2023). Models, trained using the certified defenses, greatly improve verified robustness, but drastically reduce standard accuracy. Mueller et al. (2021) addresses this issue by proposing a compositional architecture of highly accurate and robust models with a selection mechanism. In our work, we focus on training a single model that achieves optimal levels of both accuracy and robustness, which can potentially be employed by compositional methods.

Adaptive Training The idea of adaptive adversarial training is showing promising results for empirical defenses. Ding et al. (2020); Zhang et al. (2021b); Cheng et al. (2022) propose modifications to adversarial training. Xu et al. (2023) motivate adaptive strategy by studying the dynamics of the decision boundary. While these methods find adaptive perturbation magnitudes heuristically (e.g. by an early stop in PGD, a fixed step of line search, or a boundary attack), our method provides theoretically sound definition and an efficient implementation for finding the exact values. Therefore, connecting theoretical formulations with practical execution allows further research tied to the observed phenomena.

6. Conclusion

We proposed a new method called ACERT that improves the accuracy-robustness tradeoffs of the existing certified training methods. It is based on the key insight that adaptive certification of ℓ_p balls helps to reduce overregularization and improves both standard accuracy and robustness at various perturbation levels. Moreover, we provide a theoretically sound formulation of the certified radius with an efficient GPU-accelerated exact solver for finding it. We introduce a new metric called ART (short for Accuracy-Robustness Tradeoff) that serves as a useful proxy for finding models from the accuracy-robustness Pareto front. Finally, our work establishes a new perspective on optimizing certified robust models, that encourages the development of more robust and accurate machine learning models suitable for deployment in real-world scenarios.

References

Balas, V. E., Roy, S. S., Sharma, D., and Samui, P. *Handbook of deep learning applications*, volume 136. Springer, 2019.

Balunović, M. and Vechev, M. Adversarial training and provable defenses: Bridging the gap. In *International Conference on Learning Representations*, 2020.

Brent, R. P. Algorithms for minimization without derivatives. *Prentice Hall, Englewood Cliffs, NJ*, 1973.

Cheng, M., Lei, Q., Chen, P. Y., Dhillon, I., and Hsieh, C. J. Cat: Customized adversarial training for improved robustness. In *International Joint Conferences on Artificial Intelligence*, pp. 673–679, 2022.

De Palma, A., Bunel, R., Dvijotham, K., Kumar, M. P., and Stanforth, R. Ibp regularization for verified adversarial robustness via branch-and-bound. In *ICML 2022 Workshop on Formal Verification of Machine Learning*, 2022.

Ding, G. W., Sharma, Y., Lui, K. Y. C., and Huang, R. Mma training: Direct input space margin maximization through adversarial training. In *International Conference on Learning Representations*, 2020.

Dvijotham, K., Stanforth, R., Gowal, S., Mann, T., and Kohli, P. A dual approach to scalable verification of deep networks. In *Conference on Uncertainty in Artificial Intelligence*, pp. 162–171, 2018.

Gowal, S., Dvijotham, K. D., Stanforth, R., Bunel, R., Qin, C., Uesato, J., Arandjelovic, R., Mann, T., and Kohli, P. Scalable verified training for provably robust image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4842–4851, 2019.

Jovanović, N., Balunovic, M., Baader, M., and Vechev, M. On the paradox of certified training. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856.

Katz, G., Barrett, C., Dill, D. L., Julian, K., and Kochenderfer, M. J. Reluplex: An efficient smt solver for verifying deep neural networks. In *International Conference on Computer Aided Verification*, pp. 97–117, 2017.

Krizhevsky, A. Learning multiple layers of features from tiny images. *Master's thesis, University of Toronto*, 2009.

Le, Y. and Yang, X. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

LeCun, Y. The mnist database of handwritten digits. *http://yann.lecun.com/exdb/mnist/*, 1998.

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

Mirman, M., Gehr, T., and Vechev, M. Differentiable abstract interpretation for provably robust neural networks. In *International Conference on Machine Learning*, pp. 3578–3586, 2018.

Mueller, M. N., Balunovic, M., and Vechev, M. Certify or predict: Boosting certified robustness with compositional architectures. In *International Conference on Learning Representations*, 2021.

Müller, M. N., Brix, C., Bak, S., Liu, C., and Johnson, T. T. The third international verification of neural networks competition (vnn-comp 2022): Summary and results. *arXiv preprint arXiv:2212.10376*, 2022.

Müller, M. N., Eckert, F., Fischer, M., and Vechev, M. Certified training: Small boxes are all you need. In *International Conference on Learning Representations*, 2023.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035. Curran Associates, Inc., 2019.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

Raghunathan, A., Steinhardt, J., and Liang, P. Certified defenses against adversarial examples. In *International Conference on Learning Representations*, 2018.

Shi, Z., Wang, Y., Zhang, H., Yi, J., and Hsieh, C.-J. Fast certified robust training with short warmup. In *Advances in Neural Information Processing Systems*, volume 34, pp. 18335–18349, 2021.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Wong, E. and Kolter, Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pp. 5286–5295, 2018.

Wong, E., Schmidt, F., Metzen, J. H., and Kolter, J. Z. Scaling provable adversarial defenses. In *Advances in Neural Information Processing Systems*, 2018.

Xu, K., Shi, Z., Zhang, H., Huang, M., Chang, K.-W., Kailkhura, B., Lin, X., and Hsieh, C.-J. Automatic perturbation analysis on general computational graphs. In *International Conference on Machine Learning*, 2020.

Xu, Y., Sun, Y., Goldblum, M., Goldstein, T., and Huang, F. Exploring and exploiting decision boundary dynamics for adversarial robustness. In *International Conference on Learning Representations*, 2023.

Zhang, H., Weng, T.-W., Chen, P.-Y., Hsieh, C.-J., and Daniel, L. Efficient neural network robustness certification with general activation functions. *Advances in Neural Information Processing Systems*, 31, 2018.

Zhang, H., Chen, H., Xiao, C., Gowal, S., Stanforth, R., Li, B., Boning, D., and Hsieh, C.-J. Towards stable and efficient training of verifiably robust neural networks. In *International Conference on Learning Representations*, 2020.

Zhang, J., Lou, Y., Wang, J., Wu, K., Lu, K., and Jia, X. Evaluating adversarial attacks on driving safety in vision-based autonomous vehicles. *IEEE Internet of Things Journal*, 9(5):3443–3456, 2021a.

Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., and Kankanhalli, M. Geometry-aware instance-reweighted adversarial training. In *International Conference on Learning Representations*, 2021b.

A. Theoretical Results

In the following, we provide background for the certification method used, Interval Bound Propagation (Gowal et al., 2019), and give proofs of the theorems from the main paper.

We assume an L-layer feed-forward neural network f_θ for simplicity, denoting $z^{(L)} = f_\theta(x)$ and $u^{(0)} = x$:

$$z^{(\ell)} = W^{(\ell)} u^{(\ell-1)} + b^{(\ell)}, \forall \ell \in \{1, \dots, L\} \tag{14}$$

$$u^{(\ell)} = \sigma^{(\ell)}(z^{(\ell)}), \quad \forall \ell \in \{1, \dots, L-1\} \tag{15}$$

where $\theta = \{W^{(\ell)}, b^{(\ell)}\}_{\ell=1}^{L}$ is a set of affine parameters, $\sigma^{(\ell)}$ are element-wise continuous monotonic non-decreasing activation functions. We use u to represent post-activation neurons and z to represent pre-activation neurons. For $u^{(\ell)}$ and $z^{(\ell)}$, we define element-wise upper and lower bounds as $\underline{u}^{(\ell)} \leq u^{(\ell)} \leq \overline{u}^{(\ell)}$ and $\underline{z}^{(\ell)} \leq z^{(\ell)} \leq \overline{z}^{(\ell)}$.

A.1. Interval Bound Propagation

Consider input bounds $\underline{u}^{(0)} \leq u^{(0)} \leq \overline{u}^{(0)}$, where $u^{(0)} = x$, $\underline{u}^{(0)} = \underline{u}^{(0)}(x, \varepsilon)$, $\overline{u}^{(0)} = \overline{u}^{(0)}(x, \varepsilon)$. Interval bound propagation (IBP) uses simple rules to propagate bounds in a forward manner. For affine layers, it reads:

$$\overline{z}^{(\ell)} = W^{(\ell)} \frac{\overline{u}^{(\ell-1)} + \underline{u}^{(\ell-1)}}{2} + \left| W^{(\ell)} \right| \frac{\overline{u}^{(\ell-1)} - \underline{u}^{(\ell-1)}}{2} + b^{(\ell)}, \tag{16}$$

$$\underline{z}^{(\ell)} = W^{(\ell)} \frac{\overline{u}^{(\ell-1)} + \underline{u}^{(\ell-1)}}{2} - \left| W^{(\ell)} \right| \frac{\overline{u}^{(\ell-1)} - \underline{u}^{(\ell-1)}}{2} + b^{(\ell)}, \tag{17}$$

where $\left| W^{(\ell)} \right|$ is an element-wise absolute value of $W^{(\ell)}$.

For element-wise (monotonic non-decreasing) activation functions $\sigma^{(\ell)}$, the bounds are obtained as

$$\overline{u}^{(\ell)} = \sigma^{(\ell)}(\overline{z}^{(\ell)}), \tag{18}$$

$$\underline{u}^{(\ell)} = \sigma^{(\ell)}(\underline{z}^{(\ell)}). \tag{19}$$

The bound propagating approach can be extended to general computational graphs, following (Xu et al., 2020). We use the implementation of bound propagation from `auto_LiRPA`[1] library, which allows it.

A.2. Certified Robust Radius

Below, we refresh the definition of certified robust radius from the main paper and restate the theorem 3.2 with a given proof.

Definition 3.1 (Certified robust radius). A bounded certified robust radius $\mathcal{E}(x, y; f_\theta; p) \in [0, \varepsilon_{\max}]$ of an input sample $(x, y) \sim \mathcal{D}$ with respect to the classifier f_θ is the maximum radius of an ℓ_p ball $\mathcal{B}(x, \varepsilon)$, not exceeding the predefined (perceptual) limit $\varepsilon_{\max}$, where the classifier is certifiably robust. Formally,

$$\mathcal{E}_{\text{cert}}(x, y; f_\theta, p) = \begin{cases} 0, & \text{if } R_{\text{cert}}(x, 0) \geq 0, \\ \varepsilon_{\max}, & \text{if } R_{\text{cert}}(x, \varepsilon_{\max}) < 0, \\ \sup_{R_{\text{cert}}(x, \varepsilon) < 0} \varepsilon, & \text{otherwise.} \end{cases} \tag{20}$$

Theorem 3.2. *Let $R_{cert}(x, \cdot)$ be continuous and strictly monotonically increasing function. Then the certified robust radius, defined in Def. 3.1, exists and is uniquely defined as*

$$\mathcal{E}_{cert}(x, y; f_\theta, p) = \begin{cases} 0, & \text{if } R_{cert}(x, 0) > 0, \\ \varepsilon_{\max}, & \text{if } R_{cert}(x, \varepsilon_{\max}) < 0, \\ \varepsilon \text{ s.t. } R_{cert}(x, \varepsilon) = 0, & \text{otherwise.} \end{cases} \tag{21}$$

---

[1]We use the implementation of bound propagation from https://github.com/Verified-Intelligence/auto_LiRPA

Proof. By intermediate value theorem, continuity of $R_{\mathrm{cert}}(x, \cdot)$ on a bounded interval $[0, \varepsilon_{\max}]$ leads to the existence of solution in (21). Strict monotonicity ensures the uniqueness of the obtained values of $R_{\mathrm{cert}}(x, \cdot)$, hence leading to a unique solution in (21).

Now we show that the values $\mathcal{E}_{\mathrm{cert}}(x, y; f_\theta, p)$ from (21) are equal to the values in (20) for given function $R_{\mathrm{cert}}(x, \cdot)$. Consider $R_{\mathrm{cert}}(x, 0) = 0$, then in (21), $\mathcal{E}_{\mathrm{cert}}(x, y; f_\theta, p) = 0$ by the third case, and in (20) by the first case. Since for $R_{\mathrm{cert}}(x, \varepsilon_{\max}) < 0$, the definitions are the same, let us consider the certified margin function where $R_{\mathrm{cert}}(x, 0) < 0$, and $R_{\mathrm{cert}}(x, \varepsilon_{\max}) \geq 0$. Due to monotonicity, the set of ε, where $R_{\mathrm{cert}}(x, \varepsilon) < 0$, is a half-open interval $[0, \varepsilon^*)$. And due to continuity, the limit of $\sup_{[0, \varepsilon^*)} \varepsilon$ is reached at ε^*, s.t. $R_{\mathrm{cert}}(x, \varepsilon^*) = 0$. Because of the strict monotonicity assumption, the set of ε where $R_{\mathrm{cert}}(x, \varepsilon) = 0$ is a single point (solution of (21)), which is equal to ε^*, solution of (20). □

As noted in *Remark* 3.3, we observe that in the trained neural networks the assumptions of continuity and strict monotonicity of IBP-certified $R_{\mathrm{cert}}(x, \cdot)$ hold in practice. However, the assumption of strict monotonicity can be relaxed with a simple modification in the definition.

Theorem A.1. *Let $R_{cert}(x, \cdot)$ be continuous and monotonically non-decreasing function. Then the certified robust radius, defined in Def. 3.1, exists and is uniquely defined as*

$$\mathcal{E}_{cert}(x, y; f_\theta, p) = \begin{cases} 0, & \text{if } R_{cert}(x, 0) > 0, \\ \varepsilon_{\max}, & \text{if } R_{cert}(x, \varepsilon_{\max}) < 0, \\ \min \varepsilon \text{ s.t. } R_{cert}(x, \varepsilon) = 0, & \text{otherwise.} \end{cases} \tag{22}$$

Proof. Similar to the proof of Theorem 3.2, let us consider the only different case, where certified margin function $R_{\mathrm{cert}}(x, 0) < 0$, and $R_{\mathrm{cert}}(x, \varepsilon_{\max}) \geq 0$. Again, due to monotonicity, the set of ε, where $R_{\mathrm{cert}}(x, \varepsilon) < 0$, is a half-open interval $[0, \varepsilon^*)$. And due to continuity, the limit of $\sup_{[0, \varepsilon^*)} \varepsilon$ is reached at ε^*, s.t. $R_{\mathrm{cert}}(x, \varepsilon^*) = 0$. On the other hand, since $R_{\mathrm{cert}}(x, \cdot)$ is monotonous, the set of ε s.t. $R_{\mathrm{cert}}(x, \varepsilon) = 0$ is a closed interval $[a, b]$ (including the case of degenerate interval $[a, a]$). Since $a = \min \varepsilon$ s.t. $R_{\mathrm{cert}}(x, \varepsilon) = 0$, and due to monotonicity, $R_{\mathrm{cert}}(x, \varepsilon) < 0$ on $[0, a)$, it follows that $a = \varepsilon^*$. □

Further, we show that the IBP-certified margin function $R_{\mathrm{cert}}(x, \cdot)$ is continuous and monotonic for feed-forward neural networks f_θ with monotonic activation functions, such as `ReLU`, `tanh`, sigmoid, etc.

Theorem A.2. *The IBP-certified margin function $R_{cert}(x, \cdot)$ is continuous and monotonically non-decreasing w.r.t. second argument ε for feed-forward neural networks f_θ, given as (14),(15), and input bounds $\underline{u}^{(0)} = x - \varepsilon \mathbf{1}$ and $\overline{u}^{(0)} = x + \varepsilon \mathbf{1}$, where $\mathbf{1}$ is a vector of ones.*

Proof. Let $\varepsilon_1 > \varepsilon_2$. Then, the corresponding input bounds are

$$\overline{u}_1^{(0)} = x + \varepsilon_1 \mathbf{1}, \quad \underline{u}_1^{(0)} = x - \varepsilon_1 \mathbf{1},$$
$$\overline{u}_2^{(0)} = x + \varepsilon_2 \mathbf{1}, \quad \underline{u}_2^{(0)} = x - \varepsilon_2 \mathbf{1}.$$

Let us define

$$\Delta \overline{u}^{(\ell)} := \overline{u}_1^{(\ell)} - \overline{u}_2^{(\ell)}, \quad \Delta \underline{u}^{(\ell)} := \underline{u}_1^{(\ell)} - \underline{u}_2^{(\ell)}$$
$$\Delta \overline{z}^{(\ell)} := \overline{z}_1^{(\ell)} - \overline{z}_2^{(\ell)}, \quad \Delta \underline{z}^{(\ell)} := \underline{z}_1^{(\ell)} - \underline{z}_2^{(\ell)}.$$

We observe that for the IBP-defined pre-activation bounds

$$\Delta \overline{z}^{(\ell)} = \left(W^{(\ell)} + \left| W^{(\ell)} \right| \right) \frac{\Delta \overline{u}^{(\ell-1)}}{2} + \left(W^{(\ell)} - \left| W^{(\ell)} \right| \right) \frac{\Delta \underline{u}^{(\ell-1)}}{2} \geq 0, \tag{23}$$

$$\Delta \underline{z}^{(\ell)} = \left(W^{(\ell)} - \left| W^{(\ell)} \right| \right) \frac{\Delta \overline{u}^{(\ell-1)}}{2} + \left(W^{(\ell)} + \left| W^{(\ell)} \right| \right) \frac{\Delta \underline{u}^{(\ell-1)}}{2} \leq 0. \tag{24}$$

That is, the upper bounds of pre-activations are monotonically increasing, and the lower bounds are decreasing w.r.t. ε with the preceding post-activation bounds $\overline{u}_1^{(\ell-1)} \geq \overline{u}_2^{(\ell-1)}$, $\underline{u}_1^{(\ell-1)} \leq \underline{u}_2^{(\ell-1)}$. Continuity holds since the transformations

Table 3. Convolutional network architecture from (Shi et al., 2021). Each affine layer (except the last) is followed by batch normalization and ReLU activation. "Conv2D k $w \times h$+s" represents a 2D convolutional layer with k filters of size $w \times h$ and stride s.

CNN7
Conv2D 64 3×3+1
Conv2D 64 3×3+1
Conv2D 128 3×3+2
Conv2D 128 3×3+1
Conv2D 128 3×3+1
Linear 512
Linear out_size

are affine. Next, the monotonicity of post-activation bounds remains the same as of the pre-activation bounds, since the activations $\sigma^{(\ell)}$ are monotonically increasing functions, and continuity follows from continuity of $\sigma^{(\ell)}$. Thus, recursively, the upper bounds of outputs $\overline{z}^{(L)} = \overline{z}^{(L)}(x, \varepsilon)$ are monotonically increasing w.r.t. ε, and the lower bounds $\underline{z}^{(L)} = \underline{z}^{(L)}(x, \varepsilon)$ are monotonically decreasing. Since the certified margin function is a maximum over a difference of increasing and decreasing functions, i.e. $R_{\text{cert}}(x, \varepsilon) = \max_{i \neq y}\{\overline{z}^{(L)}(x, \varepsilon)_i - \underline{z}^{(L)}(x, \varepsilon)_y\}$, it is monotonically increasing (or equivalently, non-decreasing) w.r.t. ε and continuous. □

Remark A.3. The relaxed assumptions from Theorem A.1, and the properties of IBP-certified neural networks from Theorem A.2 allow us to define the certified robust radius for feed-forward neural networks. Moreover, the definition in Theorem A.1 requires only a small modification in root finding algorithms to output the leftmost root, similar to bisect_left. Therefore, it allows us to efficiently find the exact certified radii in practice.

Remark A.4. The assumption of strict monotonicity of $R_{\text{cert}}(x, \cdot)$ holds if the activation functions $\sigma^{(\ell)}$ are strictly increasing, e.g. sigmoid and tanh, and the inequalities in (23) and (24) are strict.

To summarize, the definition of certified robust radius from Theorem 3.2 of the main paper allows simplifying the presentation of algorithms while requiring strict monotonicity property of $R_{\text{cert}}(x, \cdot)$. However, a more general definition from Theorem A.1 does not require strict monotonicity and suits a large class of neural networks.

B. Experimental Results

Hyperparameters To reproduce the results of the baseline method, FastIBP (Shi et al., 2021), we follow the experimental setup from the authors' implementation[2].

For the proposed method, ACERT, we use the same setup as in FastIBP, i.e. 7-layered convolutional network CNN7 (see Table 3), IBP initialization, batch sizes, total numbers of epochs, warm-up schedulers, ReLU stability and bound tightness regularizers, gradient clipping parameters, and weight decays. Find the above hyperparameters in Table 4.

We find that training of our proposed method benefits from a cyclic learning rate scheduler, while FastIBP does not. Therefore, we use OneCycleLR with linear annealing strategy and maximum learning rate of 2e-3 for all datasets. Please find more details of our implementation in the provided code.

Error bars We provide averaged results of our proposed method, ACERT, over multiple runs with different seeds on the CIFAR-10 dataset with $\varepsilon_{\max} = {}^{8.8}/{255}$ in Table 5. The conclusions remain the same as in the main paper, where the (best) single runs are presented for all methods.

Running times The experiments were all run on a single Tesla V-100 GPU. The implementations of both bound propagation and root finding algorithms are within PyTorch (Paszke et al., 2019) framework, allowing GPU acceleration.

We report the running times of the proposed method and its variations in comparison to the baseline method, FastIBP, in Table 6. The total runtime includes both training and validation epochs. For FastIBP, the validation step also computes

---

[2]To reproduce the results of the baseline method, FastIBP, we follow the authors' implementation https://github.com/shizhouxing/Fast-Certified-Robust-Training

Table 4. Experimental setups used in ACERT.

	MNIST	CIFAR-10	TinyImageNet
Model	CNN7	CNN7	CNN7
Initialization	IBP	IBP	IBP
$\varepsilon_{\max}$	0.4	$8.8/255$	$1/255$
Batch size	128	128	256
Num epochs	100	200	200
Warm-up schedules	1-20	1-80	1-80
Reg. coefficient	0.5	0.5	0.2
L_1 weight decay	0.0	0.0	1e-4
Grad. clipping norm	10.0	10.0	10.0
LR scheduler	OneCycle	OneCycle	OneCycle
max_lr	2e-3	2e-3	2e-3

Table 5. Means and standard deviations over 7 runs on CIFAR-10 with $\varepsilon_{\max} = 8.8/255$.

Method	κ	Acc	ACR	ART
ACERT	0.0	61.64±0.34	41.79±0.19	50.74±0.13
ACERT	0.25	64.35±0.16	40.48±0.32	**51.02**±0.24

certified radii. We observe that only two iterations of the root finding algorithm, BrentQ (Brent, 1973), are sufficient to achieve a high ART score. Therefore, making the runtime of our method comparable to FastIBP.

An extreme variation of ACERT with 0 root finding iterations, ACERT-fast (given in the second row of Table 6), employs the following strategy. For misclassified samples, it sets the certified radius to 0, and for correctly classified ones – to $\varepsilon_{\max,t}$. So, in contrast to FastIBP, which sets $\varepsilon_{\max,t}$ for all samples, this variation adapts to the classification result. Since it does not compute the exact certified radii, its runtime is the same as for FastIBP. Interestingly, this zero-iteration adaptation already improves the accuracy and average certified radius over the baseline but does not reach the level of ACERT.

C. Limitations and Discussion

To get a closer look at individual performances, we evaluate certifiably trained models by calculating certified robust accuracies at different levels of ℓ_∞ perturbation radii, given in Figure 3 and Table 7. We observe in Fig. 3 that our method has higher robustness at smaller perturbation intensities and finishes with lower certified accuracy at commonly defined large $\varepsilon_{\text{test}}$ (Table 7). This limitation of our method is related to accuracy-robustness tradeoffs, discussed in Section 4.1 and demonstrated in Fig. 1. To illustrate it, we tune the baseline method, FastIBP, to have higher standard accuracy (at the level of ACERT), and see that its certified accuracy at large $\varepsilon_{\text{test}}$ drops below ACERT's.

In the field of certified training, researchers have mostly focused on achieving high certified accuracy at specific large epsilon values, even if it means sacrificing standard accuracy and certified accuracy at smaller epsilon ranges. However, this often

Table 6. Comparison of variations of the proposed method, ACERT, with different running times against the baseline on CIFAR-10 with $\varepsilon_{\max} = 8.8/255$. All models are trained with the percentage of standard loss $\kappa = 0$.

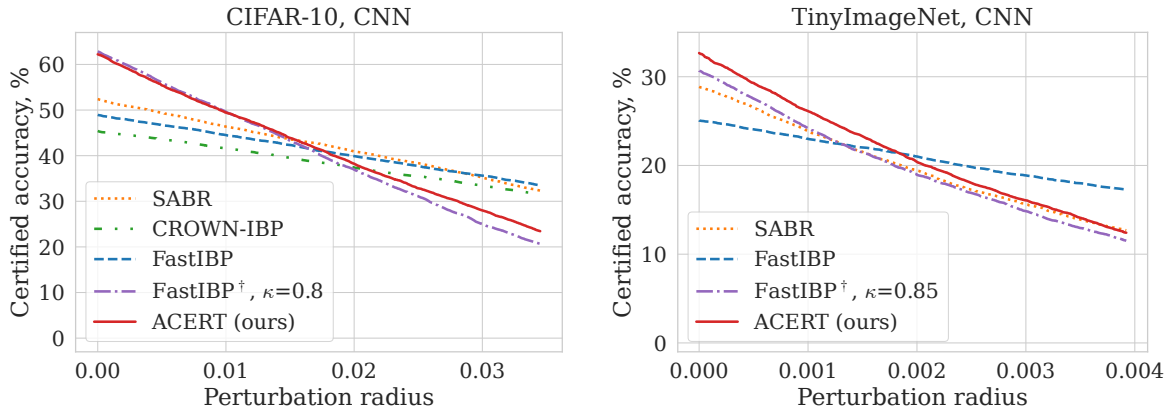| Method	Max BrentQ iters	Acc	ACR	ART	Training step, sec	Total runtime, h
FastIBP	0	48.94	41.23	44.91	0.02	6.1
ACERT-fast	0	54.35	42.37	47.92	0.02	6.1
ACERT	2	61.82	41.66	50.70	0.07	7.2
ACERT	100	61.64	41.79	**50.74**	0.22	10.0

Figure 3. Certified robust accuracies on different levels of ℓ_∞ perturbation radii. The area under a curve is equal to the (unnormalized) average certified radius (ACR) of the method. By default, models with a percentage of standard loss $\kappa = 0$ are demonstrated.

Table 7. Comparison of certified robust accuracies (Cert. Acc.) at commonly used fixed levels of ℓ_∞ perturbation radii $\varepsilon_{\text{test}}$. The certificates are obtained using IBP.

Dataset	$\varepsilon_{\text{test}}$	Method	κ	Acc	Cert. Acc.
MNIST	0.3	FastIBP	0.0	97.61	93.13
		ACERT (ours)	0.0	98.42	93.20
CIFAR-10	$\frac{8}{255}$	FastIBP	0.0	48.94	34.97
		FastIBP†	0.8	62.83	23.71
		ACERT (ours)	0.0	62.21	26.61
		ACERT† (ours)	0.25	64.63	24.41
TinyImageNet	$\frac{1}{255}$	FastIBP	0.0	25.08	17.29
		FastIBP†	0.85	30.67	11.51
		ACERT† (ours)	0.0	32.66	12.42

†denotes results with the highest ART score among different levels of $\kappa \in [0, 1]$.

leads to models that are not practical in real-world scenarios. Moreover, there is a concern that the advancements in the field could be attributed to selectively improving performance points on the same accuracy-robustness curve. Therefore, we argue that the evaluation should focus on optimizing the entire tradeoff. To achieve this, we propose to use the (κ-tuned) ART score (13) as a useful proxy metric, indicating a method's Pareto front.

D. Broader Impact

Since the proposed method bridges the gap between practical usability and safety of the trained models, it can help make real-world AI applications more resilient to adversaries, providing robustness guarantees. While ACERT achieves state-of-the-art accuracy-robustness tradeoffs, it does not necessarily indicate sufficient robustness for safety-critical applications but could give a false sense of security.