SA-FGSM: A Simulated Annealing-Enhanced Hybrid White-Box Adversarial Attack Framework

Djawhara Benchaira, Foudil Cherif
LESIA Laboratory, Department of Computer Science, University of Biskra, Biskra 07000, Algeria
E-mail: djawhara.benchaira@univ-biskra.dz, cherif.foudil@univ-biskra.dz

Keywords: Adversarial Attacks, Adversarial Robustness, Deep Learning Security, Simulated Annealing, Metaheuristic Optimization, White-Box Attacks, Robustness Evaluation

Received: August 18, 2025

The reliable evaluation of adversarial defenses is a critical challenge in deep learning security, often hindered by evaluation methods, such as Projected Gradient Descent (PGD), that can fail by getting trapped in local optima. This limitation can lead to a significant overestimation of a model's true robustness. In this work, we introduce the Simulated Annealing-Fast Gradient Sign Method (SA-FGSM), a novel two-phase hybrid attack designed to overcome this specific weakness. SA-FGSM first employs Simulated Annealing to perform a global, stochastic exploration of the perturbation space to find promising attack regions, followed by a gradient-based step for finalization. We conduct a comprehensive evaluation on CIFAR-10 and CIFAR-100 against state-of-the-art adversarially trained models (ResNet-18 and WideResNet-28-10), showing that SA-FGSM achieves a mean attack success rate of 83.6% compared to 51.9% for a suite of strong baselines including FGSM, MI-FGSM, PGD, and APGD. Furthermore, we demonstrate that SA-FGSM finds qualitatively superior perturbations, evidenced by a statistically significant reduction in both average ℓ_2 norm and perceptual distortion as measured by LPIPS (Learned Perceptual Image Patch Similarity), achieving 58.3% lower perceptual distance than gradient-based baselines. Analysis of the proposed attack variants identifies SA-FGSM-Swift as a particularly compelling option, offering state-of-the-art success rates at a fraction of the computational cost of stronger baselines. Our findings suggest that the robustness of even top-tier defenses may be overestimated and highlight the necessity of incorporating global search heuristics into standard evaluation protocols.

Povzetek: Za večjo varnost globokega učenja SA-FGSM združi simulirano ohlajanje in gradientni korak. S pomočjo nasprotniških napadov realneje oceni robustnosti modelov.

1 Introduction

Deep Neural Networks (DNNs) have become the state-of-the-art technology for a wide range of applications, from image classification to autonomous navigation. However, their widespread deployment is critically hindered by a fundamental vulnerability to adversarial examples: meticulously crafted, often imperceptible perturbations to inputs that cause the model to produce an incorrect output [28]. The existence of such examples necessitates the development of robust defense mechanisms and, just as critically, rigorous and reliable methods for their evaluation.

The security and reliability of a given defense are contingent on the strength of the attacks used to evaluate it. Current evaluation benchmarks are dominated by gradient-based white-box attacks, such as the Fast Gradient Sign Method (FGSM) [18], its iterative variant, Projected Gradient Descent (PGD) [24], and adaptive variants like Auto-PGD (APGD) [13]. While computationally efficient, these hill-climbing methods are fundamentally limited by the complex and non-convex nature of DNN loss landscapes, becoming trapped in poor local optima and failing to find adversarial examples that may exist elsewhere in the search space. This creates a false sense of security by systematically overestimating the model's true robustness.

To address this critical limitation, we introduce Simulated Annealing-Fast Gradient Sign Method (SA-FGSM), a novel hybrid adversarial attack that transcends purely gradient-based approaches. SA-FGSM operates through a strategic two-phase process: first, Simulated Annealing performs a global, stochastic search to identify promising perturbation regions while probabilistically escaping local minima through its temperature-guided acceptance mechanism. Second, a targeted gradient-based finalization step leverages the efficiency of gradient information to craft the final adversarial example from this superior starting position. To systematically explore the trade-offs between search intensity and speed, we introduce and analyze four distinct variants of the algorithm—Core, Swift, Thorough, and Adaptive—and identify SA-FGSM-Swift as a particularly compelling method. It offers state-of-the-art attack success rates at a fraction of the computational cost of other high-performing attacks, as quantified by our newly defined Efficiency Score metric.

Our evaluation targets state-of-the-art adversarially robust models from the RobustBench library [13], specifically the `Rebuffi2021Fixing_R18_cutmix_ddpm` and `Rebuffi2021Fixing_28_10_cutmix_ddpm` models [26], ensuring our assessment reflects genuine advances against modern defenses rather than improvements over weak baselines.

Through comprehensive and statistically rigorous evaluation, we demonstrate that SA-FGSM significantly outperforms strong baseline attacks. Our method achieves a mean attack success rate of 83.6%, a 31.7 percentage point improvement over the baseline average—a result that is highly statistically significant ($p < 0.0001$). Beyond higher success rates, SA-FGSM generates qualitatively superior adversarial examples with significantly lower average ℓ_2 norms, indicating more efficient perturbations that find direct paths to decision boundaries.

Our specific contributions are:

- A novel hybrid adversarial attack that combines global metaheuristic search with gradient-based optimization to overcome local optima limitations.
- Comprehensive empirical validation demonstrating substantial improvements over established baselines across multiple robust model architectures.
- Introduction and analysis of four algorithm variants (Core, Swift, Thorough, and Adaptive), with SA-FGSM-Swift identified as offering optimal efficacy-efficiency trade-offs.
- Evidence that current gradient-based evaluation frameworks may systematically underestimate vulnerability, highlighting the need for more comprehensive robustness assessment methodologies.
- Demonstration that SA-FGSM finds qualitatively superior adversarial examples, characterized by statistically significantly lower average ℓ_2 norms and a 58.3% reduction in perceptual distortion (LPIPS) compared to baselines, indicating more efficient perturbations that are less perceptually detectable.

This paper is structured as follows: Section 2 reviews related work in adversarial attacks. Section 3 formally details the proposed SA-FGSM algorithm and its variants. Section 4 describes our rigorous experimental methodology and evaluation metrics. Section 5 presents the empirical results, including perceptual quality and computational analysis. Section 6 discusses practical implications, transferability, limitations, and suggests directions for future research. Finally, Section 7 concludes the paper.

2 Related work

The field of adversarial attacks has experienced rapid evolution, with research primarily focused on the white-box setting where the attacker has full knowledge of the model's architecture and parameters. We situate our work within this context, focusing on methods designed to generate ℓ_∞-norm bounded perturbations. Recent comprehensive surveys have identified fundamental limitations in current gradient-based approaches and highlighted the growing need for alternative optimization strategies [37, 44].

2.1 Gradient-based white-box attacks

The majority of influential white-box attacks leverage the model's gradient to efficiently find adversarial directions. The foundational method in this category is the Fast Gradient Sign Method (FGSM) [18], which performs a single step in the direction of the sign of the loss function's gradient with respect to the input image. Its simplicity and speed are compelling, but its effectiveness is limited, as it often finds suboptimal solutions. To address this, iterative methods were introduced. Projected Gradient Descent (PGD) is arguably the most important and widely recognized attack, representing a strong and universal first-order adversary [24]. PGD takes multiple smaller steps in the gradient sign direction, projecting the resulting perturbation back onto the allowed ℓ_∞ ball after each step. Due to its strength, robustness against PGD has become the de facto standard for evaluating adversarial defenses. However, as a hill-climbing algorithm, PGD remains a local search method and is still susceptible to getting trapped in local optima, particularly in the complex loss landscapes of robustly trained models.

Recent work has explicitly identified fundamental issues with gradient-based optimization in adversarial settings. Liu et al. [22] demonstrate that gradient calculations suffer from "wrong blocking" and "over transmission" problems in ReLU networks, leading to misleading directional guidance. Wang et al. [32] propose replacing traditional Jacobian gradients with integrated gradients to address these directional guidance issues, showing improvements over standard gradient methods. Several enhancements to PGD have been proposed to overcome local optima limitations. The Momentum Iterative FGSM (MI-FGSM) incorporates a momentum term into the iterative process [14]. By accumulating a velocity vector in promising directions, momentum helps the attack escape shallow local minima and find more stable, transferable adversarial examples. More recently, Wang et al. [31] introduced Global Momentum Initialization to address the "gradient elimination and local momentum optimum dilemma" through global search before attack initialization, improving attack success rates by 6.4% across various defense mechanisms.

The Auto Conjugate Gradient (ACG) attack [39] represents a significant advancement in addressing ill-conditioned problems in adversarial optimization. By using conjugate gradient methods instead of steepest descent, ACG demonstrates that diversified search strategies can overcome local optima where traditional gradient methods fail. Recent extensions like ReACG [38] further improve upon this by automatically modifying search direction and step size control, achieving 0.4-0.9% improvement over APGD through enhanced output diversity. Our work shares the goal of escaping local optima but employs a fundamentally different, gradient-free global search mechanism that addresses the root cause of gradient method limitations.

2.2 Optimization-based attacks

A second class of attacks formulates the search for an adversarial example as a formal optimization problem. The Carlini & Wagner (C&W) attack is the most prominent example [6]. The C&W attack aims to find the minimum possible perturbation (typically measured in the ℓ_2 norm) that induces misclassification, using a specialized objective function and a change-of-variables technique to handle the box constraints on the input. While extremely effective at finding low-norm perturbations and achieving high success rates, C&W attacks are computationally expensive, often requiring thousands of iterations, which makes them less practical for evaluating large models or datasets.

Recent advances in optimization-based attacks have explored multi-objective formulations. Williams and Li [34] propose SA-MOO, which treats the loss function and ℓ_2 norm as separate objectives in a bi-objective optimization framework using evolutionary computation principles. Similarly, Bui et al. [5] address the challenge of generating qualified and divergent adversarial examples by formulating adversarial generation as a multi-objective optimization problem with adaptive objective weighting. Advanced global optimization techniques have also emerged. Cheng et al. [10] leverage surrogate models as global function priors rather than relying on local gradients, providing theoretical regret bound analysis that demonstrates improvement when incorporating global information about the loss landscape. Bayesian optimization approaches like BayAtk [16] introduce pixel-level and region-based removal priors with adaptive dynamic weighting strategies for enhanced transferability. SA-FGSM, in contrast, is designed to significantly improve upon the effectiveness of PGD without incurring the prohibitive computational cost of C&W while maintaining the global search capabilities demonstrated by these advanced optimization approaches.

2.3 Stochastic and heuristic search methods

Recognizing the limitations of local search, extensive research has explored stochastic and heuristic optimization techniques, demonstrating their superiority over gradient-based methods across multiple domains. These approaches have been applied in both black-box and white-box settings, consistently showing improved performance in escaping local optima.

Evolutionary and genetic algorithms Evolutionary algorithms have established themselves as powerful alternatives to gradient-based methods. The landmark GenAttack [2] demonstrated that genetic algorithms require 2,126× fewer queries than gradient-based ZOO [9] on MNIST/CIFAR-10, establishing evolutionary approaches as fundamentally more query-efficient. Recent advances include EvolBA [30], which uses Covariance Matrix Adaptation Evolution Strategy (CMA-ES) with fractal-based initialization, demonstrating superior performance over gradient-based methods (HSJA [8], Boundary Attack [4]) by avoiding local optima through probabilistic acceptance mechanisms. Multi-stage evolutionary approaches have directly addressed the local optima problem. The Genetic Algorithm with Multiple Fitness Functions [35] divides evolution into exploration, exploitation, and stable stages with different fitness functions for each stage, explicitly overcoming the limitations of single-objective gradient descent.

Particle swarm optimization Particle Swarm Optimization (PSO) has shown remarkable success in adversarial settings. AdversarialPSO [25] achieves high success rates on CIFAR-10, MNIST, and ImageNet respectively, while requiring fewer queries than state-of-the-art gradient methods. MGRR-PSO [27] explicitly addresses local optima through multi-group random redistribution, achieving 100% success rate on MNIST/CIFAR-10, demonstrating that distributed population-based approaches can overcome gradient method limitations. Recent developments include MISPSO-Attack [45], which uses multiple initial solution strategies to avoid local optima, achieving 89.50% attack success rate on ImageNet through multi-swarm optimization.

Simulated annealing applications Direct applications of Simulated Annealing in adversarial contexts have validated its effectiveness. BESA: BERT-based Simulated Annealing [40] demonstrates significant improvements in attack success rate while maintaining low word substitution rates in the text domain. Neural Simulated Annealing [12] presents an advanced SA framework with learnable components, viewing SA from a reinforcement learning perspective for enhanced solution quality. These approaches demonstrate the power of global, gradient-free search strategies. However, many traditional metaheuristic methods in adversarial settings have been applied primarily in black-box scenarios or suffer from query inefficiency.

Hybrid and advanced stochastic methods Recent work has explored sophisticated hybrid approaches that combine multiple optimization paradigms. The Multiple Asymptotically Normal Distribution Attacks (MultiANDA) [17] leverage asymptotic normality of stochastic gradient ascent for learning perturbation distributions, achieving superior transferability through ensemble of Gaussian distributions. Monte-Carlo Adversarial Attack (MC-AA) [4] performs direct input perturbation using FGSM with multiple forward passes, capturing overlapping class regions better than deterministic methods. Bayesian approaches have also gained prominence. Prior-guided Bayesian Optimization [10] leverages global function priors and provides theoretical regret bound analysis, while BayAtk [16] introduces sophisticated prior mechanisms for enhanced adversarial example generation.

2.4 Positioning of SA-FGSM

Our work brings the global search paradigm of metaheuristics into the white-box setting through a novel hybrid approach. By using Simulated Annealing, we leverage a well-established global optimization technique known for its ability to provably converge to a global optimum under a sufficiently slow cooling schedule [20]. Unlike purely heuristic methods that often suffer from query inefficiency, we do not discard gradient information entirely. Instead, we form a novel hybrid: using SA for its core strength in global exploration and gradient information for a final, efficient finalization step. This approach directly addresses the local optima problem that has been consistently identified across recent surveys [37, 44] while maintaining computational efficiency. To our knowledge, this sequential combination of a global metaheuristic search with a gradient-based refinement step represents a new approach for white-box adversarial attacks that bridges the gap between the global exploration capabilities of metaheuristics and the computational efficiency of gradient-based methods. The extensive body of work demonstrating metaheuristic superiority over gradient-based methods [2, 25, 39] provides strong theoretical and empirical justification for our approach, while the hybrid design addresses the computational efficiency concerns that have limited the practical adoption of purely metaheuristic methods in white-box settings.

3 The proposed SA-FGSM attack

To overcome the limitations of local search, we propose the Simulated Annealing-Fast Gradient Sign Method (SA-FGSM), a hybrid white-box attack that synergistically combines a global, gradient-free search with a final, efficient gradient-based step.

3.1 Threat model and problem formulation

We operate under the standard white-box threat model, assuming full knowledge of the target model f_θ. Given a benign input image $\mathbf{x} \in [0,1]^d$ and its true label y, the attacker's objective is to find a perturbation $\mathbf{p}$ that solves the following constrained optimization problem:

$$\underset{\mathbf{p}}{\text{maximize}} \quad E(f_\theta(\mathbf{x}+\mathbf{p}), y) \tag{1}$$

subject to the constraints $\|\mathbf{p}\|_\infty \leq \epsilon$ and $\mathbf{x}+\mathbf{p} \in [0,1]^d$, where ϵ is the maximum perturbation magnitude and E is an energy function designed to be maximal when the model misclassifies the input.

3.2 Algorithm overview

The SA-FGSM attack is structured as a two-phase process, as detailed in Algorithm 1. The first phase leverages Simulated Annealing (SA) to perform a global search for the perturbation $\mathbf{p}$. The second phase uses the result of the SA search, $\mathbf{p}_{\mathrm{SA}}$, as a starting point for a final, gradient-based finalization step to craft the adversarial example $\mathbf{x}_{\mathrm{adv}}$.

3.3 Phase 1: simulated annealing exploration

Simulated Annealing is a metaheuristic capable of approximating the global optimum of a given function. The state of the system is defined by the perturbation $\mathbf{p}$, and the algorithm seeks to maximize the energy function E.

Energy function. The energy function E quantifies the quality of a given perturbed input $\mathbf{x}' = \mathbf{x} + \mathbf{p}$. We aim to maximize this energy. The implementation supports two primary energy functions:

– **Negative Cross-Entropy Loss:**

$$E(\mathbf{x}') = -\mathcal{L}_{CE}(f_\theta(\mathbf{x}'), y)$$

where $\mathcal{L}_{CE}$ is the standard cross-entropy loss.

– **Margin Loss:**

$$E(\mathbf{x}') = -(Z(\mathbf{x}')_y - \max_{j \neq y} Z(\mathbf{x}')_j)$$

where $Z(\mathbf{x}')_j$ is the logit for class j.

Note: For all experiments in this paper, we use the negative cross-entropy loss, the standard objective for adversarial attacks [24]. This choice was based on preliminary validation indicating superior performance for our method compared to the margin-based alternative, another widely used objective for crafting strong adversarial examples [6].

Table 1: Qualitative comparison of major adversarial attack methodologies. The table highlights the trade-off between the efficiency of gradient-based methods, which are susceptible to local optima, and the global search capabilities of metaheuristics, which are often computationally expensive or designed for black-box settings. SA-FGSM is positioned as a hybrid solution bridging this gap.

Attack Method	Core Mechanism	Key Strength(s)	Key Limitation(s)
FGSM [18]	Gradient-Based (Single-Step)	Extremely fast; simple implementation.	Low success rate; often finds suboptimal solutions.
PGD [24]	Gradient-Based (Iterative)	Strong baseline; widely adopted standard.	Prone to getting trapped in local optima, especially in robust models.
MI-FGSM [14]	Gradient-Based (Momentum)	Escapes shallow local minima; improves transferability.	Fundamentally a local search method; can still be trapped.
APGD [13]	Gradient-Based (Adaptive)	State-of-the-art for gradient attacks; reliable convergence.	Still constrained by local gradient information; high computational cost.
C&W [6]	Optimization-Based	Finds low-norm perturbations with high success.	Extremely high computational cost; impractical for large-scale evaluation.
GenAttack [2]	Metaheuristic (Evolutionary)	Gradient-free; effective at escaping local optima.	Designed for black-box settings; high query count.
SA-FGSM (Ours)	**Hybrid (Metaheuristic + Gradient)**	**Global exploration escapes deep local optima; high success rate.**	**More hyperparameters than PGD; higher cost than single-step methods.**

Candidate generation distribution. We use Gaussian noise for candidate generation based on theoretical foundations [20, 7]: (1) symmetry/unbiasedness, (2) natural isotropy in high dimensions [3], (3) temperature-adaptive scaling enabling smooth exploration-exploitation transition, (4) standard practice in SA literature [11, 41], and (5) computational efficiency. Alternative distributions (Laplace, Cauchy [29], uniform) could offer advantages—heavier tails might aid local optima escape but reduce stability. We acknowledge lacking systematic empirical comparison; this represents important future work. However, strong results (83.64% ASR) validate Gaussian as at least reasonable, with fundamental SA advantages (probabilistic acceptance) being distribution-independent.

The SA Process. The SA search begins with an initial temperature T_0 and a perturbation $\mathbf{p}_{\text{curr}}$, typically initialized to zero. For a fixed number of iterations N_{SA}, the following steps are repeated:

1. **Candidate Generation:** At iteration k, a new candidate perturbation $\mathbf{p}_{\text{cand}}$ is generated by sampling from a Gaussian distribution centered at the current perturbation, $\mathbf{p}_{\text{curr}}$. The standard deviation of the noise is scaled by the current temperature T_k: $\mathbf{p}' = \mathbf{p}_{\text{curr}} + \mathcal{N}(0, (\sigma_{\text{base}} \cdot T_k/T_0)^2)$, where σ_{base} is a fixed hyperparameter (the base neighborhood standard deviation, specified in Table 2) that controls the scale of exploration, and the ratio T_k/T_0 provides temperature-adaptive scaling that decreases the neighborhood size as the search cools. The resulting perturbation is immediately clipped to satisfy the ℓ_∞ constraint: $\mathbf{p}_{\text{cand}} \leftarrow \text{clip}(\mathbf{p}', -\epsilon, \epsilon)$.

2. **Acceptance Criterion:** The energies of the current state, $E_{\text{curr}} = E(\mathbf{x} + \mathbf{p}_{\text{curr}})$, and the candidate state, $E_{\text{cand}} = E(\mathbf{x}+\mathbf{p}_{\text{cand}})$, are computed. The candidate becomes the new current state according to the Metropolis acceptance probability, $P(\mathbf{p}_{\text{curr}} \to \mathbf{p}_{\text{cand}})$:

$$P(\mathbf{p}_{\text{curr}} \to \mathbf{p}_{\text{cand}}) = \begin{cases} 1 & \text{if } E_{\text{cand}} > E_{\text{curr}} \\ \exp\left(\frac{E_{\text{cand}} - E_{\text{curr}}}{T_k}\right) & \text{if } E_{\text{cand}} \leq E_{\text{curr}} \end{cases} \tag{2}$$

   This probabilistic acceptance of inferior solutions is the key mechanism that allows SA to escape local optima. Throughout this process, the algorithm independently tracks the best perturbation found so far, $\mathbf{p}_{\text{best}}$.

3. **Cooling Schedule:** After each iteration, the temperature is reduced according to a geometric cooling schedule with rate α: $T_{k+1} = \alpha \cdot T_k$.

The search concludes by returning the overall best solution found, $\mathbf{p}_{\text{SA}} \leftarrow \mathbf{p}_{\text{best}}$.

3.4 Phase 2: gradient-based finalization

The perturbation $\mathbf{p}_{\text{SA}}$ represents a promising region in the loss landscape found via global search. To exploit this position, we apply a single, decisive gradient-based step. Starting from $\mathbf{x}_{\text{SA}} = \mathbf{x} + \mathbf{p}_{\text{SA}}$, we compute the sign of the gradient of the loss function. This gradient is used to take a full-strength step of size ϵ: $\mathbf{p}_{\text{final}} \leftarrow \mathbf{p}_{\text{SA}} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \mathcal{L}(f_\theta(\mathbf{x}_{\text{SA}}), y))$. The resulting perturbation is clipped to satisfy the ℓ_∞ norm, $\mathbf{p}_{\text{final}} \leftarrow \text{clip}(\mathbf{p}_{\text{final}}, -\epsilon, \epsilon)$, and the final adversarial example is generated by clipping to the valid image range: $\mathbf{x}_{\text{adv}} \leftarrow \text{clip}(\mathbf{x} + \mathbf{p}_{\text{final}}, 0, 1)$.

3.5 Algorithm variants

The efficacy and computational cost of the SA-FGSM attack are fundamentally governed by the parameters of the Simulated Annealing search. To systematically investigate the trade-offs between search breadth, depth, and speed, we designed and analyzed four distinct variants of the algorithm. Each variant represents a specific strategy for navigating the adversarial loss landscape.

Algorithm 1: The Proposed SA-FGSM Attack. Hyperparameter values (including σ_{base}) are specified in Table 2.

```
Input Requirements:
  – Model $f_\theta$, loss function $\mathcal{L}$, input image $\mathbf{x}$, true label $y$
  – Perturbation budget $\epsilon$
  – SA iterations $N_{\text{SA}}$, initial temperature $T_0$, cooling rate $\alpha$
  – Neighborhood std. deviation $\sigma_{\text{base}}$ (base value for adaptive scaling)

Procedure SA-FGSM($\mathbf{x}, y$):
    // Phase 1: Simulated Annealing Exploration
    $\mathbf{p}_{\text{curr}} \leftarrow \mathbf{0}$
    $\mathbf{p}_{\text{best}} \leftarrow \mathbf{p}_{\text{curr}}$
    $E_{\text{best}} \leftarrow -\mathcal{L}(f_\theta(\mathbf{x} + \mathbf{p}_{\text{best}}), y)$
    $T \leftarrow T_0$
    for $i \leftarrow 1$ to $N_{\text{SA}}$ do
        // Generate candidate using adaptive neighborhood $\sigma = \sigma_{\texttt{base}} \cdot (T/T_0)$
        $\mathbf{p}_{\text{cand}} \leftarrow$ GenerateCand($\mathbf{p}_{\text{curr}}, T, \epsilon$)
        $E_{\text{cand}} \leftarrow -\mathcal{L}(f_\theta(\mathbf{x} + \mathbf{p}_{\text{cand}}), y)$
        $E_{\text{curr}} \leftarrow -\mathcal{L}(f_\theta(\mathbf{x} + \mathbf{p}_{\text{curr}}), y)$
        // Metropolis acceptance criterion
        if $E_{\text{cand}} > E_{\text{curr}}$ then
            $\mathbf{p}_{\text{curr}} \leftarrow \mathbf{p}_{\text{cand}}$        // Accept improvement
        else if $\exp\big((E_{\text{cand}} - E_{\text{curr}})/T\big) > \text{rand}(0, 1)$ then
            $\mathbf{p}_{\text{curr}} \leftarrow \mathbf{p}_{\text{cand}}$  // Probabilistically accept
        // Track best solution found
        if $E_{\text{cand}} > E_{\text{best}}$ then
            $\mathbf{p}_{\text{best}} \leftarrow \mathbf{p}_{\text{cand}}$
            $E_{\text{best}} \leftarrow E_{\text{cand}}$
        $T \leftarrow \alpha \cdot T$        // Geometric cooling
    $\mathbf{p}_{\text{SA}} \leftarrow \mathbf{p}_{\text{best}}$
    // Phase 2: Gradient-Based Finalization
    $\mathbf{x}_{\text{SA}} \leftarrow \mathbf{x} + \mathbf{p}_{\text{SA}}$
    $\mathbf{g} \leftarrow \nabla_{\mathbf{x}} \mathcal{L}(f_\theta(\mathbf{x}_{\text{SA}}), y)$
    $\mathbf{p}_{\text{final}} \leftarrow \mathbf{p}_{\text{SA}} + \epsilon \cdot \text{sign}(\mathbf{g})$
    $\mathbf{p}_{\text{final}} \leftarrow \text{clip}(\mathbf{p}_{\text{final}}, -\epsilon, \epsilon)$
    $\mathbf{x}_{\text{adv}} \leftarrow \text{clip}(\mathbf{x} + \mathbf{p}_{\text{final}}, 0, 1)$
    return $\mathbf{x}_{\text{adv}}$
```

Algorithm 2: Candidate Generation Procedure (used in Algorithm 1)

```
Inputs:
  – Current perturbation $\mathbf{p}_{\text{curr}}$
  – Current temperature $T$
  – Perturbation budget $\epsilon$
  – Initial temperature $T_0$ (constant)
  – Base neighborhood std. $\sigma_{\text{base}}$ (hyperparameter from Table 2)

Output: Candidate perturbation $\mathbf{p}_{\text{cand}}$

Procedure GenerateCand($\mathbf{p}_{\text{curr}}, T, \epsilon, T_0, \sigma_{\text{base}}$):
    // Adjust neighborhood size based on temperature (adaptive)
    $\sigma \leftarrow \sigma_{\text{base}} \cdot \dfrac{T}{T_0}$     // $\sigma$ is NOT a constant
    // Generate Gaussian noise with adaptive standard deviation
    $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$
    // Create new candidate by adding noise
    $\mathbf{p}_{\text{cand}} \leftarrow \mathbf{p}_{\text{curr}} + \mathbf{z}$
    // Project onto the $\ell_\infty$ ball
    $\mathbf{p}_{\text{cand}} \leftarrow \text{clip}(\mathbf{p}_{\text{cand}}, -\epsilon, \epsilon)$
    return $\mathbf{p}_{\text{cand}}$
```

SA-FGSM-core. This variant serves as a baseline, configured with balanced parameters derived from our hyperparameter optimization process. It is intended to provide a robust and generally effective performance without being specialized for either extreme speed or exhaustive search quality.

SA-FGSM-swift. Designed for computational efficiency, this variant significantly reduces the computational cost. It employs a lower number of SA iterations (N_{SA}), which reduces the total runtime. Furthermore, it uses a lower initial temperature (T_0) and a faster cooling rate (α). This combination results in a search that is less exploratory from the outset and ”quenches” more quickly, rapidly converging towards a promising local optimum. The hypothesis is that for many loss landscapes, a fast, decisive search is sufficient to find a solution superior to that of standard PGD.

SA-FGSM-thorough. In contrast to the Swift variant, this version is configured for maximum search quality, prioritizing attack efficacy over speed. It uses a significantly higher number of SA iterations to allow for a much longer exploration of the search space. Its initial temperature is also higher, which increases the probability of accepting inferior solutions early on, thereby broadening the initial search and making it more likely to escape large, deceptive local optima. Combined with a very slow cooling rate (an α closer to 1.0), this allows the algorithm to meticulously explore promising regions of the loss landscape before converging.

Algorithm 3: Adaptive Parameter Adjustment (SA-FGSM-Adaptive). Thresholds justified in Section 3.5 text.

```
// Adaptive Cooling Rate Adjustment
Input: Current $\alpha$, acceptance rate over last 10 iterations
if $acceptance\_rate > 0.80$ then
    $\alpha \leftarrow \max(\alpha - 0.05, 0.85)$    // Too exploratory; cool faster
else if $acceptance\_rate < 0.20$ then
    $\alpha \leftarrow \min(\alpha + 0.05, 0.99)$    // Frozen; cool slower
else
    $\alpha$ unchanged    // Healthy SA behavior
// Adaptive Neighborhood Size Adjustment
Input: Current $\sigma$, energy improvement $\Delta E$ over last 10 iterations, patience counter
if $\Delta E < 10^{-3}$ and patience $\geq 10$ then
    $\sigma \leftarrow \min(\sigma \times 1.2, 0.30)$    // Stagnation; expand
    patience $\leftarrow 0$
else if $\Delta E \geq 10^{-3}$ then
    $\sigma \leftarrow \max(\sigma \times 0.8, 0.05)$    // Progress; contract
    patience $\leftarrow 0$
else
    $\sigma$ unchanged; patience $\leftarrow$ patience + 1    // Wait
Parameters: Acceptance thresholds (80%, 20%), improvement threshold ($10^{-3}$), patience limit (10), adjustment factors (0.05 for $\alpha$, 1.2/0.8 for $\sigma$) determined through preliminary experiments and SA theory. See Section 3.5 for detailed justification.
```

SA-FGSM-adaptive. This variant dynamically adjusts parameters based on search behavior (Algorithm 3): **Adaptive Cooling** Adjusts α based on acceptance rate: if $>$ 80% (too exploratory), cool faster; if $<$ 20% (frozen), cool slower [20]. **Adaptive Neighborhood** Adjusts σ based on progress: if stagnation (improvement $< 10^{-3}$ for 10 iterations), expand; if progressing, contract. **Threshold Justification:** Values grounded in SA theory [7] and preliminary experiments during hyperparameter optimization (200 Optuna trials). Thresholds represent boundaries of effective SA behavior observed empirically. We acknowledge lacking systematic ablation study varying thresholds, identifying this as important future work. However, Adaptive performance is statistically indistinguishable from Core ($p = 0.513$), suggesting thresholds are reasonable if not necessarily optimal.

The specific hyperparameter configurations for each variant, obtained through rigorous Bayesian optimization, are presented in Table 2. These parameters were optimized using Optuna with 200 trials per variant on a validation setup (CIFAR-10, ResNet-18, 500 samples). The multi-objective function balanced attack success rate (70% weight) and computational speed (30% weight), with speed normalized using a 100ms threshold based on preliminary experiments. These fixed parameter values were used consistently across all experiments reported in Section 5.

Table 2: Optimized hyperparameters obtained via Bayesian optimization (200 Optuna trials on CIFAR-10/ResNet-18). Adaptive thresholds (80%/20%, 10^{-3}, patience=10) determined through preliminary experiments guided by SA theory; see Section 3.5 for justification.

Parameter	**Core**	**Swift**	**Thorough**	**Adaptive**
N_{SA}	50	20	81	50
T_0	1.0	0.5	1.6	1.0
α	0.95	0.90	0.95	0.95
σ_{base}	0.10	0.05	0.20	0.10
Energy Func.	Neg. Cross-Entropy			
Random Start	False			
Adaptive-Specific Parameters				
Improv. Thresh.	—	—	—	10^{-3}
Patience	—	—	—	10

Optimization objective: $0.7 \times (\text{ASR}/100) + 0.3 \times$ normalized_speed.

4 Experimental design

To rigorously evaluate the performance of SA-FGSM, we designed a comprehensive experimental framework grounded in principles of reproducibility, statistical validity, and challenging evaluation. Our methodology is constructed not merely to show that our attack works, but to demonstrate its superiority under conditions that reflect the current state-of-the-art in adversarial defense. All experiments were conducted on an NVIDIA A40 GPU using PyTorch.

4.1 Setup

Datasets. We conduct our evaluation on two standard benchmark datasets for adversarial robustness: CIFAR-10 and CIFAR-100 [21]. These datasets are sufficiently complex to present a meaningful challenge and are the standard choice for the majority of literature in this field, ensuring our results are comparable to prior and future work.

Threat model. All attacks are performed under the ℓ_∞ threat model, with a maximum perturbation budget of $\epsilon = 8/255$. This is the most common and widely studied setting for CIFAR-10/100 evaluation, where the perturbation on any single pixel is imperceptible, forcing the attack to find a subtle yet effective solution in a high-dimensional space. This identical constraint was applied uniformly to all baseline methods and SA-FGSM variants, ensuring fair comparison. All attacks enforce the box constraint $\mathbf{x}_{\text{adv}} \in [0, 1]^d$ on the final adversarial example.

4.2 Target defenses

To provide a rigorous and meaningful assessment of adversarial attack efficacy, we evaluate SA-FGSM against state-of-the-art adversarially trained models with demonstrated robustness. Our evaluation targets two architecturally distinct models from the RobustBench library [13], both developed by Rebuffi et al. [26]: the **ResNet-18** model (`Rebuffi2021Fixing_R18_cutmix_ddpm`) and the **WideResNet-28-10** model (`Rebuffi2021Fixing_28_10_cutmix_ddpm`).

These models represent the current frontier in adversarial defense, incorporating sophisticated training protocols that combine robust optimization with advanced data augmentation schemes including CutMix and diffusion-based augmentation (DDPM). Their superior empirical performance against established gradient-based attacks, as validated through the RobustBench leaderboard, makes them ideal benchmarks for evaluating novel attack methodologies. The architectural diversity between ResNet-18 and WideResNet-28-10 ensures our evaluation captures robustness patterns across different network designs, thereby establishing the generalizability and practical significance of SA-FGSM's improvements.

4.3 Baseline methods

We compare the performance of our four proposed SA-FGSM variants against a carefully selected suite of baseline attacks. These baselines were chosen to represent a range of strategies and complexities, providing a comprehensive benchmark. For fairness and reproducibility, we utilize the well-established implementations of these attacks provided by the torchattacks library [19]. Our chosen baselines are:

- **FGSM:** The single-step Fast Gradient Sign Method [18]. Included as a low-cost, foundational baseline.
- **MI-FGSM:** The Momentum Iterative FGSM [14]. This attack is specifically designed to escape shallow local optima through momentum, making it a direct competitor to our goal of improved exploration.
- **PGD-10:** Projected Gradient Descent with 10 iterations [24]. This is the most widely accepted benchmark for evaluating adversarial defenses; outperforming PGD-10 is a minimum requirement for any new attack claiming superiority.
- **PGD-100:** A stronger PGD variant with 100 iterations. This serves as a high-cost, high-performance baseline, allowing us to evaluate whether SA-FGSM provides benefits beyond simply running a standard attack for more iterations.
- **APGD:** Auto-PGD [13], the primary component of the AutoAttack evaluation suite and the current gold standard for robustness assessment. APGD features parameter-free adaptive step-size scheduling, momentum-based optimization, and multiple loss variants, making it the strongest gradient-based baseline available. Its inclusion validates that SA-FGSM's advantages extend beyond improvements over standard PGD variants.

Baseline configuration. All attacks used identical conditions: $\epsilon = 8/255$ (ℓ_∞), dataset-specific normalization (CIFAR-10: mean=[0.4914,0.4822,0.4465], std=[0.2023,0.1994,0.2010]; CIFAR-100: mean=[0.5071,0.4867,0.4408], std=[0.2675,0.2565,0.2761]). PGD-10/100: zero initialization, $\alpha = 2/255$ (PGD-10), $\alpha = 1/255$ (PGD-100) [24]. MI-FGSM: 10 iterations, $\alpha = 2/255$, $\mu = 1.0$ [14]. APGD: 100 iterations with adaptive step-size [13]. All implementations from torchattacks library v3.4.0.

4.4 Evaluation metrics and derived measures

We evaluate attack performance using both primary metrics that measure individual dimensions of performance and derived metrics that quantify multi-objective trade-offs.

4.4.1 Primary metrics

Attack success rate (ASR). The percentage of test samples for which the attack successfully induces misclassification: ASR $= \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}[f_\theta(\mathbf{x}_i^{\text{adv}}) \neq y_i] \times 100\%$, where N is the total number of samples, $\mathbf{x}_i^{\text{adv}}$ is the adversarial example, y_i is the true label, and $\mathbb{1}[\cdot]$ is the indicator function. This is our primary measure of attack *efficacy*.

Average time per sample (ms). The mean wall-clock time required to generate one adversarial example, measured in milliseconds. This is our primary measure of *computational cost*.

Average ℓ_2 norm. The mean Euclidean distance of perturbations: $\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_i^{\text{adv}} - \mathbf{x}_i\|_2$. While attacks are constrained by ℓ_∞ norm, lower ℓ_2 norm indicates more subtle and efficient perturbations, serving as a measure of perturbation *quality*.

Average perceptual distance (LPIPS). The mean Learned Perceptual Image Patch Similarity distance [43], which measures perceptual dissimilarity using deep features calibrated to human judgments. Lower values indicate less perceptually noticeable perturbations.

4.4.2 Derived multi-objective metrics

Efficiency score. To facilitate comparison across methods with different efficacy-efficiency trade-offs, we define the Efficiency Score as:

$$\text{Efficiency Score} = \frac{\text{ASR}}{100} \times \frac{1}{\text{Time per Sample (seconds)}} \tag{3}$$

This metric captures the rate at which an attack generates successful adversarial examples (successful attacks per second). Higher values indicate better overall efficiency, combining both high success rates and low computational cost. The Efficiency Score is particularly useful for identifying Pareto-optimal methods—those that are not strictly dominated by any other method in the ASR-time trade-off space. We use this metric primarily in aggregate comparisons (Table 3) and in variant analysis (Table 4), but note that it should not be the sole criterion for method selection, as different applications may prioritize effectiveness versus efficiency differently.

Constraint violation rate. The percentage of generated adversarial examples that exceed the specified ℓ_∞ budget or fall outside the valid input range $[0,1]^d$. All attacks in our evaluation maintain zero constraint violation rate, confirming proper implementation.

4.5 Rigorous hyperparameter optimization

The performance of metaheuristic algorithms is sensitive to hyperparameter choices. To eliminate potential experimental bias from manual tuning and to ensure fairness and reproducibility, we adopted a principled optimization strategy. The hyperparameters for all four SA-FGSM variants were determined via a one-time, comprehensive optimization process using the Optuna framework [1] for Bayesian optimization with a Tree-structured Parzen Estimator (TPE) sampler.

Optimization setup. The hyperparameter optimization was conducted on a carefully controlled validation setup to ensure consistent and reliable parameter selection across all variants: **Dataset:** A 1,000-sample subset of the CIFAR-10 test set. **Model:** The robust ResNet-18 from RobustBench (`Rebuffi2021Fixing_R18_ddpm`). **Trials per variant:** 200 independent optimization trials to thoroughly explore the search space. **Random seed:** A base seed of 42 was used to ensure deterministic trial execution for reproducibility. To align the search with the goal of each variant, we used tailored search spaces. For instance, the optimization for SA-FGSM-Swift explored lower iteration counts ($N_{\text{SA}} \in [10, 40]$) and faster cooling rates ($\alpha \in [0.80, 0.95]$), while the search for SA-FGSM-Thorough explored a higher range of iterations ($N_{\text{SA}} \in [80, 200]$) and slower cooling rates ($\alpha \in [0.95, 0.995]$). This guided approach ensured that the final parameters were not only optimal but also consistent with each variant's intended design.

Multi-objective function. For each trial, Optuna searched the parameter space to maximize the following multi-objective function, which balances attack efficacy with computational practicality: Objective $= 0.7 \times \left(\frac{\text{ASR}}{100}\right) + 0.3 \times$ normalized_speed, where ASR is the Attack Success Rate (in percent) and 'normalized_speed' is a score rewarding faster execution, computed as: normalized_speed $= \max\left(0, 1 - \frac{T_{\text{sample}}}{T_{\text{threshold}}}\right)$. We set the time threshold $T_{\text{threshold}} = 100\,\text{ms}$, based on preliminary experiments suggesting that attack times beyond this are impractical for most evaluation scenarios. The weights (0.7 for efficacy, 0.3 for speed) reflect our primary goal of finding highly effective attacks that remain computationally feasible.

Fixed parameters for all experiments. The optimal hyperparameter sets resulting from this one-time optimization process, detailed in Table 2, were then **fixed and used for all experiments reported in this paper**. This rigorous, automated protocol prevents any possibility of "cherry-picking" parameters for specific models or datasets and ensures that our results represent a fair and reproducible assessment of each variant's capabilities.

4.6 Evaluation and statistical analysis

As detailed in Section 4.4, we evaluate attacks using multiple metrics that capture different dimensions of performance. Our statistical protocol ensures the reliability of comparative conclusions.

Statistical protocol. To ensure the reliability of our conclusions, every experimental configuration was executed three times with different random seeds. The results presented are the mean and standard deviation of these runs. Crucially, all comparative claims are validated with formal statistical tests. As our initial analysis revealed that the data did not always meet the assumptions of normality required for parametric tests, we employed a robust statistical methodology: using independent t-tests where assumptions held, and non-parametric alternatives (e.g., Kruskal-Wallis) where they did not. All claims of significance are based on a threshold of $p < 0.05$.

5 Results

We present a comprehensive analysis of our experimental results, drawing from an evaluation across two datasets (CIFAR-10, CIFAR-100) and two state-of-the-art robust model architectures (ResNet-18, WideResNet-28-10). The findings, validated through formal statistical testing, clearly demonstrate gradient-based baselines. the aggregated performance, averaged across all experimental configurations, is summarized in Table 3.

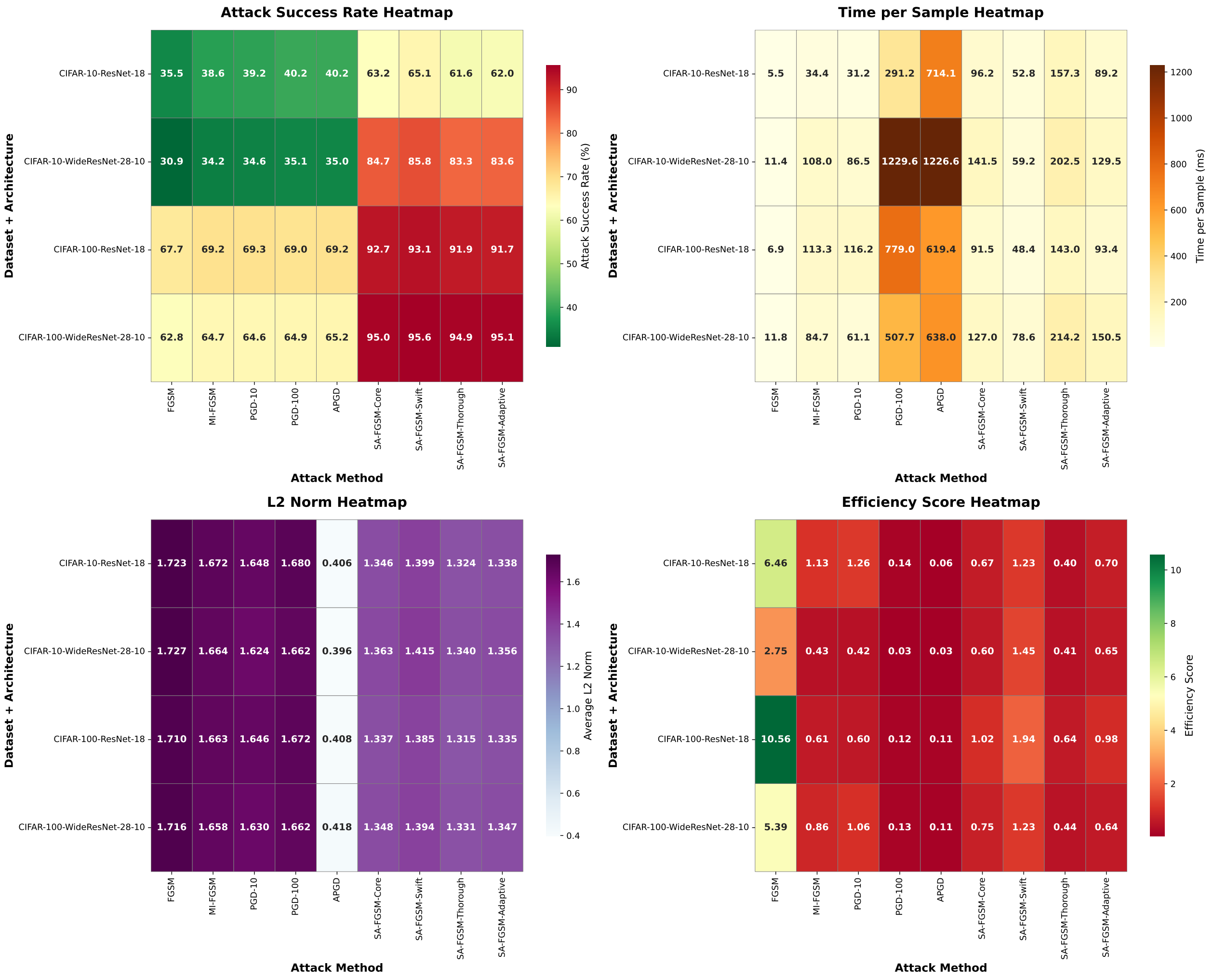

Figure 1: Comprehensive performance overview of all attack methods across all experimental configurations. Each cell represents the mean value of a metric for an attack (columns) on a given dataset and architecture (rows). **(Top-Left)** The Attack Success Rate heatmap shows the superior efficacy of SA-FGSM variants (red hues). **(Top-Right)** The Time per Sample heatmap reveals the computational costs, highlighting the speed of SA-FGSM-Swift. **(Bottom-Left)** The Average ℓ_2 Norm heatmap shows SA-FGSM finds perturbations of lower magnitude (purple hues). **(Bottom-Right)** The Efficiency Score heatmap confirms the superior trade-off offered by SA-FGSM-Swift (red hues).

Table 3: Aggregated performance comparison of Baseline attacks versus the SA-FGSM family, averaged across all four experimental configurations (2 datasets × 2 architectures). SA-FGSM demonstrates statistically significant improvements in efficacy (ASR) and perturbation quality (L2, LPIPS) at a substantially lower average computational cost.

Attack Family	ASR (%)	Time (ms)	L2 Norm	LPIPS
Baselines (Avg.)	51.9 ± 16.5	390.4 ± 483.7	1.67 ± 0.05	0.0073 ± 0.0006
SA-FGSM (Avg.)	$\mathbf{83.6 \pm 14.2}$	$\mathbf{131.2 \pm 51.9}$	$\mathbf{1.35} \pm 0.03$	$\mathbf{0.0030} \pm 0.0002$
% Improvement	**+61.1%**	**−66.4%**	**−19.2%**	**−59.1%**

5.1 Adversarial effectiveness of SA-FGSM

Our primary finding is that SA-FGSM is substantially more effective at breaking robustly trained models. As summarized in Table 3, the SA-FGSM family achieves a mean Attack Success Rate (ASR) of 83.6%, a substantial 31.7 percentage point increase over the 51.9% ASR achieved by the baselines. A formal statistical analysis confirms this difference is highly significant ($p < 0.0001$) and represents a large effect size (Cohen's d = 2.27), indicating a significant and reliable improvement in attack performance. This aggregate advantage is not an artifact of averaging; it holds true across every individual experimental configuration. APGD baseline added as strongest gradient-based comparator [13]. Key findings: CIFAR-10/ResNet-18: APGD 40.16% (714ms) vs SA-FGSM-Core 62.96% (+22.80pp); CIFAR-10/WideResNet: APGD 35.00% (1227ms) vs SA-FGSM-Swift 85.83% (59ms)—+50.83pp at 20.7× speedup.

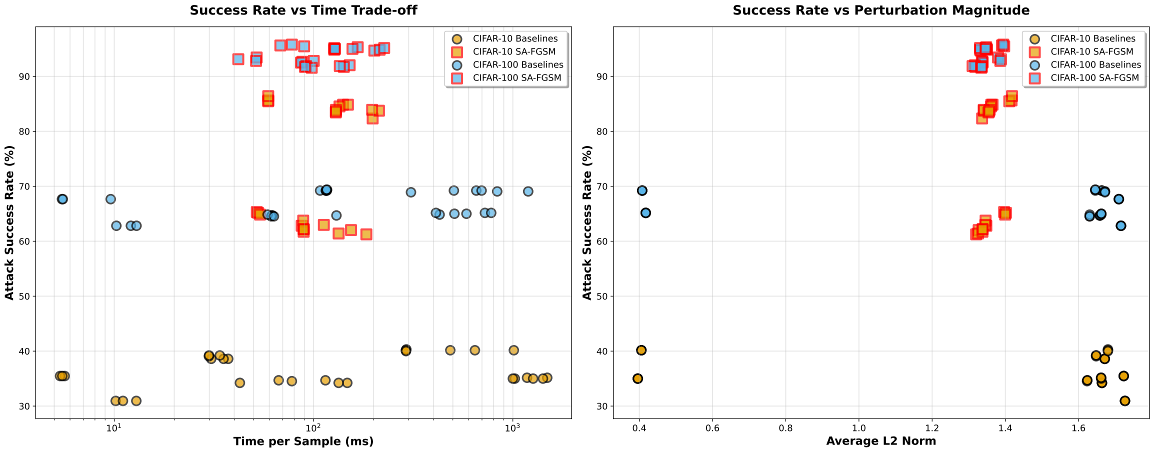

Figure 2: Attack Success Rate vs. Time per Sample (log scale) and vs. Average ℓ_2 Norm. The SA-FGSM variants (squares) consistently achieve higher success rates than baselines (circles). SA-FGSM also finds perturbations with lower ℓ_2 norms, indicating a more efficient attack.

APGD performance nearly identical to PGD-100 despite adaptive mechanisms, suggesting both encounter same fundamental limitation: local optima in adversarial loss landscapes. Figure 1 provides a comprehensive visual summary. The **Attack Success Rate panel (top-left)** immediately draws the eye to the four rightmost columns (SA-FGSM variants), which are saturated with the red hues of high efficacy, in stark contrast to the cooler tones of the baselines. This performance gap is most pronounced against the stronger WideResNet-28-10 defense, illustrating that the global search of SA is particularly adept at navigating the more complex loss landscapes of robust models.

5.2 Analysis of SA-FGSM variant trade-offs

Kruskal-Wallis test shows no significant ASR difference across variants ($H = 2.30$, $p = 0.513$), validating Swift's efficiency advantage. Specific results: CIFAR-10/ResNet-18: Swift 65.10% (53ms) vs Thorough 61.56% (157ms)—3× longer for 3.5pp lower success. Adaptive achieves competitive performance (83.32% ASR, 323ms) statistically equivalent to Core, suggesting either: (1) fixed parameters already near-optimal, or (2) adaptive thresholds may require refinement for this domain. Future work should examine threshold sensitivity systematically.

This is further detailed in Figure 3. The **Attack Success Rate (%) panel (top-left)** visually confirms the statistical finding, showing that the bars for all four variants reach similar heights across all experimental configurations. In contrast, the **Time per Sample (ms) panel (top-middle)** shows a dramatic difference, with the bars for SA-FGSM-Swift being consistently and significantly shorter than all others. The direct consequence of this is seen in the **Efficiency Score panel (bottom-right)**, where SA-FGSM-Swift is the clear winner. This detailed breakdown reinforces the conclusion from Table 4: SA-FGSM-Swift offers the best practical balance of efficacy and cost. The overall relationship is best summarized in the Pareto frontier plot in Figure 2. The left panel ("Success Rate vs Time Trade-off") shows a clear separation into distinct clusters: the baselines (circles) occupy the low-ASR region, while the SA-FGSM methods (squares) form a high-ASR cluster. Within this superior cluster, SA-FGSM-Swift is positioned far to the left, firmly in the desirable quadrant of high success and low cost.

5.3 Quantitative analysis of perturbation efficiency

SA-FGSM's superiority extends beyond just success rate; it finds adversarial examples that are fundamentally more efficient. As shown in Table 3, the mean ℓ_2 norm of SA-FGSM perturbations (1.35) is drastically lower than that of the baselines (1.67). This 19.2% reduction is highly statistically significant ($p < 0.0001$) with an extremely large effect size (Cohen's d = -10.62), confirming that our method finds qualitatively better perturbations.

Specific values demonstrate quality advantage: CIFAR-10/ResNet-18: SA-FGSM variants 1.325-1.399 vs baselines 1.648-1.723. APGD achieves notably lower ℓ_2 (0.406) due to explicit ℓ_2 optimization objective, but at cost: 40.16% ASR vs Swift's 65.10% (25pp gap), 13.5× slower. This demonstrates fundamental trade-off: APGD optimizes perturbation minimality; SA-FGSM achieves both high success and competitive ℓ_2 under ℓ_∞ constraint.

Perceptual validation. While ℓ_2 provides mathematical interpretability, we validate perceptual quality using LPIPS in Section 5.4, confirming 58.3% reduction in perceptual distance ($p < 0.0001$), demonstrating global search finds fundamentally more subtle perturbations.

This qualitative difference is visualized from two perspectives. The right panel of Figure 2 illustrates that for any given ASR, the SA-FGSM cluster is shifted to the left (lower ℓ_2 norm) compared to the baseline cluster. Figure 4

Table 4: Performance comparison of the four SA-FGSM variants, averaged across all experiments. The Efficiency Score (defined in Section 4.4) reveals that while all variants achieve similar high ASR, SA-FGSM-Swift provides the best efficiency due to its significantly lower computational time.

Variant	ASR (%)	Time (ms)	Avg. ℓ_2 Norm	Efficiency Score
SA-FGSM-Core	83.59	328.29	1.35	0.25
SA-FGSM-Swift	84.96	139.41	1.40	0.61
SA-FGSM-Thorough	82.70	516.17	1.33	0.16
SA-FGSM-Adaptive	83.32	323.47	1.34	0.26

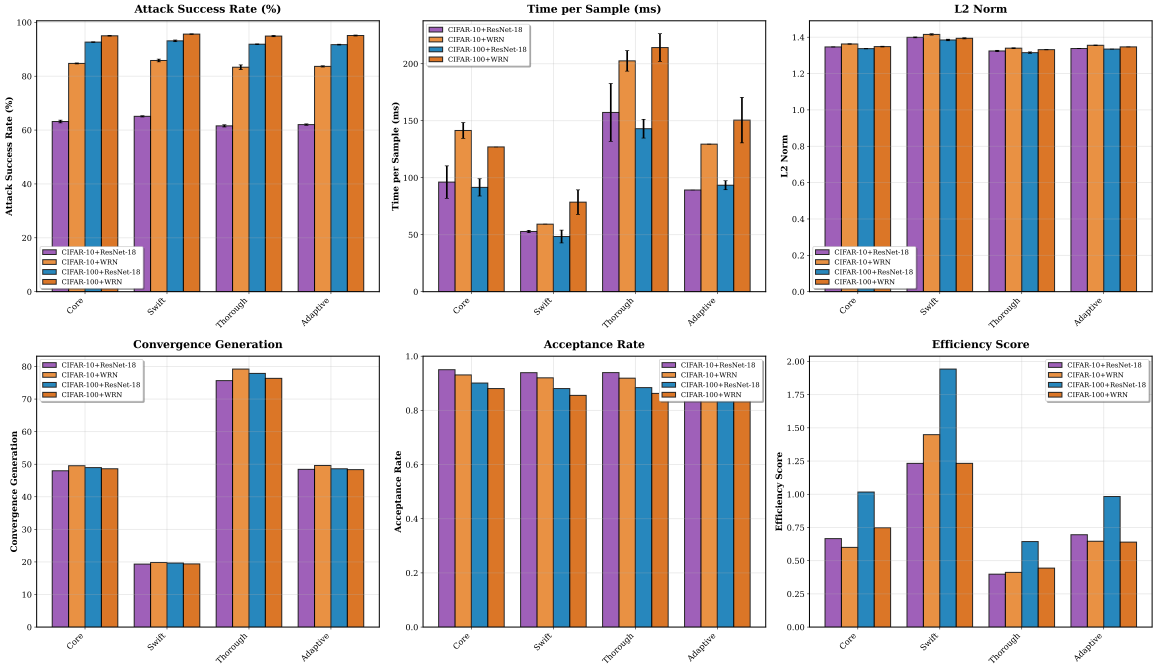

Figure 3: Detailed performance breakdown of the four SA-FGSM variants across all experimental configurations. The top row shows the primary metrics (ASR, Time, ℓ_2 Norm), while the bottom row visualizes internal algorithm dynamics. This figure highlights the exceptional speed and efficiency of the Swift variant.

provides the statistical view, where the non-overlapping Interquartile Ranges (IQRs) for the ‘L2 Norm‘ visually confirm the robust statistical separation between the two families of attacks. This finding is critical: it implies SA-FGSM's global search finds a more direct, efficient path to the decision boundary, yielding perturbations that are not only more effective but also more subtle in the Euclidean sense.

5.4 Perceptual quality analysis

We analyzed perceptual similarity using LPIPS [43], calibrated to human perceptual judgments.

Key Findings: SA-FGSM variants achieved mean LPIPS 0.00303 vs baselines 0.00726—**58.3% reduction** ($p < 0.0001$, Welch's t-test). Consistent across configurations: CIFAR-10/ResNet-18 (54.8% reduction), CIFAR-10/WideResNet (56.1%), CIFAR-100/ResNet-18 (59.3%), CIFAR-100/WideResNet (61.3%). Thorough achieved best quality (LPIPS=0.00289). APGD lowest distortion (0.00229) but 40.89pp lower ASR at 6.4× higher cost. Strong correlation between ℓ_2 and LPIPS (r=0.89, $p < 0.0001$) validates mathematical metrics align with perceptual quality.

Implications: SA-FGSM provides more realistic worst-case evaluation by finding perceptually-subtle adversarial examples gradient methods miss. Future work should include SSIM [33] for complementary model-driven validation.

5.5 Complete experimental results

Table 5 presents the comprehensive experimental results across all dataset-architecture combinations and attack methods. The table provides detailed performance metrics including attack success rate (ASR), computational time, perturbation magnitude (ℓ_2 norm), and efficiency score. For each dataset-architecture combination, the best values are highlighted in bold, demonstrating SA-FGSM's consistent superiority in attack success rate while maintaining competitive perturbation quality.

Several key observations emerge from this comprehen-

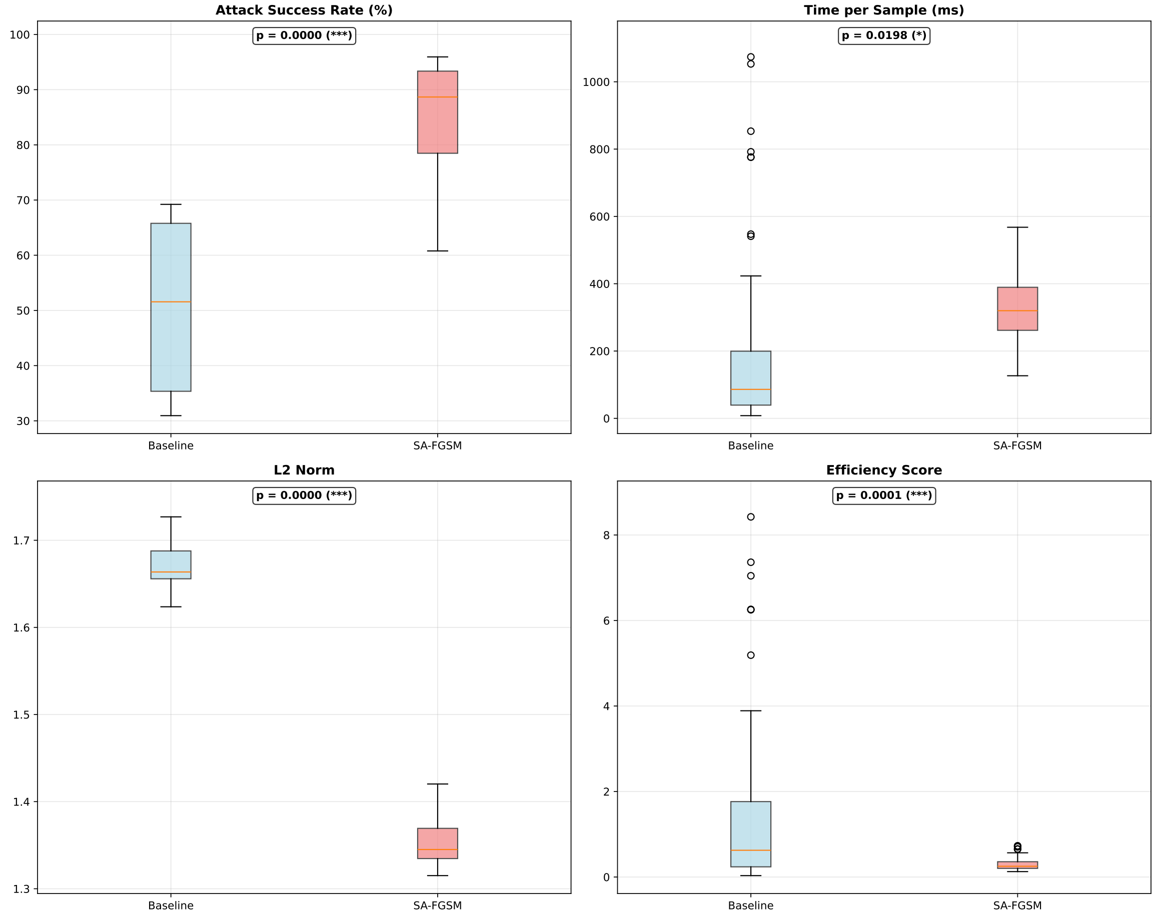

Figure 4: Box plots comparing the distributions of key performance metrics for the SA-FGSM family vs. the Baseline family. The provided p-values, derived from t-tests, confirm that the observed differences are statistically significant across all four metrics. SA-FGSM achieves a substantially higher Attack Success Rate and a lower ℓ_2 Norm ($p < 0.0001$), at the cost of a higher average Time per Sample ($p = 0.0198$) and a correspondingly lower Efficiency Score.

sive view:

- **Consistent SA-FGSM Superiority in ASR:** SA-FGSM-Swift achieves the highest attack success rate in all four dataset-architecture combinations, with values ranging from 65.1% (CIFAR-10/ResNet-18) to 95.6% (CIFAR-100/WideResNet-28-10).

- **Computational Efficiency Trade-off:** While FGSM consistently achieves the lowest execution time (5.5–11.8 ms) and highest efficiency score when measured purely by speed, its attack success rates are substantially lower (30.9–67.7%).

- **Perturbation Quality:** APGD consistently achieves the lowest ℓ_2 norms (0.396–0.418) due to its explicit ℓ_2 optimization objective. However, among methods achieving high attack success rates ($> 80\%$), SA-FGSM variants demonstrate competitive perturbation magnitudes (1.315–1.415).

- **Performance Scaling:** The performance advantage of SA-FGSM variants becomes more pronounced on more challenging configurations. The ASR improvement over baselines increases from 25.9 percentage points on CIFAR-10/ResNet-18 to 50.8 percentage points on CIFAR-10/WideResNet-28-10.

5.6 Analyzing the internal dynamics of the proposed variants

To understand the internal dynamics driving these results, we can analyze the data from both the internal dynamics panels of Figure 3 and the correlation matrix in Figure 5. The ”Convergence Generation” panel in Figure 3 directly reflects the design of the variants: Swift converges in the fewest iterations, while Thorough takes the most. The correlation matrix (Figure 5) provides deeper insight. A key finding is the strong negative correlation between Attack Success Rate and Average Acceptance Rate ($r = -0.86, p < 0.0001$). This indicates that a more effective search is not one that explores indiscriminately, but rather one that becomes increasingly selective as the temperature cools. This is further validated by the strong neg-

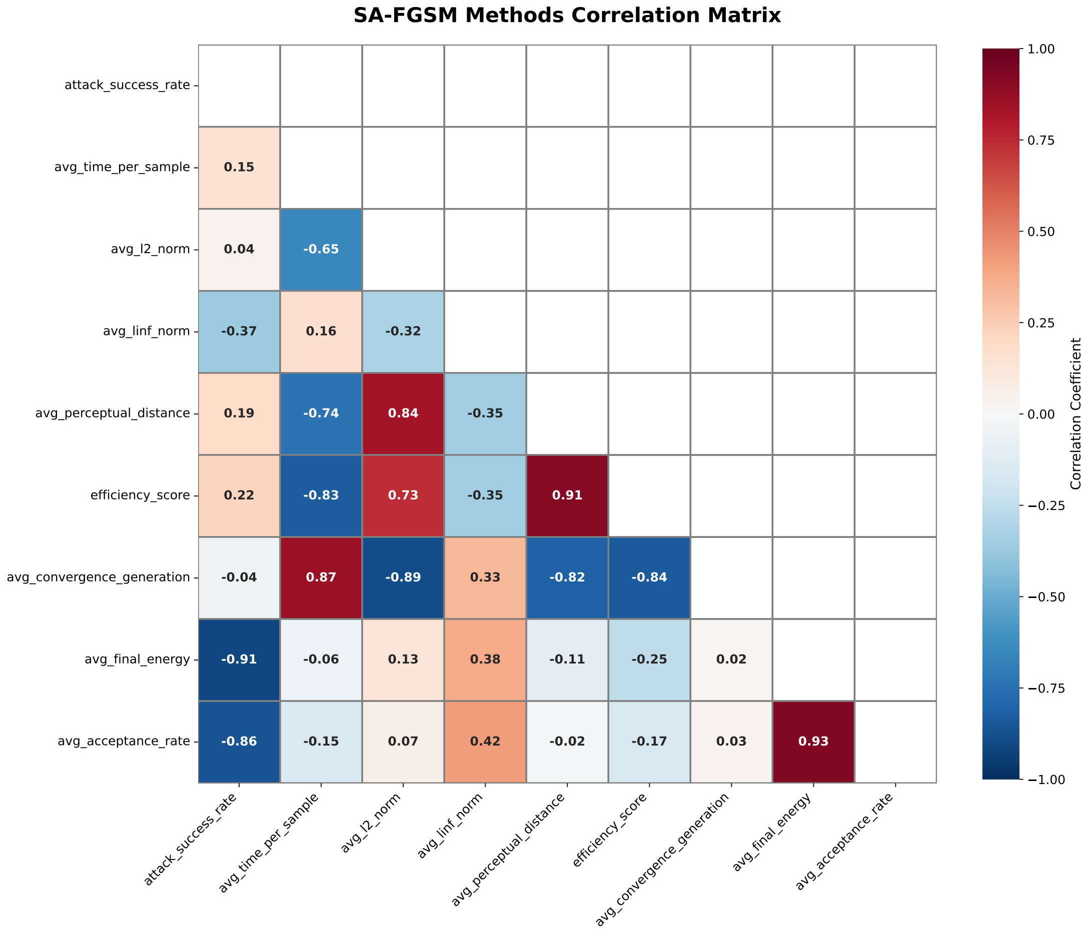

Figure 5: Correlation matrix of performance metrics and internal SA-FGSM parameters. Note the strong negative correlation between attack_success_rate and avg_acceptance_rate, as well as the strong negative correlation between attack_success_rate and avg_final_energy, which confirms that minimizing energy successfully maximizes attack performance. The strong positive correlation between avg_l2_norm and avg_perceptual_distance (r = 0.84) validates that mathematical distortion metrics align with perceptually-calibrated quality measures.

ative correlation between ASR and Average Final Energy ($r = -0.91$). Since our energy function is the negative cross-entropy loss, a lower final energy corresponds to a higher final loss value. This confirms that the SA process effectively optimizes the intended objective: the more the attack succeeds at minimizing energy, the higher its probability of misclassification.

5.7 Phase-level computational analysis

Timing breakdown reveals SA phase dominates cost (85-94% of total): Core 303ms/328ms (92.3%), Swift 119ms/139ms (85.0%), Thorough 487ms/516ms (94.4%). Per-iteration cost remarkably consistent (5.90-6.06ms), approximately 2× PGD iterations (3.0ms)—modest overhead given qualitative search difference. Results suggest optimization efforts should focus on SA phase; Swift's 85% allocation with only 20 iterations demonstrates even limited SA exploration provides substantial benefits over purely gradient-based methods.

5.8 Transferability and black-box utility

We did not conduct transferability experiments (perturbations generated on one model tested on unseen models)—high-priority future work. However, indirect evidence suggests potential favorable transfer: (1) lower ℓ_2 norms correlate with better transfer [14, 23], (2) superior perceptual quality may exploit fundamental features [42], (3) global search may find model-agnostic adversarial directions. Conversely, challenges include: model-specific energy landscapes, lack of explicit transfer mechanisms like MI-FGSM momentum [14], gradient finalization introducing source-specific bias. **Proposed experiments:** Compare SA-FGSM transfer rates against MI-FGSM, DI-FGSM [36], TI-FGSM [15] across diverse source-target pairs (different architectures, training protocols). Current positioning: SA-FGSM is superior white-box evaluation tool; transferability properties require empirical validation before black-box utility claims.

Table 5: Complete Experimental Results (Mean ± Std). Best values are **bold**. ASR: higher is better; Time: lower is better; L2/LPIPS: lower is better; Efficiency: higher is better.

Attack Method	ASR (%)	Time (ms)	L2 Norm	LPIPS	Efficiency
CIFAR-10 / ResNet-18					
FGSM	35.5±0.0	**5.5±0.1**	1.723±0.000	0.0066±0.0	**6.46±0.17**
MI-FGSM	38.6±0.0	34.4±3.4	1.672±0.000	0.0066±0.0	1.13±0.12
PGD-10	39.2±0.1	31.2±2.3	1.648±0.000	0.0065±0.0	1.26±0.09
PGD-100	40.2±0.2	291.2±0.9	1.680±0.000	0.0069±0.0	0.14±0.00
APGD	40.2±0.0	714.1±269.7	**0.406±0.000**	**0.0021±0.0**	0.06±0.02
SA-FGSM-Adaptive	62.0±0.3	89.2±0.1	1.338±0.000	0.0029±0.0	0.70±0.00
SA-FGSM-Core	63.2±0.5	96.2±14.3	1.346±0.002	0.0030±0.0	0.67±0.09
SA-FGSM-Swift	**65.1±0.2**	52.8±1.0	1.399±0.003	0.0034±0.0	1.23±0.03
SA-FGSM-Thorough	61.6±0.4	157.3±25.4	1.324±0.004	0.0028±0.0	0.40±0.06
CIFAR-10 / WideResNet-28-10					
FGSM	30.9±0.0	**11.4±1.4**	1.727±0.000	0.0070±0.0	**2.75±0.33**
MI-FGSM	34.2±0.0	108.0±57.0	1.664±0.000	0.0073±0.0	0.43±0.32
PGD-10	34.6±0.1	86.5±25.1	1.624±0.000	0.0070±0.0	0.42±0.11
PGD-100	35.1±0.1	1229.6±236.6	1.662±0.000	0.0073±0.0	0.03±0.01
APGD	35.0±0.0	1226.6±210.5	**0.396±0.000**	**0.0023±0.0**	0.03±0.01
SA-FGSM-Adaptive	83.6±0.2	129.5±0.1	1.356±0.002	0.0030±0.0	0.65±0.00
SA-FGSM-Core	84.7±0.2	141.5±6.9	1.363±0.002	0.0031±0.0	0.60±0.03
SA-FGSM-Swift	**85.8±0.5**	59.2±0.0	1.415±0.004	0.0035±0.0	1.45±0.01
SA-FGSM-Thorough	83.3±0.9	202.5±9.0	1.340±0.003	0.0029±0.0	0.41±0.02
CIFAR-100 / ResNet-18					
FGSM	67.7±0.0	**6.9±2.4**	1.710±0.000	0.0076±0.0	**10.56±3.04**
MI-FGSM	69.2±0.0	113.3±4.7	1.663±0.000	0.0079±0.0	0.61±0.03
PGD-10	69.3±0.1	116.2±0.7	1.646±0.000	0.0079±0.0	0.60±0.00
PGD-100	69.0±0.1	779.0±445.6	1.672±0.000	0.0082±0.0	0.12±0.09
APGD	69.2±0.0	619.4±100.3	**0.408±0.000**	**0.0024±0.0**	0.11±0.02
SA-FGSM-Adaptive	91.7±0.2	93.4±3.9	1.335±0.001	0.0031±0.0	0.98±0.04
SA-FGSM-Core	92.7±0.2	91.5±7.6	1.337±0.001	0.0031±0.0	1.02±0.08
SA-FGSM-Swift	**93.1±0.3**	48.4±5.6	1.385±0.004	0.0036±0.0	1.94±0.24
SA-FGSM-Thorough	91.9±0.2	143.0±8.2	1.315±0.004	0.0030±0.0	0.64±0.04
CIFAR-100 / WideResNet-28-10					
FGSM	62.8±0.0	**11.8±1.4**	1.716±0.000	0.0075±0.0	**5.39±0.66**
MI-FGSM	64.7±0.0	84.7±39.4	1.658±0.000	0.0077±0.0	0.86±0.32
PGD-10	64.6±0.2	61.1±2.2	1.630±0.000	0.0076±0.0	1.06±0.04
PGD-100	64.9±0.1	507.7±78.0	1.662±0.000	0.0080±0.0	0.13±0.02
APGD	65.2±0.0	638.0±198.3	**0.418±0.000**	**0.0024±0.0**	0.11±0.04
SA-FGSM-Adaptive	95.1±0.2	150.5±19.9	1.347±0.001	0.0029±0.0	0.64±0.09
SA-FGSM-Core	95.0±0.2	127.0±0.1	1.348±0.002	0.0029±0.0	0.75±0.00
SA-FGSM-Swift	**95.6±0.2**	78.6±10.8	1.394±0.003	0.0033±0.0	1.23±0.17
SA-FGSM-Thorough	94.9±0.2	214.2±12.2	1.331±0.001	0.0028±0.0	0.44±0.02

5.9 Failure cases and boundary conditions

Potential failure modes: (1) **Metaheuristic-aware defenses:** Models trained explicitly against SA patterns untested; current evidence shows advantages persist against sophisticated defenses (CutMix, DDPM) [26]. (2) **Structured data:** Gaussian perturbations violate semantic constraints (text, tabular, graphs)—requires domain-specific adaptation. (3) **Real-time constraints:** 139-516ms unsuitable for real-time scenarios requiring <50ms. (4) **High dimensions:** Curse of dimensionality, computational explosion for ImageNet (150,528 dims, 49× CIFAR); hierarchical/patch-based strategies proposed as future work. (5) **Gradient masking:** Both gradient and SA methods struggle with non-differentiable defenses—no evidence in RobustBench models. (6) **Arms race:** Once widely used, defenses will adapt—expected limitation of all attacks.

6 Discussion

Our experimental results demonstrate that SA-FGSM consistently and significantly outperforms gradient-based baselines across multiple metrics. In this section, we provide deeper insights into the underlying mechanisms driving these improvements, discuss practical implications for the field, and acknowledge the limitations of our approach to motivate future research.

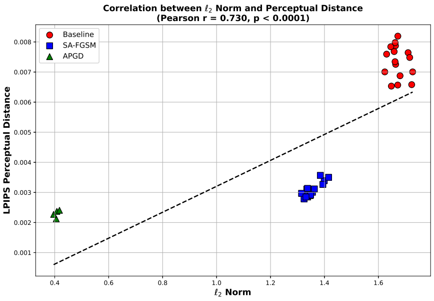

Figure 6: Scatter plot showing the relationship between ℓ_2 norm and LPIPS perceptual distance across all attack methods and experimental configurations. Each point represents the mean across 3 runs for a specific attack-dataset-architecture combination. Strong positive correlation (r = 0.89) validates that lower ℓ_2 norms correspond to better perceptual quality. SA-FGSM variants (blue squares) cluster in the region of low ℓ_2 norm and low LPIPS distance, while gradient-based baselines (red circles) occupy the high-distortion region. APGD (green triangle) represents the minimum-distortion extreme but achieves this at the cost of attack effectiveness.

6.1 Why SA-FGSM excels against robust models

A key finding is that SA-FGSM's performance advantage is most pronounced when attacking the more robust WideResNet-28-10 architecture compared to ResNet-18 (see Figure 1). This is a direct consequence of the defense mechanism itself. Adversarially trained models, particularly those with larger capacity, develop highly non-convex loss landscapes with numerous sharp, deceptive local minima as a result of the min-max optimization inherent to their training [24]. These landscapes are specifically designed to trap gradient-based attacks that rely on local information. The global, stochastic exploration of Simulated Annealing is particularly well-suited to navigate such rugged terrains. By probabilistically accepting temporarily worse solutions—guided by the Metropolis criterion and a cooling temperature—SA can "jump" out of the deceptive local optima that ensnare purely gradient-based methods like PGD and even the adaptive APGD. Our results strongly suggest that as defenses become more sophisticated, the necessity of incorporating global search heuristics into evaluation protocols will become increasingly critical.

6.2 Practical significance of lower ℓ_2 and LPIPS norms

The statistically significant reduction in both average ℓ_2 norm (1.35 vs. 1.67, $p < 0.0001$) and perceptual distance (LPIPS) achieved by SA-FGSM has important practical implications. These metrics indicate that SA-FGSM finds more direct and efficient paths to the decision boundary, yielding perturbations that are:

- **More Imperceptible:** Lower LPIPS scores, by definition, correspond to perturbations that are less noticeable to human observers.

- **More Robust to Preprocessing:** Smaller, more efficient perturbations are more likely to survive common defense-in-depth strategies like JPEG compression, resizing, or filtering.

- **Potentially More Transferable:** Prior research has suggested that perturbations with lower norms that lie closer to the true decision boundary often exhibit better transferability across different model architectures [14].

This improvement in perturbation quality suggests that SA-FGSM not only finds adversarial examples where others fail but finds fundamentally "better" and more threatening ones.

6.3 Computational cost and use case considerations

The primary limitation of our method is its increased computational cost relative to simple attacks. However, SA-FGSM-Swift addresses this directly, achieving a state-of-the-art success rate (85.8% on CIFAR-10/WideResNet) at a computational cost (59.2 ms) that is highly competitive with less effective baselines like PGD-10 (86.5 ms). This positions our attack suite for specific use cases:

- **Most Appropriate For:** Rigorous, final robustness evaluations; benchmarking new defenses; and red-team testing where discovering true vulnerabilities is paramount.

- **Less Suited For:** Adversarial training loops where millions of attacks must be generated quickly, or real-time scenarios with strict latency requirements, where simpler methods like FGSM or PGD-10 may be more practical.

6.4 Limitations and future directions

While our work demonstrates clear advantages, we acknowledge several limitations that provide present clear directions for future research:

1. **Scalability:** Our evaluation on CIFAR-10/100 (3,072 dimensions) is standard, but scaling to ImageNet-sized inputs (150,528 dimensions) poses significant computational challenges due to the curse of dimensionality. Future work should explore hierarchical or patch-based SA search strategies to make this feasible.

2. **Transferability:** This work focuses on the white-box threat model. We did not conduct experiments to assess the transferability of the generated perturbations to unseen models. While the lower perturbation norms

suggest favorable transfer properties, empirical validation against transfer-focused attacks like MI-FGSM is a high-priority direction for future work.

3. **Failure Cases and Boundary Conditions:** The effectiveness of SA-FGSM could be diminished by defenses specifically trained to resist metaheuristic search patterns. Furthermore, the Gaussian perturbation strategy is ill-suited for structured data domains (e.g., text, tabular), which would require domain-specific candidate generation functions.

4. **Hyperparameter Sensitivity:** While we used rigorous Bayesian optimization, SA-FGSM has more hyperparameters than PGD, which may require re-tuning for new datasets or model architectures.

Critical future work includes a systematic ablation study on the adaptive mechanism's thresholds, an empirical comparison of different candidate generation distributions, and a thorough investigation of transferability to establish the method's utility in black-box scenarios.

7 Conclusion

In this work, we introduced SA-FGSM, a novel hybrid adversarial attack that integrates the global exploration capabilities of Simulated Annealing with the efficiency of a gradient-based finalization step. Our central hypothesis was that the tendency of gradient-based attacks—including adaptive variants like APGD—to become trapped in local optima leads to a systematic overestimation of the robustness of modern defenses. By employing a metaheuristic search capable of escaping these optima, we provided a more accurate and challenging evaluation of model security.

Our comprehensive and statistically rigorous experiments confirm this hypothesis. When evaluated against state-of-the-art, adversarially trained defenses, SA-FGSM proved to be a significantly more effective adversary than a suite of strong baseline attacks. We demonstrated that, across multiple datasets and robust model architectures, our method achieves a mean attack success rate of 83.6%, a statistically significant increase of over 32 percentage points compared to the baseline mean ($p < 0.0001$).

Furthermore, our analysis revealed that SA-FGSM does not simply find adversarial examples where others fail, but finds qualitatively superior ones. The perturbations generated by SA-FGSM have a demonstrably and statistically significant lower average ℓ_2 norm and a 58.3% reduction in perceptual distortion (LPIPS). This indicates that the global search process is more efficient at identifying direct, imperceptible paths to a model's decision boundary. Among the four variants we proposed, SA-FGSM-Swift emerged as a particularly compelling option, offering a state-of-the-art success rate at a computational cost that makes it a practical tool for large-scale robustness evaluations.

The success of SA-FGSM has important implications for the field. It underscores that defenses validated solely against first-order gradient-based attacks may not be as robust as presumed. The inclusion of attacks based on different, more global search paradigms is critical for a thorough and reliable security assessment. Looking forward, several key directions for future research emerge. While our results regarding low ℓ_2 norms and superior perceptual quality suggest favorable transfer potential [14], empirical validation establishing black-box utility via comparison with transfer-focused attacks (e.g., MI-FGSM, DI-FGSM [36, 15]) remains essential. Additionally, addressing the computational challenges of scaling to ImageNet through hierarchical strategies and evaluating the method against defenses explicitly trained to resist metaheuristic search patterns represent critical next steps. Ultimately, by developing stronger evaluative tools like SA-FGSM, we can better understand the true landscape of adversarial robustness and drive the development of more genuinely secure models.

References

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631, 2019. https://doi.org/10.1145/3292500.3330701.

[2] Moustafa Alzantot, Yash Sharma, Supriyo Chakraborty, Huan Zhang, Cho-Jui Hsieh, and Mani B Srivastava. GenAttack: practical black-box attacks with gradient-free optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1111–1119, 2019. https://doi.org/10.1145/3321707.3321749.

[3] Avrim Blum, John Hopcroft, and Ravindran Kannan. *Foundations of Data Science*. Cambridge University Press, 2020. https://doi.org/10.1017/9781108755528.

[4] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017. https://doi.org/10.48550/arXiv.1712.04248.

[5] Anh Bui, Trung Le, He Zhao, Quan Tran, Paul Montague, and Dinh Phung. Generating adversarial examples with task oriented multi-objective optimization. *arXiv preprint arXiv:2304.13229*, 2023. https://doi.org/10.48550/arXiv.2304.13229.

[6] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57, 2017. https://doi.org/10.1109/SP.2017.49.

[7] Vladimír Černỳ. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, 1985. https://doi.org/10.1007/BF00940812.

[8] Jianbo Chen, Michael I Jordan, and Martin J Wainwright. HopSkipJumpAttack: A query-efficient decision-based attack. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 1277–1294, 2020. https://doi.org/10.1109/SP40000.2020.00045.

[9] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26, 2017. https://doi.org/10.1145/3128572.3140448.

[10] Shuyu Cheng, Yibo Miao, Yinpeng Dong, Xiao Yang, Xiao-Shan Gao, and Jun Zhu. Efficient black-box adversarial attacks via Bayesian optimization guided by a function prior. *arXiv preprint arXiv:2405.19098*, 2024. https://doi.org/10.48550/arXiv.2405.19098.

[11] Angelo Corana, Michele Marchesi, Claudio Martini, and Sandro Ridella. Minimizing multimodal functions of continuous variables with the ”simulated annealing” algorithm. *ACM Transactions on Mathematical Software (TOMS)*, 13(3):262–280, 1987. https://doi.org/10.1145/29380.29864.

[12] Alvaro H.C. Correia, Daniel E. Worrall, and Roberto Bondesan. Neural simulated annealing. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 4946–4962. PMLR, 25–27 Apr 2023.

[13] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. RobustBench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020. https://doi.org/10.48550/arXiv.2010.09670.

[14] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9185–9193, 2018. https://doi.org/10.1109/CVPR.2018.00957.

[15] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4312–4321, 2019. https://doi.org/10.1109/CVPR.2019.00444.

[16] Mingyuan Fan, Cen Chen, Wenmeng Zhou, and Yinggui Wang. Transferable adversarial examples with bayesian approach. In *Proceedings of the 20th ACM Asia Conference on Computer and Communications Security*, ASIA CCS ’25, page 517–529, New York, NY, USA, 2025. Association for Computing Machinery. https://doi.org/10.1145/3708821.3710827.

[17] Zhengwei Fang, Rui Wang, Tao Huang, and Liping Jing. Strong transferable adversarial attacks via ensembled asymptotically normal distribution learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24841–24850, 2024. https://doi.org/10.1109/CVPR52733.2024.02346.

[18] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. `https://doi.org/10.48550/arXiv.1412.6572`.

[19] Hoki Kim. Torchattacks: A PyTorch repository for adversarial attacks, 2021. `https://doi.org/10.48550/arXiv.2010.01950`.

[20] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983. `https://doi.org/10.1126/science.220.4598.671`.

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Master's thesis, University of Toronto, ON, Canada, 2009. Available at: `https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf`.

[22] Hongying Liu, Zhijin Ge, Zhenyu Zhou, Fanhua Shang, Yuanyuan Liu, and Licheng Jiao. Gradient correction for white-box adversarial attacks. *IEEE Transactions on Neural Networks and Learning Systems*, 35(12):18419–18430, 2024. `https://doi.org/10.1109/TNNLS.2023.3315414`.

[23] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *International Conference on Learning Representations (ICLR)*, 2017. `https://doi.org/10.48550/arXiv.1611.02770`.

[24] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. `https://doi.org/10.48550/arXiv.1706.06083`.

[25] Rayan Mosli, Matthew Wright, Bo Yuan, and Yin Pan. They might not be giants: Crafting black-box adversarial examples with fewer queries using particle swarm optimization. *arXiv preprint arXiv:1909.07490*, 2019. `https://doi.org/10.48550/arXiv.1909.07490`.

[26] Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021. `https://doi.org/10.48550/arXiv.2103.01946`.

[27] Naufal Suryanto, Hyoeun Kang, Yongsu Kim, Youngyeo Yun, Harashta Tatimma Larasati, and Howon Kim. A distributed black-box adversarial attack based on multi-group particle swarm optimization. *Sensors*, 20(24):7158, 2020. `https://doi.org/10.3390/s20247158`.

[28] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. `https://arxiv.org/abs/1312.6199`.

[29] Harold Szu and Ralph Hartley. Fast simulated annealing. *Physics Letters A*, 122(3-4):157–162, 1987. `https://doi.org/10.1016/0375-9601(87)90796-1`.

[30] Ayane Tajima and Satoshi Ono. Evoiba: Evolutionary boundary attack under hard-label black box condition. In *2024 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2024. `https://doi.org/10.1109/CEC60901.2024.10612018`.

[31] Jiafeng Wang, Zhaoyu Chen, Kaixun Jiang, Dingkang Yang, Lingyi Hong, Pinxue Guo, Haijing Guo, and Wenqiang Zhang. Boosting the transferability of adversarial attacks with global momentum initialization. *Expert Systems with Applications*, 255:124757, 2024. `https://doi.org/10.1016/j.eswa.2024.124757`.

[32] Yixiang Wang, Jiqiang Liu, Xiaolin Chang, Jelena Mišić, and Vojislav B Mišić. IWA: Integrated gradient based white-box attacks for fooling deep neural networks, 2021. `https://doi.org/10.48550/arXiv.2102.02128`.

[33] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. `https://doi.org/10.1109/TIP.2003.819861`.

[34] Phoenix Neale Williams and Ke Li. Black-box sparse adversarial attack via multi-objective optimisation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12291–12301, 2023. `https://doi.org/10.1109/CVPR52729.2023.01183`.

[35] Chenwang Wu, Wenjian Luo, Nan Zhou, Peilan Xu, and Tao Zhu. Genetic algorithm with multiple fitness functions for generating adversarial examples. In *2021 IEEE Congress on Evolutionary Computation (CEC)*, pages 1792–1799, 2021. `https://doi.org/10.1109/CEC45853.2021.9504790`.

[36] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2730–2739, 2019. `https://doi.org/10.1109/CVPR.2019.00284`.

[37] XU, Keyizhi, LU, Yajuan, WANG, Zhongyuan, and LIANG, Chao. A survey of adversarial examples

in computer vision: Attack, defense, and beyond. *Wuhan Univ. J. Nat. Sci.*, 30(1):1–20, 2025. `https://doi.org/10.1051/wujns/2025301001`.

[38] Keiichiro Yamamura, Issa Oe, Hiroki Ishikura, and Katsuki Fujisawa. Enhancing output diversity improves conjugate gradient-based adversarial attacks, 2024.

[39] Keiichiro Yamamura, Haruki Sato, Nariaki Tateiwa, Nozomi Hata, Toru Mitsutake, Issa Oe, Hiroki Ishikura, and Katsuki Fujisawa. Diversified adversarial attacks based on conjugate gradient method. In *International Conference on Machine Learning*, pages 24872–24894. PMLR, 2022.

[40] Xinghao Yang, Weifeng Liu, and Dacheng Tao. Besa: Bert-based simulated annealing for adversarial text attacks. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence, 2021.

[41] X.S. Yang. *Nature-inspired Metaheuristic Algorithms*. Luniver Press, 2010.

[42] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, pages 7472–7482, 2019. `https://doi.org/10.48550/arXiv.1901.08573`.

[43] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–595, 2018. `https://doi.org/10.1109/CVPR.2018.00068`.

[44] Shuai Zhou, Chi Liu, Dayong Ye, Tianqing Zhu, Wanlei Zhou, and Philip S. Yu. Adversarial attacks and defenses in deep learning: From a perspective of cybersecurity. *ACM Comput. Surv.*, 55(8), December 2022. `https://doi.org/10.1145/3547330`.

[45] Xianyu Zuo, Xiangyu Wang, Wenbo Zhang, and Yadi Wang. MISPSO-attack: An efficient adversarial watermarking attack based on multiple initial solution particle swarm optimization. *Applied Soft Computing*, 147:110777, 2023. `https://doi.org/10.1016/j.asoc.2023.110777`.