Efficient Robustness Evaluation via Constraint Relaxation

Chao Pan[1,2], Yu Wu[1], Ke Tang[1], Qing Li[2], Xin Yao[3*]

[1]Department of Computer Science and Engineering,
Southern University of Science and Technology, Shenzhen 518055, China
[2]The Hong Kong Polytechnic University, Hong Kong, China
[3]School of Data Science, Lingnan University, Hong Kong, China
{11930665, 11930386}@mail.sustech.edu.cn, tangk3@sustech.edu.cn, csqli@comp.polyu.edu.hk, xinyao@ln.edu.hk

Abstract

The study of enhancing model robustness against adversarial examples has become increasingly critical in the security of deep learning, leading to the development of numerous adversarial defense techniques. While these defense methods have shown promise in mitigating the impact of adversarial perturbations, evaluating their effectiveness remains a critical challenge. The recently introduced AutoAttack technique has been recognized as a standardized method for assessing model robustness. However, the computational demands of the AutoAttack method significantly limits its applicability, underscoring the urgent need for efficient evaluation techniques. To address this challenge, we propose a novel and efficient evaluation framework based on strategic constraint relaxation. Our key insight is that temporarily expanding the adversarial perturbation bounds during the attack process can help discover more effective adversarial examples. Based on this insight, we develop the Constraint Relaxation Attack (CR Attack) method, which systematically relaxes and resets perturbation constraints during optimization. Extensive experiments on 105 robust models show that CR Attack outperforms AutoAttack in both attack success rate and efficiency, reducing forward and backward propagation time by 38.3× and 15.9× respectively. Through comprehensive analysis, we validate that the constraint relaxation mechanism is crucial for the method's effectiveness.

Code —
https://github.com/fzjcdt/constraint-relaxation-attack

Introduction

Deep neural networks have achieved remarkable success in various applications. However, these networks are vulnerable to adversarial examples - carefully crafted inputs designed to deceive models into producing erroneous results (Goodfellow, Shlens, and Szegedy 2014; Madry et al. 2017; Szegedy et al. 2013; Gowal et al. 2019; Ning et al. 2024). Even state-of-the-art neural networks can be fooled by slightly perturbed inputs, raising significant security concerns.

To address this vulnerability, researchers have proposed various defense strategies. Among them, adversarial training has emerged as the most effective approach (Salman et al. 2020; Rebuffi et al. 2021; Gowal et al. 2021; Shafahi et al. 2019; Wong, Rice, and Kolter 2020; Wang et al. 2023).

Dataset	Higher ASR ratio	Forward speedup	Backward speedup
CIFAR-10	54 / 60	46.0×	18.6×
CIFAR-100	28 / 30	27.1×	12.1×
ImageNet	13 / 15	31.2×	13.0×
Total	95 / 105	38.3×	15.9×

Table 1: Comparison of our method and AutoAttack on different datasets. Our method outperforms AutoAttack with a higher Attack Success Rate (ASR) on most of the robust models. Moreover, our method exhibits a significantly reduced average forward inference requirement, being only 1/38 of that required by AutoAttack. Similarly, the backward inference requirement is also strikingly lower at 1/16 of AutoAttack on average.

Evaluating the effectiveness of these defense strategies remains challenging, primarily due to the lack of standardized evaluation methods and metrics (Croce and Hein 2020b; Liu et al. 2022). This absence of standardization makes it difficult to compare different defense mechanisms directly and objectively.

AutoAttack has recently emerged as the state-of-the-art evaluation method for model robustness (Croce and Hein 2020b). By incorporating diverse attack techniques and datasets (Croce and Hein 2020a; Andriushchenko et al. 2020), it provides a comprehensive assessment framework. Its effectiveness has led to its adoption as the standard evaluation method on RobustBench (Croce et al. 2020), a leading platform for comparing model robustness.

Despite its effectiveness, AutoAttack's major limitation lies in its computational burden. For instance, evaluating the top two models on RobustBench's CIFAR-10 leaderboard (Krizhevsky, Hinton et al. 2009; Croce et al. 2020; Bartoldson et al. 2024; Amini et al. 2024) requires approximately 114.6 and 177.0 hours respectively on a single NVIDIA Tesla V100 GPU. This computational intensity poses significant challenges for researchers needing quick and efficient robustness evaluations.

While recent efforts have attempted to address this effi-

*Corresponding Author.

ciency challenge (Liu et al. 2022; Liu, Peng, and Tang 2023), the improvements in evaluation speed remain insufficient. There is a pressing need for new methods that can evaluate model robustness more efficiently while maintaining the comprehensiveness of attacks.

In this paper, we propose a novel approach to generating adversarial examples by strategically relaxing input constraints during the attack process. Our key insight is that temporarily expanding the search space beyond the original constraints can help discover promising regions that lead to successful adversarial examples. By alternating between relaxed and original constraints, our method ensures the final adversarial examples still satisfy all original constraints while achieving higher success rates in finding them.

We present the Constraint Relaxation Attack (CR Attack), which implements this insight through an iterative optimization process where constraints are relaxed and reset. Through extensive evaluation on 105 robust models, CR Attack demonstrates superior performance compared to AutoAttack, achieving both higher attack success rates and remarkable computational efficiency.

The contributions of this paper are as follows:

- We propose a novel robustness evaluation framework based on constraint relaxation, which strategically expands and contracts the adversarial perturbation bounds during optimization. This dynamic approach enables more effective exploration of the attack space while ensuring the final adversarial examples satisfy original constraints.
- We develop the Constraint Relaxation Attack (CR Attack) method that demonstrates superior performance over the state-of-the-art AutoAttack benchmark. Our extensive experiments on 105 robust models show that CR Attack achieves higher attack success rates while reducing computational costs by 38.3× in forward propagation and 15.9× in backward propagation.
- Through comprehensive ablation studies and analysis, we demonstrate that the constraint relaxation strategy is fundamental to both the effectiveness and efficiency of our method. We show how this approach enhances exploration through dynamic search space expansion while maintaining efficient exploitation through adaptive refinement.

Related Work

This section reviews the literature on adversarial attacks and defenses, along with standardized benchmarks for evaluating adversarial robustness.

Adversarial Attacks

Deep neural networks are vulnerable to adversarial attacks, where carefully crafted, imperceptible perturbations to input data lead to incorrect predictions (Goodfellow, Shlens, and Szegedy 2014; Madry et al. 2017). The development of adversarial attacks has evolved significantly since Szegedy et al. (Szegedy et al. 2013) introduced the L-BFGS method. Due to L-BFGS's computational inefficiency, Goodfellow et al. (Goodfellow, Shlens, and Szegedy 2014) proposed the Fast Gradient Sign Method (FGSM), which generates adversarial examples using single-step gradients. Building upon FGSM, Projected Gradient Descent (PGD) (Madry et al. 2017) employs iterative attacks and random initialization strategies, becoming a standard approach for adversarial training.

More sophisticated attacks have emerged, such as the Fast Adaptive Boundary Attack (FAB) (Croce and Hein 2020a), which identifies minimal perturbations needed to deceive neural networks, and the MultiTargeted (MT) attack (Gowal et al. 2019), which strengthens attacks by selecting different target classes at each restart. However, these individual attacks often overestimate model robustness (Croce and Hein 2020b).

AutoAttack (Croce and Hein 2020b) addresses these limitations by combining four complementary attacks:

- APGD-CE: A step size-free variant of PGD using cross-entropy loss
- APGD-DLR: An attack based on the DLR loss function
- FAB: Minimizes adversarial perturbation norms
- Square Attack: A query-efficient black-box attack

This ensemble approach provides a more reliable evaluation of model robustness.

Adversarial Defenses

Adversarial Training (AT) (Madry et al. 2017) has emerged as the leading defense strategy, incorporating adversarial examples during model training and consistently outperforming alternatives in defense competitions (Zhang et al. 2019b; Rice, Wong, and Kolter 2020). Recent work has also shown that ensemble approaches can improve model robustness (Zhao et al. 2024). However, AT's high computational cost has spurred the development of more efficient approaches, including methods that reuse gradient information (Shafahi et al. 2019; Zhang et al. 2019a) and single-step training techniques (Liu, Khalil, and Khreishah 2021; Vivek and Babu 2020; Pan, Li, and Yao 2024; Wong, Rice, and Kolter 2020).

A significant challenge in adversarial training is robust overfitting (Rice, Wong, and Kolter 2020), where model robustness deteriorates as training progresses. Research suggests that limited dataset size fundamentally constrains robust generalization (Schmidt et al. 2018). Recent work has shown that expanding training data through external sources (Carmon et al. 2019; Uesato et al. 2019; Rebuffi et al. 2021; Carmon et al. 2019) or synthetic data generation (Gowal et al. 2021; Wang et al. 2023; Rebuffi et al. 2021) can significantly enhance model robustness.

Benchmark for Robustness

RobustBench (Croce et al. 2020) serves as the current standard for evaluating adversarial robustness. This open benchmark maintains an updated ranking of defense methods, evaluating hundreds of models using AutoAttack (Croce and Hein 2020b). Our study utilizes the robust models provided by RobustBench for testing.

Methodology

This section presents the framework for our approach, beginning with fundamental concepts of adversarial attacks and training, followed by our proposed constraint relaxation method for enhancing attack effectiveness.

Preliminaries

In the context of neural networks, an adversarial attack involves deliberately modifying input data to induce misclassification. Consider a neural network $f_{\boldsymbol{\theta}}(\cdot)$ with parameters $\boldsymbol{\theta}$, an input $\boldsymbol{x}$, and its true label y. The objective is to find a perturbation $\boldsymbol{\delta}$ that causes misclassification while maintaining perceptual similarity to the original input. This can be formalized as an optimization problem:

$$\arg\max_{\boldsymbol{\delta}} \{\mathcal{L}(f(\boldsymbol{x}+\boldsymbol{\delta}), y)\}, \tag{1}$$

where $\mathcal{L}(\cdot)$ represents the loss function measuring the disparity between network output and true label, subject to the constraint $\|\boldsymbol{\delta}\|_p \leq \epsilon$ for some small $\epsilon > 0$.

To defend against such attacks, adversarial training incorporates adversarial examples into the model's training process. This approach can be expressed as a min-max optimization problem:

$$\min_{\theta} \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left[\max_{\|\boldsymbol{x'}-\boldsymbol{x}\|_p \leq \epsilon} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x'}), y) \right]. \tag{2}$$

Here, $(\boldsymbol{x}, y) \sim \mathcal{D}$ represents samples from the data distribution, and $\boldsymbol{x'}$ denotes the adversarially perturbed input. The inner maximization seeks the most effective adversarial perturbation within the ϵ-bounded neighborhood, while the outer minimization aims to reduce the model's vulnerability to these adversarial examples.

Constraint Relaxation

Constraint relaxation (CR) is a well-established optimization technique that temporarily loosens problem constraints to facilitate more effective solution search. In the context of adversarial attacks, CR can be particularly powerful as it enables exploration of a broader solution space while ultimately maintaining the required perturbation bounds.

The motivation for applying CR to adversarial attacks stems from several key observations:

1) **Enhanced Initialization**: Relaxed constraints allow for the discovery of promising initial perturbations that, while temporarily violating original constraints, can guide the search toward effective adversarial examples within the valid bounds. This is crucial since initialization quality significantly impacts attack success rates (Madry et al. 2017; Croce and Hein 2020a).

2) **Expanded Search Space**: Traditional adversarial attacks are confined to a strictly bounded search space defined by perturbation constraints (e.g., $\|\boldsymbol{\delta}\|_p \leq \epsilon$). By temporarily relaxing these constraints, the attack can explore previously inaccessible regions that may contain valuable information about the model's decision boundaries.

3) **Defense Exploitation**: Adversarially trained models, as formulated in Eq. 2, are typically optimized to resist perturbations within specific bounds. However, they may be vulnerable to the strategic use of temporarily larger perturbations, even when the final adversarial example respects the original constraints.

The effectiveness of CR in adversarial attacks can be attributed to its ability to overcome local optima that often trap traditional constraint-bounded approaches. By allowing controlled constraint violations during the search process, CR can identify promising search directions that would be impossible to discover within strict bounds, while ensuring the final adversarial examples satisfy all original constraints through a subsequent refinement phase.

This foundation motivates our proposed Constraint Relaxation Attack, detailed in the following section, which systematically implements these insights into a practical attack algorithm.

Algorithm 1: Constraint Relaxation Attack

Require: A classifier $f_{\boldsymbol{\theta}}$ with loss function $\mathcal{L}$; an example $\boldsymbol{x}$ and ground-truth label y; data set category count C; perturbation size ϵ; maximum iterations T; step size decay interval S.

Ensure: An adversarial example $\boldsymbol{x'}$ with $||\boldsymbol{x'} - \boldsymbol{x}||_p \leq \epsilon$.

$\boldsymbol{x'_{pgd}} = \text{PGD}(\boldsymbol{x}, \epsilon)$; {Initialize with PGD perturbation}
$\boldsymbol{p} = f_{\boldsymbol{\theta}}(\boldsymbol{x'_{pgd}})$; {Predict probabilities}
$\boldsymbol{p_y} = 0$; {Ignore the true label to avoid direct attack}
for $c = 1$ **to** $\lfloor \log(C) \rfloor$ **do**
  $\alpha = \epsilon$; {Reset step size for each target label loop}
  $y_t = \arg\max(\boldsymbol{p})$; {Select target label}
  $\boldsymbol{p_{y_t}} = 0$; {Eliminate chosen target label for next iteration}
  $\boldsymbol{x'_0} = \boldsymbol{x}$; {Reset adversarial example to original}
  for $i = 0$ **to** $T - 1$ **do**
    if $i \bmod S = 0$ **then**
      $\alpha = \frac{1}{2}\alpha$; {Halve step size at specified intervals}
    end if
    $\epsilon' = \epsilon$; {Reset perturbation size for each iteration}
    if $i \bmod S < \lfloor \frac{1}{3} \cdot S \rfloor$ **then**
      $\epsilon' = \epsilon + \alpha$; {Relax constraint in first third of decay interval}
    else if $i \bmod S = \lfloor \frac{2}{3} \cdot S \rfloor$ **then**
      $\epsilon' = \epsilon - \alpha$; {Partial restart}
    end if
    Input $\boldsymbol{x'_i}$ to $f_{\boldsymbol{\theta}}$ and obtain the gradient $\boldsymbol{g_i} = \nabla_{\boldsymbol{x'_i}}(\mathcal{L}(\boldsymbol{x'_i}, y) - \mathcal{L}(\boldsymbol{x'_i}, y_t))$; {Compute gradient for loss difference}
    Update $\boldsymbol{x'_{i+1}}$ by applying the sign gradient as $\boldsymbol{x'_{i+1}} = \boldsymbol{x'_i} + \alpha \cdot \text{sign}(\boldsymbol{g_i})$; {Apply signed gradient update}
    $\boldsymbol{x'_{i+1}} = \text{Clip}(\boldsymbol{x'_{i+1}}, \epsilon')$; {Clip to relaxed constraint}
    $\boldsymbol{x'} = \text{Clip}(\boldsymbol{x'_{i+1}}, \epsilon)$; {Ensure perturbation is within original bound}
    if $\boldsymbol{x'}$ is an adversarial example **then**
      return $\boldsymbol{x'}$; {Return if adversarial criteria met}
    end if
  end for
end for

Constraint Relaxation Attack

We propose the Constraint Relaxation Attack (CR Attack), a novel adversarial attack methodology that leverages constraint relaxation to enhance the discovery of adversarial examples. The key innovation lies in its two-phase approach: (1) an exploration phase with relaxed constraints to identify promising perturbation directions, and (2) a refinement phase that ensures the final perturbation satisfies the original constraints.

Building upon the MultiTargeted Attack framework (Gowal et al. 2019), our method systematically explores multiple target labels to increase attack success rates. The algorithm begins with a PGD-initialized perturbation and employs an adaptive constraint relaxation strategy:

1) **Dynamic Constraint Adjustment**: During the first third of each decay interval (a period of S steps), the perturbation bound is temporarily relaxed from ϵ to $\epsilon+\alpha$, allowing for broader exploration:

$$\epsilon' = \begin{cases} \epsilon + \alpha & \text{if } i \bmod S < \lfloor \frac{1}{3} \cdot S \rfloor \\ \epsilon - \alpha & \text{if } i \bmod S = \lfloor \frac{2}{3} \cdot S \rfloor \\ \epsilon & \text{otherwise} \end{cases} \tag{3}$$

This strategy is further enhanced by a partial restart mechanism applied when $i \bmod S = \lfloor \frac{2}{3} \cdot S \rfloor$. At this point, the perturbation magnitude is slightly reduced to $\epsilon - \alpha$, guiding the optimization process back towards the feasible region. This controlled perturbation reset helps to refine the attack direction and escape local optima.

2) **Adaptive Step Size**: The step size α is halved at regular intervals to facilitate fine-grained optimization:

$$\alpha_{\text{new}} = \begin{cases} \frac{1}{2}\alpha & \text{if } i \bmod S = 0 \\ \alpha & \text{otherwise} \end{cases} \tag{4}$$

The attack proceeds by iteratively selecting target labels based on the model's prediction probabilities, excluding the true label to avoid direct attacks. For each target, the algorithm performs gradient-based optimization with the dynamic constraint adjustment mechanism, ensuring that the final adversarial example satisfies $||\boldsymbol{x'} - \boldsymbol{x}||_p \leq \epsilon$.

Justification for the Effectiveness of Constraint Relaxation Attack

The effectiveness of the CR Attack stems from its novel approach to adversarial example generation, fundamentally grounded in the well-established exploration-exploitation trade-off (Črepinšek, Liu, and Mernik 2013). By strategically manipulating the search space through dynamic constraint adjustment, the attack achieves superior performance through two key mechanisms: enhanced exploration via constraint relaxation and focused exploitation through adaptive refinement.

Enhanced Exploration through Dynamic Search Space Expansion The CR Attack's exploration phase systematically expands the adversarial search space by temporarily relaxing the perturbation constraints. This can be formalized as an optimization problem:

$$\max_{\boldsymbol{\delta}} \mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x} + \boldsymbol{\delta}), y) \quad \text{s.t.} \quad ||\boldsymbol{\delta}||_p \leq \epsilon + \alpha, \tag{5}$$

where α represents the relaxation margin. This expansion significantly increases the searchable volume by a factor proportional to:

$$\frac{V(\epsilon + \alpha)}{V(\epsilon)} = (\frac{\epsilon + \alpha}{\epsilon})^n, \tag{6}$$

where n represents the input dimension. This expanded search space allows the attack to:

- Discover adversarial directions that might be inaccessible under strict constraints
- Escape local optima that often trap traditional bounded attacks
- Identify promising regions that can later be refined into valid adversarial examples

Efficient Exploitation through Adaptive Refinement The attack's refinement phase employs an adaptive mechanism that systematically narrows the search space while preserving promising perturbation directions. This is achieved through:

1) **Progressive Constraint Tightening**: The perturbation bound is periodically reduced according to:

$$\epsilon'_t = \epsilon + \alpha_t, \quad \text{where} \quad \alpha_t = \alpha_0 \cdot 2^{-\lfloor t/S \rfloor} \tag{7}$$

2) **Targeted Refinement**: The attack focuses computational resources on the most promising regions identified during exploration, ensuring efficient convergence to valid adversarial examples.

This dual-phase approach enables the CR Attack to maintain a dynamic balance between exploration and exploitation, resulting in more effective adversarial example generation compared to traditional single-phase attacks.

Parameter-Free Nature of CR

While our Constraint Relaxation (CR) method introduces a structured approach to perturbation generation, it inherits and maintains the parameter-free philosophy that makes AutoAttack (AA) practical for robustness evaluation. The key parameters in CR are carefully designed to be fixed constants that directly correspond to their counterparts in AA's components, eliminating the need for case-by-case tuning across different models or datasets.

Specifically, we establish the following parameter correspondences: 1) The α scaling factor in CR serves an analogous function to AutoPGD's step size adjustment, controlling the magnitude of perturbation updates 2) The decay interval in CR parallels AutoPGD's convergence parameters, governing the optimization dynamics 3) The threshold values in CR correspond to the checkpoints in AutoPGD's optimization process, providing similar progress monitoring capabilities

These parameters are universally applied with fixed values across all evaluations, maintaining the standardized nature of the assessment process. This design choice preserves AA's valuable characteristic of being parameter-free while simultaneously improving computational efficiency.

Experiments

In this section, we describe the data sets used, the experimental setup and the analysis of the experimental results.

Experiment Setup

We conduct our experiments using three standard datasets from the RobustBench framework: CIFAR-10, CIFAR-100, and ImageNet. For ImageNet, following RobustBench protocol, we utilize the first 5000 images from the validation subset to ensure standardized comparison.

The problem was posited as an adversarial attack using the l_∞ norm. To ensure appropriateness and impartiality of the analysis, we drew from all viable models available in the RobustBench (Croce et al. 2020) without any bias towards specific selections. Following RobustBench guidelines, we set the perturbation size to 8/255 for CIFAR-10 and CIFAR-100, and 4/255 for ImageNet.

We set the maximum number of iterations (T) to 150 and the decay steps (S) to 30, using margin loss as the loss function. The number of restarts was fixed at 5. To accelerate the process, from the second restart onwards, we only sampled instances with a margin loss less than 0.05 from the previous attempt. To ensure reliability, we conducted each experiment 3 times and reported the average results.

We compare our algorithm with AutoAttack (Croce and Hein 2020b), a widely used method for evaluating model robustness. The evaluation metric used is the robust accuracy, defined as the accuracy maintained by the model under adversarial attacks. A lower robust accuracy indicates a higher Attack Success Rate (ASR), where ASR represents the fraction of test samples for which the model's predicted label differs from the true label under adversarial perturbations.

Results and Analysis

Due to space limitations, we present results for a representative subset of the 105 evaluated robust models. Table 2 showcases the top-performing models for each dataset: 30 for CIFAR-10, 15 for CIFAR-100, and 5 for ImageNet, representing 50% of the available robust models per dataset.

Our experimental results in Table 2 demonstrate that CR attack consistently outperforms AutoAttack, achieving lower robustness accuracy across most models. Notably, CR attack surpasses the best-known robust accuracy benchmarks recorded in RobustBench. Beyond effectiveness, our method demonstrates significant computational advantages over AutoAttack, achieving 38× speedup in forward propagation and 16× in backward propagation. These results validate CR attack as both an effective and efficient approach for generating adversarial examples and evaluating model robustness.

Analysis of Attack Consistency

To verify the consistency of our method's performance advantage and eliminate potential random factors, we conducted additional experiments comparing CR with AutoAttack (AA). We focused on 18 specific samples where CR initially succeeded while AA failed, using the robust model from (Cui et al. 2023). We ran AA with 100 different random seeds on these samples to thoroughly examine its performance variability.

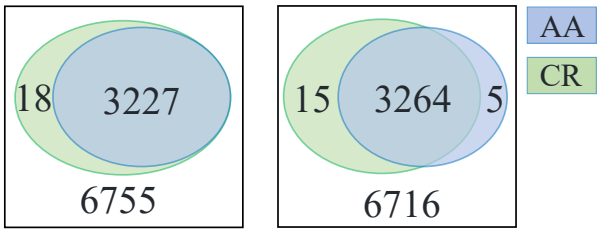

Figure 1: Complementarity analysis of adversarial examples generated by CR and AutoAttack (AA) on CIFAR-10 test set, using two state-of-the-art WRN-28-10 models from RobustBench (Cui et al. 2023) [left] and (Wang et al. 2023) [right].

The results demonstrate that CR's success is indeed systematic rather than due to chance. For 10 out of 18 samples, AA consistently failed across all 100 random seeds, indicating these cases represent genuine limitations of AA that CR successfully addresses. The remaining samples showed varying but generally low success rates for AA (ranging from 5% to 47%), with an average success rate of 12.56% across all samples.

These findings suggest that while the overall performance difference between CR and AA may be modest, it is consistent and statistically significant. This supports our assertion that CR offers reliable improvements in certain challenging cases where AA struggles, while maintaining competitive performance across all test cases. The results can be interpreted as CR complementing rather than completely outperforming AA, offering a valuable alternative approach for robustness evaluation.

Complementarity Analysis

To better understand the relationship between CR and AutoAttack (Croce and Hein 2020b), we conducted a complementarity analysis on CIFAR-10 test dataset using the two best-performing WideResNet-28-10 models from RobustBench (Cui et al. 2023; Wang et al. 2023). As shown in Fig. 1, there is a substantial overlap between the adversarial examples found by CR and AA, indicating that the two methods largely identify similar vulnerabilities in the models.

For both models, CR successfully finds almost all adversarial examples that AA can discover. Specifically, out of all samples that AA successfully attacks, CR fails on only a negligible portion (0 and 5 samples for the two models, respectively). Moreover, CR demonstrates superior attack capability by identifying additional adversarial examples that AA fails to find (18 and 15 samples for the two models, respectively). This suggests that while CR and AA share similar attack mechanisms, CR's constraint relaxation strategy enables it to explore a broader attack surface and uncover

#	Model	Best Known	AA Acc. (↓)	Avg Forward	Avg Backward	CR Acc. (↓)	Avg Forward	Avg Backward
CIFAR-10, l_∞, ϵ=8/255								
1	(Bartoldson et al. 2024)	73.71	73.71	5810	1431	**73.59**	108 (53.8×)	65 (22.0×)
2	(Amini et al. 2024)	72.08	72.08	5674	1399	**71.85**	104 (54.6×)	62 (22.6×)
3	(Bartoldson et al. 2024)	71.59	71.59	5647	1393	**71.42**	107 (52.8×)	64 (21.8×)
4	(Peng et al. 2023)	71.07	71.07	5637	1390	**70.99**	104 (54.2×)	63 (22.1×)
5	(Wang et al. 2023)	70.69	70.69	5541	1370	**70.56**	119 (46.6×)	73 (18.8×)
6	(Cui et al. 2023)	67.73	67.73	5344	1322	**67.55**	104 (51.4×)	62 (21.3×)
7	(Bai et al. 2023)	68.06	68.06	5459	1354	**67.35**	145 (37.7×)	92 (14.7×)
8	(Wang et al. 2023)	67.31	67.31	5338	1322	**67.21**	118 (45.2×)	72 (18.4×)
9	(Rebuffi et al. 2021)	66.56	66.58	4998	1243	**66.51**	111 (45.0×)	67 (18.6×)
10	(Gowal et al. 2021)	66.10	66.11	5148	1275	**66.08**	109 (47.2×)	66 (19.3×)
11	(Gowal et al. 2020)	65.87	65.88	5055	1253	**65.74**	117 (43.2×)	71 (17.6×)
12	(Huang et al. 2022)	65.79	65.79	5210	1289	**65.71**	113 (46.1×)	69 (18.7×)
13	(Rebuffi et al. 2021)	64.58	64.64	4977	1234	**64.47**	116 (42.9×)	71 (17.4×)
14	(Rebuffi et al. 2021)	64.20	64.25	4915	1220	**64.18**	116 (42.4×)	71 (17.2×)
15	(Gowal et al. 2021)	63.38	63.44	4928	1221	**63.36**	108 (45.6×)	65 (18.8×)
16	(Pang et al. 2022)	63.35	63.35	4868	1209	**63.28**	115 (42.3×)	70 (17.3×)
17	(Rade et al. 2021)	62.83	62.83	4793	1193	**62.66**	111 (43.2×)	68 (17.5×)
18	(Sehwag et al. 2021)	62.79	62.79	4758	1181	**62.53**	110 (43.3×)	66 (17.9×)
19	(Gowal et al. 2020)	62.76	62.80	4729	1176	**62.71**	121 (39.1×)	74 (15.9×)
20	(Huang et al. 2021)	62.50	62.54	4824	1200	**62.49**	106 (45.5×)	64 (18.8×)
21	(Huang et al. 2021)	**61.56**	**61.56**	4665	1160	61.59	102 (45.7×)	61 (19.0×)
22	(Dai, Mahloujifar, and Mittal 2022)	61.55	61.55	4776	1188	**61.45**	108 (44.2×)	66 (18.0×)
23	(Pang et al. 2022)	61.04	61.04	4774	1186	**60.91**	118 (40.5×)	72 (16.5×)
24	(Rade et al. 2021)	60.97	60.97	4727	1174	**60.80**	109 (43.4×)	66 (17.8×)
25	(Rebuffi et al. 2021)	60.73	60.75	4728	1175	**60.68**	117 (40.4×)	72 (16.3×)
26	(Sridhar et al. 2022)	60.41	60.41	4664	1159	**60.32**	132 (35.3×)	82 (14.1×)
27	(Sehwag et al. 2021)	60.27	60.27	4648	1155	**60.23**	104 (44.7×)	62 (18.6×)
28	(Wu, Xia, and Wang 2020)	60.04	60.04	4667	1162	**59.98**	120 (38.9×)	74 (15.7×)
29	(Sridhar et al. 2022)	59.66	59.66	4668	1163	**59.57**	109 (42.8×)	66 (17.6×)
30	(Zhang et al. 2020)	59.64	59.64	4641	1160	**59.12**	158 (29.4×)	100 (11.6×)
CIFAR-100, l_∞, ϵ=8/255								
1	(Wang et al. 2023)	42.67	42.67	3351	844	**42.57**	119 (28.2×)	71 (11.9×)
2	(Cui et al. 2023)	39.18	39.18	3009	764	**39.10**	96 (31.3×)	54 (14.2×)
3	(Wang et al. 2023)	38.83	38.83	2959	749	**38.67**	116 (25.5×)	69 (10.9×)
4	(Bai et al. 2023)	38.72	38.72	3082	789	**38.63**	94 (32.8×)	54 (14.6×)
5	(Gowal et al. 2020)	36.88	36.88	2695	686	**36.87**	95 (28.4×)	54 (12.7×)
6	(Debenedetti et al. 2022)	35.08	35.08	2665	680	**34.96**	99 (26.9×)	57 (11.9×)
7	(Bai et al. 2023)	35.15	35.15	2679	693	**34.80**	85 (31.5×)	47 (14.7×)
8	(Rebuffi et al. 2021)	34.64	34.64	2594	658	**34.56**	99 (26.2×)	57 (11.5×)
9	(Debenedetti et al. 2022)	34.21	34.21	2724	692	**34.09**	103 (26.4×)	61 (11.3×)
10	(Pang et al. 2022)	33.05	33.05	2595	659	**32.97**	96 (27.0×)	55 (12.0×)
11	(Cui et al. 2023)	32.52	32.52	2597	661	**32.46**	81 (32.1×)	45 (14.7×)
12	(Debenedetti et al. 2022)	32.19	32.19	2522	645	**32.10**	104 (24.2×)	61 (10.6×)
13	(Cui et al. 2023)	31.65	31.65	2518	640	**31.62**	82 (30.7×)	45 (14.2×)
14	(Chen and Lee 2024)	31.13	31.13	2456	626	**31.12**	85 (28.9×)	47 (13.3×)
15	(Rebuffi et al. 2021)	32.06	32.06	2470	628	**31.95**	95 (26.0×)	54 (11.6×)
ImageNet, l_∞, ϵ=4/255								
1	(Amini et al. 2024)	**59.64**	**59.64**	5075	1248	59.70	165 (30.8×)	99 (12.6×)
2	(Liu et al. 2024)	59.56	59.56	4918	1212	**59.46**	165 (29.8×)	99 (12.2×)
3	(Liu et al. 2024)	**58.48**	**58.48**	5013	1235	58.50	161 (31.1×)	96 (12.9×)
4	(Singh, Croce, and Hein 2024)	57.70	57.70	5106	1257	**57.62**	158 (32.3×)	94 (13.4×)
5	(Liu et al. 2024)	56.16	56.16	4795	1183	**56.10**	156 (30.7×)	93 (12.7×)

Table 2: Comparative analysis of robust accuracy (%) using AutoAttack (AA) and CR attack for different defense strategies.

Method	Acc. (%) ↓	Forward	Backward
Model from (Cui et al. 2023)			
AA	67.73	5,344	1,322
-APGD-CE	70.36	378	378
-APGD-DLR	67.78	649	642
-FAB	68.26	1,299	639
-Square	74.05	3,873	0
CR (Ours)	**67.55**	104	62
Model from (Wang et al. 2023)			
AA	67.31	5,338	1,322
-APGD-CE	70.05	378	377
-APGD-DLR	67.35	646	639
-FAB	67.81	1,292	636
-Square	73.73	3,862	0
CR (Ours)	**67.21**	118	72

Table 3: Comparison of robust accuracy (%) using CR, AutoAttack (AA) and its components on WRN-28-10 models.

Model	AA	Modified AA	CR
(Cui et al. 2023)	67.73	67.70	**67.55**
(Wang et al. 2023)	67.31	67.26	**67.21**

Table 4: Robust accuracy (%) comparison on WRN-28-10 models under different attack variants

vulnerabilities that AA might miss.

This complementarity analysis not only validates CR's effectiveness but also indicates that it could serve as a more comprehensive evaluation tool for model robustness compared to AutoAttack.

Comparison with Individual Attacks

To provide a more nuanced analysis of our method's efficiency, we compare CR with individual attacks that constitute AutoAttack: APGD-CE, APGD-DLR (Croce and Hein 2020b), FAB (Croce and Hein 2020a), and SQUARE (Andriushchenko et al. 2020). Table 3 presents the robust accuracy and computational costs for each attack method on two state-of-the-art WRN-28-10 models.

The results demonstrate that CR achieves better robustness evaluation than AA while significantly reducing computational overhead. Notably, when compared to APGD-CE, which is the most efficient component in AA, CR demonstrates a 3.4× speedup in forward propagation and 5.6× in backward propagation while delivering superior performance.

Integration with AutoAttack

To investigate the potential of combining our Constraint Relaxation (CR) method with existing attack frameworks, we integrated our approach into AutoAttack's AutoPGD component. We evaluated this integration on two state-of-the-art WRN-28-10 models from RobustBench (CIFAR10).

Table 4 presents the comparative results across three attack variants: the original AutoAttack (AA), CR-modified AutoAttack (Modified AA), and our standalone CR attack.

#	Model	CR	w/o CR
CIFAR-10			
1	(Bartoldson et al. 2024)	**73.59**	73.71
2	(Amini et al. 2024)	**71.85**	72.12
3	(Bartoldson et al. 2024)	**71.42**	71.61
CIFAR-100			
1	(Wang et al. 2023)	**42.57**	42.76
2	(Cui et al. 2023)	**39.13**	39.27
3	(Wang et al. 2023)	**38.67**	38.85

Table 5: Robust accuracy (%) under different defensive strategies with and without constraint relaxation. Bold indicates lower robust accuracy.

The results indicate that while CR integration yields modest improvements, the effectiveness of constraint relaxation appears to be dependent on the specific optimization strategy of the base attack algorithm. This suggests that optimal integration of CR into existing attack frameworks may require attack-specific adaptations. These findings highlight opportunities for future research in combining attack strategies while preserving their individual strengths.

Ablation Experiment

We delved deeper into the function of the constraint relaxation module in the context of the CR attack. Specifically, we conducted an experiment wherein the constraint relaxation module was eliminated from the CR Attack, while ensuring that all parameters and settings remained unchanged.

We present the attack success rates in Table 5 for different defense strategies (with top-3 robust accuracy on various datasets), comparing optimization with and without altering constraints. Empirical findings reveal that when constraints are not altered during optimization, the effectiveness of our attack algorithm experiences a slight decrease across all tested models. This suggests that altering constraints during optimization contributes meaningfully to the algorithm's overall performance.

Conclusions

In this paper, we introduce a novel and efficient approach to adversarial robustness evaluation through constraint relaxation. Our proposed CR Attack method strategically expands the search space during optimization while ensuring final adversarial examples satisfy original constraints, addressing the critical challenge of computational efficiency in robustness evaluation. Through comprehensive experiments on 105 robust models from RobustBench, we demonstrate that CR Attack not only achieves superior attack success rates compared to the state-of-the-art AutoAttack but also delivers remarkable efficiency improvements. These results, supported by ablation studies, establish CR Attack as a practical and effective tool for evaluating neural network robustness, particularly valuable for researchers and practitioners in safety-critical applications where both accuracy and efficiency are paramount.

Acknowledgments

We extend our sincere thanks to the anonymous reviewers, whose detailed and thoughtful feedback led to substantial improvements in the manuscript. This work was supported by the National Natural Science Foundation of China under Grant 62250710682, Guangdong Provincial Key Laboratory under Grant 2020B121201001, and the Program for Guangdong Introducing Innovative and Entrepreneurial Teams under Grant 2017ZT07X386. The work described in this paper was (partially) supported by a grant from HK RGC Theme-based Research Scheme (PolyU No.: T43-513/23-N).

References

Amini, S.; Teymoorianfard, M.; Ma, S.; and Houmansadr, A. 2024. MeanSparse: Post-training robustness enhancement through mean-centered feature sparsification. *arXiv preprint arXiv:2406.05927*.

Andriushchenko, M.; Croce, F.; Flammarion, N.; and Hein, M. 2020. Square attack: A query-efficient black-box adversarial attack via random search. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII*, 484–501. Springer.

Bai, Y.; Anderson, B. G.; Kim, A.; and Sojoudi, S. 2023. Improving the accuracy-robustness trade-off of classifiers via adaptive smoothing. *arXiv preprint arXiv:2301.12554*.

Bartoldson, B. R.; Diffenderfer, J.; Parasyris, K.; and Kailkhura, B. 2024. Adversarial robustness limits via scaling-law and human-alignment studies. *arXiv preprint arXiv:2404.09349*.

Carmon, Y.; Raghunathan, A.; Schmidt, L.; Duchi, J. C.; and Liang, P. S. 2019. Unlabeled data improves adversarial robustness. *Advances in Neural Information Processing Systems*, 32.

Chen, E.-C.; and Lee, C.-R. 2024. Data filtering for efficient adversarial training. *Pattern Recognition*, 151: 110394.

Črepinšek, M.; Liu, S.-H.; and Mernik, M. 2013. Exploration and exploitation in evolutionary algorithms: A survey. *ACM Computing Surveys (CSUR)*, 45(3): 1–33.

Croce, F.; Andriushchenko, M.; Sehwag, V.; Debenedetti, E.; Flammarion, N.; Chiang, M.; Mittal, P.; and Hein, M. 2020. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*.

Croce, F.; and Hein, M. 2020a. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*, 2196–2205. PMLR.

Croce, F.; and Hein, M. 2020b. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning*, 2206–2216. PMLR.

Cui, J.; Tian, Z.; Zhong, Z.; Qi, X.; Yu, B.; and Zhang, H. 2023. Decoupled kullback-leibler divergence loss. *arXiv preprint arXiv:2305.13948*.

Dai, S.; Mahloujifar, S.; and Mittal, P. 2022. Parameterizing activation functions for adversarial robustness. In *2022 IEEE Security and Privacy Workshops (SPW)*, 80–87. IEEE.

Debenedetti, E.; Sehwag, V.; Mittal, P.; and ". 2022. A light recipe to train robust vision transformers. *arXiv preprint arXiv:2209.07399*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Gowal, S.; Qin, C.; Uesato, J.; Mann, T.; and Kohli, P. 2020. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*.

Gowal, S.; Rebuffi, S.-A.; Wiles, O.; Stimberg, F.; Calian, D. A.; and Mann, T. A. 2021. Improving robustness using generated data. *Advances in Neural Information Processing Systems*, 34: 4218–4233.

Gowal, S.; Uesato, J.; Qin, C.; Huang, P.-S.; Mann, T.; and Kohli, P. 2019. An alternative surrogate loss for pgd-based adversarial testing. *arXiv preprint arXiv:1910.09338*.

Huang, H.; Wang, Y.; Erfani, S.; Gu, Q.; Bailey, J.; and Ma, X. 2021. Exploring architectural ingredients of adversarially robust deep neural networks. *Advances in Neural Information Processing Systems*, 34: 5545–5559.

Huang, S.; Lu, Z.; Deb, K.; and Boddeti, V. N. 2022. Revisiting residual networks for adversarial robustness: An architectural perspective. *arXiv preprint arXiv:2212.11005*.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Liu, C.; Dong, Y.; Xiang, W.; Yang, X.; Su, H.; Zhu, J.; Chen, Y.; He, Y.; Xue, H.; and Zheng, S. 2024. A comprehensive study on robustness of image classification models: Benchmarking and rethinking. *International Journal of Computer Vision*, 1–23.

Liu, G.; Khalil, I.; and Khreishah, A. 2021. Using single-step adversarial training to defend iterative adversarial examples. In *Proceedings of the Eleventh ACM Conference on Data and Application Security and Privacy*, 17–27.

Liu, S.; Peng, F.; and Tang, K. 2023. Reliable robustness evaluation via automatically constructed attack ensembles. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 8852–8860.

Liu, Y.; Cheng, Y.; Gao, L.; Liu, X.; Zhang, Q.; and Song, J. 2022. Practical evaluation of adversarial robustness via adaptive auto attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15105–15114.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Ning, L.-b.; Dai, Z.; Fan, W.; Su, J.; Pan, C.; Wang, L.; and Li, Q. 2024. Joint universal adversarial perturbations with interpretations. *arXiv preprint arXiv:2408.01715*.

Pan, C.; Li, Q.; and Yao, X. 2024. Adversarial initialization with universal adversarial perturbation: A new approach to fast adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 21501–21509.

Pang, T.; Lin, M.; Yang, X.; Zhu, J.; and Yan, S. 2022. Robustness and accuracy could be reconcilable by (proper) definition. *arXiv preprint arXiv:2202.10103*.

Peng, S.; Xu, W.; Cornelius, C.; Hull, M.; Li, K.; Duggal, R.; Phute, M.; Martin, J.; and Chau, D. H. 2023. Robust principles: Architectural design principles for adversarially robust cnns. *arXiv preprint arXiv:2308.16258*.

Rade, R.; Moosavi, D.; Seyed, M.; and ". 2021. Helper-based adversarial training: Reducing excessive margin to achieve a better accuracy vs. robustness trade-off. In *ICML 2021 Workshop on Adversarial Machine Learning*.

Rebuffi, S.-A.; Gowal, S.; Calian, D. A.; Stimberg, F.; Wiles, O.; and Mann, T. 2021. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*.

Rice, L.; Wong, E.; and Kolter, Z. 2020. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, 8093–8104. PMLR.

Salman, H.; Ilyas, A.; Engstrom, L.; Kapoor, A.; and Madry, A. 2020. Do adversarially robust imagenet models transfer better? *Advances in Neural Information Processing Systems*, 33: 3533–3545.

Schmidt, L.; Santurkar, S.; Tsipras, D.; Talwar, K.; and Madry, A. 2018. Adversarially robust generalization requires more data. *Advances in neural information processing systems*, 31.

Sehwag, V.; Mahloujifar, S.; Handina, T.; Dai, S.; Xiang, C.; Chiang, M.; and Mittal, P. 2021. Robust learning meets generative models: Can proxy distributions improve adversarial robustness? *arXiv preprint arXiv:2104.09425*.

Shafahi, A.; Najibi, M.; Ghiasi, M. A.; Xu, Z.; Dickerson, J.; Studer, C.; Davis, L. S.; Taylor, G.; and Goldstein, T. 2019. Adversarial training for free! *Advances in Neural Information Processing Systems*, 32.

Singh, N. D.; Croce, F.; and Hein, M. 2024. Revisiting adversarial training for imagenet: Architectures, training and generalization across threat models. *Advances in Neural Information Processing Systems*, 36.

Sridhar, K.; Sokolsky, O.; Lee, I.; and Weimer, J. 2022. Improving neural network robustness via persistency of excitation. In *2022 American Control Conference (ACC)*, 1521–1526. IEEE.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Uesato, J.; Alayrac, J.-B.; Huang, P.-S.; Stanforth, R.; Fawzi, A.; and Kohli, P. 2019. Are labels required for improving adversarial robustness? *arXiv preprint arXiv:1905.13725*.

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

Wang, Z.; Pang, T.; Du, C.; Lin, M.; Liu, W.; and Yan, S. 2023. Better diffusion models further improve adversarial training. *arXiv preprint arXiv:2302.04638*.

Wong, E.; Rice, L.; and Kolter, J. Z. 2020. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*.

Wu, D.; Xia, S.-T.; and Wang, Y. 2020. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33: 2958–2969.

Zhang, D.; Zhang, T.; Lu, Y.; Zhu, Z.; and Dong, B. 2019a. You only propagate once: Accelerating adversarial training via maximal principle. *Advances in Neural Information Processing Systems*, 32.

Zhang, H.; Yu, Y.; Jiao, J.; Xing, E.; El Ghaoui, L.; and Jordan, M. 2019b. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, 7472–7482. PMLR.

Zhang, J.; Zhu, J.; Niu, G.; Han, B.; Sugiyama, M.; and Kankanhalli, M. 2020. Geometry-aware instance-reweighted adversarial training. *arXiv preprint arXiv:2010.01736*.

Zhao, Y.; Huang, W.; Liu, W.; and Yao, X. 2024. Negatively correlated ensemble against transfer adversarial attacks. *Pattern Recognition*, 111155.