FAME: FORMAL ABSTRACT MINIMAL EXPLANATION FOR NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose **FAME** (Formal Abstract Minimal Explanations), a new class of abductive explanations grounded in abstract interpretation. FAME is the first method to scale to large neural networks while reducing explanation size. Our main contribution is the design of dedicated perturbation domains that eliminate the need for traversal order. FAME progressively shrinks these domains and leverages LiRPA-based bounds to discard irrelevant features, ultimately converging to a **formal abstract minimal explanation**. To assess explanation quality, we introduce a procedure that measures the worst-case distance between an abstract minimal explanation and a true minimal explanation. This procedure combines adversarial attacks with an optional VERIX+ refinement step. We benchmark FAME against VERIX+ and demonstrate consistent gains in both explanation size and runtime on medium- to large-scale neural networks.

1 INTRODUCTION

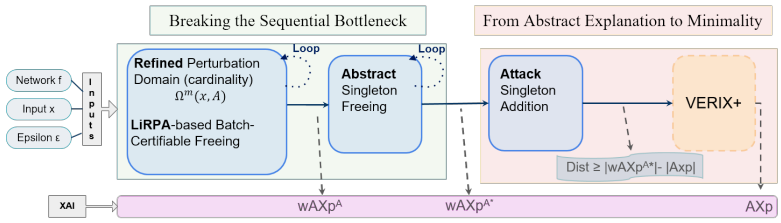

Figure 1: **FAME Framework.** The pipeline operates in two main phases **(1) Abstract Batch Freeing** phase leverages abstract interpretation (LiRPA) to simultaneously free a large number of irrelevant features (Section 4.2) based on an iterative process operating within a refined, cardinality-constrained perturbation domain, $\Omega^m(x, A)$ (Eq. 5); To ensure that the final explanation is as small as possible, the remaining features that could not be freed in batches are tested individually (Section 5). **(2) From Abstract to Minimal** phase identifies the final necessary features using singleton addition attacks and, if needed, a final run of VERIX+ (Section 6). The difference in size, $|WAXp^{A*}| - |AXp|$, serves as a metric to evaluate the efficiency of phase 1.

Neural network-based systems are being applied across a wide range of domains. Given AI tools' strong capabilities in complex analytical tasks, a significant portion of these applications now involves tasks that require reasoning. These tools often achieve impressive results in problems requiring intricate analysis to reach correct conclusions. Despite these successes, a critical challenge remains: understanding the reasoning behind neural network decisions. The internal logic of a neural network is often opaque, with its conclusions presented without accompanying justifications. This lack of transparency undermines the trustworthiness and reliability of neural networks, especially in high-stakes or regulated environments. Consequently, the need for interpretable and explainable AI (XAI) has become a growing focus in recent research.

Two main approaches have emerged to address this challenge. The first employs statistical and heuristic techniques to infer explanations based on network's internal representations (12). The

second leverages automated reasoners and formal verification methods to produce provably correct explanations grounded in logical reasoning. While statistical methods are generally faster and more scalable, formal verification techniques provide stronger guarantees about the correctness of their explanations.

Here, we use the term "formal XAI" to refer to a family of concepts including minimal explanations, also known as local-minimal, minimal unsatisfiable subsets (29), prime implicants (35) or abductive explanations (AXp) (20). These explanations characterize feature sets where the removal of any single feature invalidates the explanation. In a machine learning context, they represent subsets of input features that preserve robustness. However, a major hurdle for formal XAI is its high computational cost due to the complexity of reasoning, preventing it from scaling to large neural networks (NNs) (31). This limitation, combined with the scarcity of open-source libraries, significantly hinders its adoption. Initial hybrid approaches, such as the EVA method (11), have attempted to combine formal and statistical methods, but these often fail to preserve the mathematical properties of the explanation. However, robustness-based approaches address the scalability challenges of formal XAI for NN by leveraging a fundamental connection between AXps and adversarial examples (15).

In this work, we present FAME, a scalable framework for formal XAI that addresses the core limitations of existing methods. Our contributions are fourfold:

- **Formal abstract explanations.** We introduce the first class of abductive explanations derived from abstract interpretation, enabling explanation algorithms to handle high-dimensional neural networks.
- **Eliminating traversal order.** We design perturbation domains and a recursive refinement procedure that leverage Linear Relaxation based Perturbation Analysis (LiRPA)-based certificates to simultaneously discard multiple irrelevant features. This removes the sequential bottleneck inherent in prior work and yields an abstract minimal explanation.
- **Provable quality guarantees.** We provide the first procedure to measure the worst-case gap between abstract minimal explanations and true minimal abductive explanations, combining adversarial search with optional VERIX+ refinement.
- **Scalable evaluation.** We benchmark FAME on medium- and large-scale neural networks, showing consistent improvements in both explanation size and runtime over VERIX+. We release our framework as open source to facilitate further research.

2 Abductive Explanations & Verification

2.1 Notations

Scalars are denoted by lower-case letters (e.g., x), and the set of real numbers by $\mathbb{R}$. Vectors are denoted by bold lower-case letters (e.g., $\mathbf{x}$), and matrices by upper-case letters (e.g., W). The i-th component of a vector $\mathbf{x}$ (resp. line of a matrix W) is written as $\mathbf{x}_i$ (resp. W_i). The matrix $W^{\geq 0}$ (resp. $W^{\leq 0}$) represents the same matrix with only nonnegative (resp. nonpositive) weights. Sets are written in calligraphic font (e.g., $\mathcal{S}$). We denote the perturbation domain by Ω and the property to be verified by P.

2.2 The Verification Context

We consider a neural network as a function $f : \mathbb{R}^n \to \mathbb{R}^k$. The core task of verification is to determine whether the network's output $f(x')$ satisfies a given property P for every possible input x' within a specified domain $\Omega(x) \subseteq \mathbb{R}^n$. When verification fails, it means there is at least one input x' in the domain $\Omega(x)$ that violates the property P (a counterexample). The verification task can be written as: $\forall x' \in \Omega(x),$ does $f(x')$ satisfy P? This requires defining two components:

1. **The Perturbation Domain** (Ω): This domain defines the set of perturbations. It is often an l_p-norm ball around a nominal input x, such as an l_∞ ball for modeling imperceptible noise: $\Omega = \{\mathbf{x}' \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{x}'\|_\infty \leq \epsilon\}$.
2. **The Property** (P): This is the specification the network must satisfy. For a classification task where the network correctly classifies an input $\mathbf{x}$ into class c, the standard robustness

property P asserts that the logit for class c remains the highest for any perturbed input $\mathbf{x}'$:

$$\mathrm{P}(\mathbf{x}') \equiv \min_{i \neq c} \{f_c(\mathbf{x}') - f_i(\mathbf{x}')\} > 0 \tag{1}$$

A large body of work has investigated formal verification of neural networks, with adversarial robustness being the most widely studied property (39). Numerous verification tools are now available off-the-shelf, and for piecewise-linear models f with corresponding input domains and properties, exact verification is possible (25; 28). In practice, however, exact methods quickly become intractable for realistic networks, so most approaches rely on relaxations that trade precision for efficiency. A common relaxation strategy is to bound approximation errors using abstract interpretation or linear perturbation analysis (44; 36). These methods over-approximate the network's output by enclosing it between two affine functions, given knowledge of the architecture and weights. Such abstractions enable sound but conservative verification: if the relaxed property holds, the original one is guaranteed to hold as well.

2.3 Abductive Explanations: Pinpointing the "Why"

Understanding Model Robustness with Formal Explanations: Neural networks often exhibit sensitivity to minor input perturbations, a phenomenon that certified training can mitigate but not eliminate (9). Even robustly trained models may only have provably safe regions spanning a few pixels for complex tasks like ImageNet classification (34). To build more reliable systems, it is crucial to understand *why* a model's prediction is robust (or not) within a given context. Formal explainability provides a rigorous framework for this analysis.

We focus on *abductive explanations* (AXps, also called distance-restricted explanations (ϵ-AXp)) (20; 15), which identify a subset of input features that are *sufficient* to guarantee that the property P holds. Formally, a local formal abductive explanation is defined as a subset of input features that, if collapsed to their nominal values (i.e., the sample $\mathbf{x}$), ensure that the local perturbation domain Ω surrounding the sample contains no counterexamples.

Definition 2.1 (Weak Abductive Explanation (WAXp))**.** Formally, given a triple $(\mathbf{x}, \Omega, \mathrm{P})$, an *explanation* is a subset of feature indices $\mathcal{X} \subseteq \mathcal{F} = \{1, \ldots, n\}$ such that

$$\text{WAXp: } \forall \mathbf{x}' \in \Omega(\mathbf{x}), \quad \left(\bigwedge_{i \in \mathcal{X}} (\mathbf{x}'_i = \mathbf{x}_i) \right) \implies f(\mathbf{x}') \models \mathrm{P}. \tag{2}$$

While many such explanations may exist (the set of all features $\mathcal{F}$ is a trivial one), the most useful explanations are the most concise ones (4). We distinguish between three levels of conciseness:

Minimal Explanation: An explanation $\mathcal{X}$ is *minimal* if removing any single feature from it would break the guarantee (i.e., $\mathcal{X} \setminus \{j\}$ is no longer an explanation for any $j \in \mathcal{X}$). These are also known as minimal unsatisfiable subsets(19; 4).

Minimum Explanation: An explanation $\mathcal{X}$ is *minimum* if it has the smallest possible number of features (cardinality) among all possible minimal explanations.

Figure 2 illustrates a 3D classification task. For the starred sample, we seek an explanation for its classification within a local cube-shaped domain. As shown in Figure 3, fixing only feature $\mathbf{x}_2$ (i.e. freeing $\{\mathbf{x}_1, \mathbf{x}_3\}$, restricting perturbations to the orange plane) is not enough to guarantee the property, since a counterexample exists. However, fixing both $\mathbf{x}_2$ and $\mathbf{x}_3$ (orange line on free x_1) defines a 'safe' subdomain where the desired property holds true, since no counterexample exists in that subdomain. Therefore, $\mathcal{X} = \{\mathbf{x}_2, \mathbf{x}_3\}$ is an abductive explanation. Since neither $\{\mathbf{x}_2\}$ nor $\{\mathbf{x}_3\}$ are explanations on their own, $\{\mathbf{x}_2, \mathbf{x}_3\}$ is minimal. But it is not minimum since $\mathcal{X} = \{\mathbf{x}_1\}$ is also a minimal abductive explanation with a smaller cardinality. Two special cases are worth noting: an empty explanation (all features are irrelevant) and a full explanation (the entire input is necessary). In the rest of this paper, we will use the terms abductive explanation or formal explanation and the notation WAXp to refer to Definition 2.1.

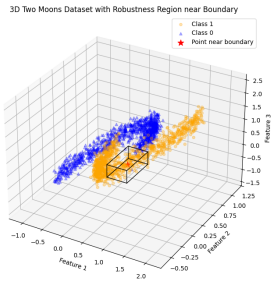

Figure 2: A 3D classification task.

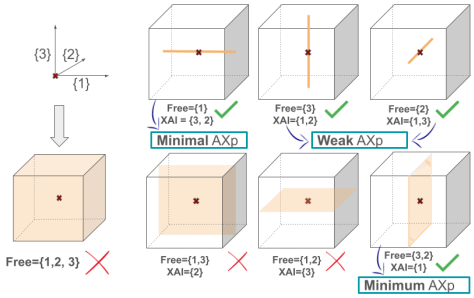

Figure 3: AXps with different properties.

3 Related Work

Substantial progress has been made in the practical efficiency of computing formal explanations. While finding an abductive explanation (AXp) is tractable for some classifiers (30; 8; 13; 14; 22; 32; 33), it becomes computationally hard for complex models like random forests and neural networks (18; 23). This is often because these methods encode the problem as a logical formula, leveraging automated reasoners like SAT, SMT, and Mixed Integer Linear Programming (MILP) solvers (1; 16; 17; 18; 23). Early approaches, such as deletion-based (7) and insertion-based (37) algorithms, are inherently sequential, thus requiring an ordering of the input features traditionally denoted as *traversal ordering*. They require a number of costly verification calls linear with the number of features, which prevents effective parallelization. As an alternative, surrogate models have been used to compute formal explanations for complex models (5), but the guarantee does not necessary hold on the original model.

Recent work aims to break the sequential bottleneck, by linking explainability to adversarial robustness and formal verification. DistanceAXp (15; 27) is a key example, aligning with our definition of AXp and enabling the use of verification tools.

The latest literature focuses on breaking the sequential bottleneck using several strategies that include parallelization. This is achieved either by looking for several counterexamples at once (21; 4) or by identifying a set of irrelevant features simultaneously, as seen in VERIX (41), VERIX+ (40), and prior work (4). For instance, VERIX+ introduced stronger traversal strategies to alleviate the sequential bottleneck. Their binary search approach splits the remaining feature set and searches for batches of consecutive irrelevant features, yielding the same result as sequential deletion but with fewer solver calls. They also adapted QuickXplain (24), which can produce even smaller explanations at the cost of additional runtime by verifying both halves. Concurrently, (4) proposed strategies like the singleton heuristic to reuse verification results and derived provable size bounds, but their approach remains significantly slower than VERIX and lacks publicly available code.

The identified limitations are twofold. First, existing methods rely heavily on exact solvers such as Marabou (26), which do not scale to large NNs and are restricted to CPU execution. Recent verification benchmarks (6; 10; 45) consistently demonstrate that GPU acceleration and distributed verification are indispensable for achieving scalability. Second, these approaches critically depend on traversal order. As shown in VERIX, the chosen order of feature traversal strongly impacts both explanation size and runtime. Yet, determining an effective order requires prior knowledge of feature importance, precisely the information that explanations are meant to uncover, thus introducing a circular dependency. Nevertheless, VERIX+ currently represents the SOTA for abductive explanations in NNs, achieving the best trade-off between explanation size and computation time.

Our work builds on this foundation by directly addressing the sequential bottleneck of formal explanation without requiring a traversal order, a first in formal XAI. We demonstrate that leveraging incomplete verification methods and GPU hardware is essential for practical scalability. Our approach offers a new solution to the core scalability issues, complementing other methods that aim to reduce explanation cost through different means (3; 2).

4 FAME: FORMAL ABSTRACT MINIMAL EXPLANATION

In this section, we leverage abstract interpretation (LiRPA) to build an *abstract abductive explanation*, as defined in Definition 4.1.

Definition 4.1 (Abstract Abductive Explanation (WAXp^A))**.** Formally, given a triple $(\mathbf{x}, \Omega, \mathrm{P})$, an *abstract abductive explanation* is a subset of feature indices $\mathcal{X}^A \subseteq \mathcal{F} = \{1, \ldots, n\}$ such that, under an abstract interpretation $\overline{f}$ of the model f, the following holds:

$$\text{WAXp}^A : \forall \mathbf{x}' \in \Omega(\mathbf{x}), \quad \left(\bigwedge_{i \in \mathcal{X}^A} (\mathbf{x}'_i = \mathbf{x}_i) \right) \implies \overline{f}(\mathbf{x}') \models \mathrm{P}. \tag{3}$$

Here, $\overline{f} = \text{LiRPA}(f, \Omega)$ denotes the sound over-approximated bounds of the model outputs on the domain Ω, as computed by the LiRPA method. If Eq. (3) holds, any feature outside $\mathcal{X}^A$ can be considered irrelevant *with respect to the abstract domain*. This ensures that the concrete implication $f(\mathbf{x}') \models \mathrm{P}$ also holds for all $x' \in \Omega$. In line with the concept of abductive explanations, we define an *abstract minimal explanation* as an abstract abductive explanation (WAXp$^{A^\star}$) a set of features $\mathcal{X}^A$ from which no feature can be removed without violating Eq. (3).

Due to the over-approximation, as detailed in Section 2.2, any *abstract abductive explanation* is a *weak abductive explanation* for the model f. In the following we present the first steps described in Figure 1 to build such a WAXp^A.

4.1 THE ASYMMETRY OF PARALLEL FEATURE SELECTION

In the context of formal explanations, **adding a feature** means identifying it as essential to a model's decision (causes the model to violate the desired property P), so its value must be fixed. Conversely, **freeing a feature** means identifying it as irrelevant, allowing it to vary without affecting the prediction. A key insight is the asymmetry between these two actions: while adding necessary features can be parallelized naively, freeing features cannot due to complex interactions.

The core problem of parallelizing the freeing of features is that it's unsound to free multiple features at once based only on individual verification queries, as two features may be individually irrelevant yet jointly critical. This failure stems from treating the verifier as a simple binary oracle (SAT/UNSAT), which hides the information about feature dependencies. The formal propositions detailing the asymmetry of parallel feature selection are provided in the Appendix B.

To overcome this limitation, we propose a sound method in the next section that simultaneously frees several features by leveraging abstract interpretation.

4.2 ABSTRACT INTERPRETATION FOR SIMULTANEOUS FREEING

Standard solvers act as a "binary oracle," and their outcomes (SAT/UNSAT) are insufficient to certify batches of features for freeing without a traversal order. This is because of feature dependencies and the nature of the verification process. We adress this by leveraging *inexact* verifiers based on abstract interpretation (LiRPA) to extract *proof objects*—linear bounds that conservatively track the contribution of any feature set. Specifically, we use CROWN(44) to define an *abstract batch certificate* Φ in Definition 4.1. If one succeeds in freeing a set of features $\mathcal{A}$ given Φ, we denote such an explanation as a *formal abstract explanation* that satisfies Proposition 4.1.

Definition 4.2 (Abstract Batch Certificate)**.** Let $\mathcal{A}$ be a set of features and Ω any perturbation domain. The *abstract batch certificate* is defined as:

$$\Phi(\mathcal{A}; \Omega) \;=\; \max_{i \neq c} \Big(\overline{b}^i(x) + \sum_{j \in \mathcal{A}} c_{i,j} \Big),$$

where the baseline bias (worst-case margin of the model's output) at x is $\overline{b}^i(x) = \overline{W}^i \cdot x + \overline{w}^i$,

and the contribution of each feature $j \in \mathcal{A}$ is $c_{i,j} \;=\; \max \left\{ \overline{W}_j^{i,\geq 0} \, (\overline{x}_j - x_j),\ \overline{W}_j^{i,\leq 0} \, (\underline{x}_j - x_j) \right\}$,

with $\overline{x}_j = \max\{x'_j : x' \in \Omega(x)\}$ and $\underline{x}_j = \min\{x'_j : x' \in \Omega(x)\}$. The weights $\overline{W}^i$ and biases

$\overline{w}^i$ are obtained from LiRPA bounds, which guarantee for each target class $i \neq c$, with c being the groundtruth class:

$$\forall x' \in \Omega(x), \quad f_i(x') - f_c(x') \leq \overline{f}_{i,c}(x') \;=\; \overline{W}^i \cdot x' + \overline{w}^i,$$

Proposition 4.1 (Batch-Certifiable Freeing). If $\Phi(\mathcal{A}; \Omega) \leq 0$, then $\mathcal{F} \setminus \mathcal{A}$ is a weak abductive explanation (WAXp).

Lemma 4.1. If $\Phi(\mathcal{A}) \leq 0$, freeing all features in $\mathcal{A}$ is sound; that is, the property P holds for every $x' \in \Omega(x)$ with $\{x'_k = x_k\}_{k \in \mathcal{F} \setminus \mathcal{A}}$.

The proof of Proposition 4.1 is given in Appendix B. The trivial case $\mathcal{A} = \emptyset$ always satisfies the certificate, but our goal is to efficiently certify large feature sets. The abstract batch certificate also highlights two extreme scenarii. In the first, if $\Phi(\mathcal{F}) \leq 0$, all features are irrelevant, meaning the property P holds across Ω without fixing any inputs. In the second, if $\overline{b}^i(x) \geq 0$ for some $i \neq c$, then $\Phi(\emptyset) > 0$ and no feature can be safely freed; this situation arises when the abstract relaxation is too loose, producing vacuous bounds. Avoiding this degenerate case requires careful selection of the *perturbation domain*, a consideration we highlight for the first time in the context of abductive explanations. The choice of abstract domain is discussed in Section 5.

4.3 Minimizing the Size of an Abstract Explanation via a Knapsack Formulation

Between the trivial and degenerate cases lies the nontrivial setting: finding a maximal *set of irrelevant features* $\mathcal{A}$ to free given the abstract batch certificate Φ. Let $\mathcal{F}$ denote the index set of features. Maximizing $|\mathcal{A}|$ can be naturally formulated as a $0/1$ Multidimensional Knapsack Problem (MKP). For each feature $j \in \mathcal{F}$, we introduce a binary decision variable y_j indicating whether the feature is selected. The optimization problem then reads:

$$\max_{y} \sum_{j \in \mathcal{F}} y_j \text{ s.t. } \sum_{j \in \mathcal{F}} c_{ij} y_j \leq -\overline{b}^i(x), \quad i \in I,\, i \neq c \tag{4}$$

where $c_{i,j}$ represents the contribution of feature j to constraint i, and $-\overline{b}^i(x)$ is the corresponding knapsack capacity. The complexity of this MKP depends on the number of output classes. For binary classification ($k = 2$), the problem is linear[1]. In the standard multiclass setting ($k > 2$), however, the MKP is NP-hard. While moderately sized instances can be solved exactly using a MILP solver, this approach does not scale to large feature spaces. To ensure scalability, we propose a simple and efficient greedy heuristic, formalized in Algorithm 1. Rather than solving the full MKP, the heuristic iteratively selects the feature $j^\star$ that is *least likely* to violate any of the $k-1$ constraints, by minimizing the maximum normalized cost across all classes. An example is provided in appendix C. This procedure is highly parallelizable, since all costs can be computed simultaneously. While suboptimal by design, it produces a set $\mathcal{A}$ such that $\Phi(\mathcal{A}; \Omega) \leq 0$. In Section 7, we compare the performance of this greedy approach against the optimal MILP solution, demonstrating that it achieves competitive results with dramatically improved scalability.

Algorithm 1 Greedy Abstract Batch Freeing (One Step)

Input: model f, perturbation domain Ω^m, candidate set F
Initialize: $\mathcal{A} \leftarrow \emptyset$, linear bounds $\{\overline{W}^i, \overline{w}^i\} = \text{LiRPA}(f, \Omega^m(x))$
Do: compute $c_{i,j}$ in parallel
while $\Phi(\mathcal{A}) \leq 0$ and $|\mathcal{F}| > 0$ **do**
  pick $j^\star = \arg\min_{j \in F \setminus \mathcal{A}} \max_{i \neq c} c_{i,j}/(-\overline{b}_i)$ ▷ parallel reduction
  if $\Phi(\mathcal{A} \cup \{j^\star\}) \leq 0$ and $|\mathcal{A}| \leq m$ **then**
    $\mathcal{A} \leftarrow \mathcal{A} \cup \{j^\star\}$
  end if
  $F \leftarrow F \setminus \{j^\star\}$ ▷ Remove candidate
end while
Return: $\mathcal{A}$

[1] it can be solved optimally in $\mathcal{O}(n)$ time by sorting features by ascending contribution $c_{1,j}$ and greedily adding them until the capacity is exhausted.

5 Refining the Perturbation Domain for Abductive Explanation

Previous approaches for batch freeing reduce the perturbation domain using a traversal order π, defining $\Omega_{\pi,i}(\mathbf{x}) = \{\mathbf{x}' \in \mathbb{R}^n : \|\mathbf{x} - \mathbf{x}'\|_\infty \leq \epsilon,\ \mathbf{x}'_{\pi_{i:}} = \mathbf{x}_{\pi_{i:}}\}$. These methods only consider freeing dimensions up to a certain order. However, as discussed previously, determining an effective order requires prior knowledge of feature importance—the very information that explanations aim to uncover—introducing a circular dependency. This reliance stems from the combinatorial explosion: the number of possible subsets of input features grows exponentially, making naive enumeration of abstract domains intractable.

To address this, we introduce a new perturbation domain, denoted the *cardinality-constrained perturbation domain*. For instance, one can restrict to ℓ_0-bounded perturbations:

$$\Omega^m(\mathbf{x}) = \{\mathbf{x}' \in \mathbb{R}^n : \|\mathbf{x} - \mathbf{x}'\|_\infty \leq \epsilon,\ \|\mathbf{x} - \mathbf{x}'\|_0 \leq m\},$$

which ensures that at most m features may vary simultaneously. This concept is closely related to the ℓ_0 norm and has been studied in verification (42), but, to the best of our knowledge, it is applied here for the first time in the context of abductive explanations. The greedy procedure in Algorithm 1 can then certify a batch of irrelevant features $\mathcal{A}$ under this domain. Once a set $\mathcal{A}$ is freed, the feasible perturbation domain becomes strictly smaller, enabling tighter bounds and the identification of additional irrelevant features. We formalize this as the *refined abstract domain* that ensures that at most m features can vary in addition to the set of previously seclected ones $\mathcal{A}$:

$$\Omega^m(\mathbf{x}; \mathcal{A}) = \{\mathbf{x}' \in \mathbb{R}^n : \|\mathbf{x} - x'\|_\infty \leq \epsilon,\ \|\mathbf{x}_{\mathcal{F}\setminus\mathcal{A}} - \mathbf{x}'_{\mathcal{F}\setminus\mathcal{A}}\|_0 \leq m\}. \tag{5}$$

By construction, $\Omega^m(\mathbf{x}; \mathcal{A}) \subseteq \Omega^{m+|\mathcal{A}|}(\mathbf{x})$, so any free set derived from $\Omega^m(x; \mathcal{A})$ remains sound for the original budget $m + |\mathcal{A}|$. Recomputing linear bounds on this tighter domain often yields strictly smaller abstract explanation. This refinement naturally suggests a recursive strategy: after one round of greedy batch freeing, we restrict the domain to $\Omega^m(\mathbf{x}; \mathcal{A})$, recompute LiRPA bounds, and reapply Algorithm 1 for $m = 1 \ldots |\mathcal{F} \setminus \mathcal{A}|$. As detailed in Algorithm 5, this process can be iterated, progressively shrinking the domain and expanding $\mathcal{A}$. In practice, recursion terminates once no new features can be freed. Finally, any remaining candidate features can be tested individually using the binary search approach proposed by VeriX+ but replacing Marabou by CROWN (see Algorithm 4). This final step ensures that we obtain a formal abstract minimal explanation, as defined in Definition 4.1

6 Distance from Abstract Explanation to Minimality

Algorithm 5 returns a *minimal abstract explanation*: with respect to the chosen LiRPA relaxation, the certified free set $\mathcal{A}$ cannot be further enlarged. This guarantee is strictly weaker than minimality in the exact sense. The remaining features may still include irrelevant coordinates that abstract interpretation fails to certify, due to the coarseness of the relaxation. In other words, minimality is relative to the verifier: stronger but more expensive verifiers (e.g., Verix+ with Marabou) are still required to converge to a *true minimal explanation*.

The gap arises from the tradeoff between verifier accuracy and domain size. Abstract methods become more conservative as the perturbation domain grows, while exact methods remain sound but scale poorly. This motivates hybrid strategies that combine fast but incomplete relaxations with targeted calls to exact solvers. As an additional acceleration step, adversarial attacks can be used. By Lemma B.1, if attacks identify features that must belong to the explanation, they can be added simultaneously (*see Algorithm 3*). Unlike abstract interpretation, the effectiveness of adversarial search typically increases with the domain size: larger regions make it easier to find counterexamples.

Towards minimal explanations. Our strategy is to use the *minimal abstract explanation* ((WAXp$^{A^\star}$)) as a starting point, and then search for the closest minimal explanation. Concretely, we aim to identify the largest candidate set of potentially irrelevant features that, if freed together, would allow all remaining features to be safely added to the explanation at once. A good traversal order of the candidate space is crucial here, as it determines how efficiently such irrelevant features can be pinpointed. Formally, if $\mathcal{X}^A$ denotes the minimal abstract explanation and $\mathcal{X}^{A^\star}$ the closest minimal explanation, we define the *absolute distance to minimality* as the number of irrelevant features not captured by the abstract method: $d(\mathcal{X}^A, \mathcal{X}^{A^\star}) = \left|\mathcal{X}^A \setminus \mathcal{X}^{A^\star}\right|$.

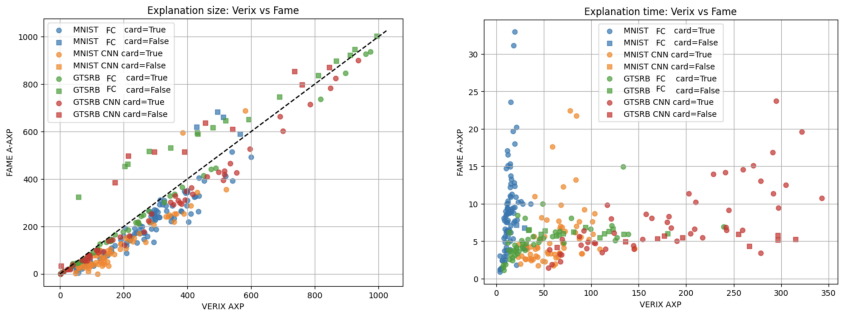

Figure 4: **FAME's iterative refinement approach against the VERIX+ baseline**. The left plot compares the size of the final explanations. The right plot compares the runtime (in $seconds$). The data points for each model are distinguished by color, and the use of circles (card=True) and squares (card=False) indicates whether a cardinality constraint ($||x - x'||_0 \leq m$) was applied.

7 EXPERIMENTS

To evaluate the benefits and reliability of our proposed explainability method, FAME, we performed a series of experiments comparing its performance against the SoTA VERIX+ implementation. We assessed the quality of the explanations generated by FAME by comparing them to those of VERIX+ across four distinct models, including both fully connected and convolutional neural networks (CNNs). We considered two primary performance metrics: the runtime required to compute a single explanation and the size (cardinality) of the resulting explanation.

Our experiments, as in VERIX+ (40), were conducted on two widely-used image classification datasets: MNIST(43) and GTSRB(38). Each score was averaged over non-robust samples from the 100 samples of each dataset. For the comparison results, the explanations were generated using the FAME framework only, and with a final run of VERIX+ to ensure minimality (See Figure 1).

Traversal order	VERIX+ (alone)		FAME: Single-round				FAME: Iterative refinement				FAME-accelerated VERIX+		
	bounds		/				/				/ + bounds		
Search procedure	binary		MILP		Greedy		MILP		Greedy		Greedy + binary		
Metrics ↓	$\|AXp\|$	time	$\|wAXP^A\|$	time	$\|wAXP^A\|$	time	$\|wAXP^A\|$	time	$\|wAXP^A\|$	time	$\|$candidate-set$\|$	$\|AXp\|$	time
MNIST-FC	280.16	13.87	441.05	4.4	448.37	0.35	229.73	14.30	225.14	8.78	44.21	**224.41**	**13.72**
MNIST-CNN	159.78	56.72	181.24	5.59	190.29	0.51	124.9	12.35	122.09	5.6	104.09	**113.53**	**33.75**
GTSRB-FC	**313.42**	56.18	236.85	9.68	243.18	0.97	331.84	12.28	332.74	5.26	11.93	332.66	**9.26**
GTSRB-CNN	338.28	185.03	372.66	12.45	379.34	1.35	321.92	17.74	321.98	7.42	219.57	**322.42**	**138.12**

Table 1: Average explanation size and generation time (in seconds) are compared for FAME (single-round and iterative MILP/Greedy) with FAME-accelerated VERIX+ to achieve minimality.

Experimental Setup All experiments were carried out on a machine equipped with an Apple M2 Pro processor and 16 GB of memory. The analysis is conducted on fully connected (-FC) and convolutional (-CNN) models from the MNIST and GTSRB datasets, with ϵ set to 0.05 and 0.01 respectively. The verified perturbation analysis was performed using the DECOMON library[2], applying the LiRPA CROWN method with an l_∞-norm. The NN verifier Marabou (26) is used within VERIX+. The complete set of hyperparameters and the detailed architectures of the models used for both the MNIST and GTSRB experiments are provided in Appendix D for full reproducibility.

7.1 GREEDY VS. MILP FOR ABSTRACT BATCH FREEING

Performance in a Single Round This experiment, in the 'FAME: Single Round' column of Table 1, compares the runtime and size of the largest free set obtained in a single round using the greedy method versus an exact MILP solver for the abstract batch freeing (Algorithm 1).

Across all models, the greedy heuristic consistently provided a significant **speedup (ranging from** $9\times$ **to** $12\times$**)** while achieving an abstract explanation size very close (fewer than 9 features in average)

[2]https://github.com/airbus/decomon

to that of the optimal MILP solver. This demonstrates that, for single-round batch freeing, the greedy method offers a more practical and scalable solution.

Performance with Iterative Refinement This experiment compares the two methods in an iterative setting of the abstract batch freeing, where the perturbation domain is progressively refined (Section 5). For the iterative refinement process, the greedy approach maintained a substantial runtime advantage over the MILP solver, with a speedup up to $2.4\times$ on the GTSRB-CNN model, while producing abstract explanations that were consistently close in size to the optimal solution. The distinction between the circle and square markers is significant in Figure 4. The square markers (card=False) tend to lie closer to or even above the diagonal line. This suggests that the cardinality-constrained domain, when successful, is highly effective at finding more compact explanations.

7.2 Comparison with State-of-the-Art (VERIX+)

We compare in this section the results of VERIX+ (alone) vs. FAME-accelerated VERIX+.

Explanation Size and Runtime: FAME consistently produces smaller explanations than VERIX+ while being significantly faster, mainly due to FAME's iterative refinement approach, as visually confirmed by the plots in Figure 4 that show a majority of data points falling below the diagonal line for both size and time comparisons. The runtime gains are particularly substantial for the GTSRB models (green and red markers), where FAME's runtime is often only a small fraction of VERIX+'s as shown in Table 1. In some cases, FAME delivers a non-minimal set that is smaller than VERIX+ 's minimal set, with up to a $25\times$ speedup (321.98 features in 7.4s compared to 338.28 in 185.03s for the GTSRB-CNN model) while producing WAXp^A that were consistently close in size to the optimal solution.

The Role of Abstract Freeing: The effectiveness of FAME's approach is further supported by the "distance to minimality" metric. The average distance to minimality was 44.21 for MNIST-FC and 104.09 for MNIST-CNN. An important observation from our experiments is that when the abstract domains in FAME are effective, they yield abstract abductive explanations WAXp^A that are smaller than the abductive explanations (AXp) from VERIX+. This is not immediately obvious from the summary table, as the final explanations may differ. Conversely, when FAME's abstract domains fail to find a valid free set, our method defaults to a binary search approach similar to VERIX+. However, since we do not use the Marabou solver in this phase, the resulting WAXp^A is larger than the AXp provided by Marabou. This highlights the trade-off and the hybrid nature of our approach.

8 Conclusion and Discussion

In this work, we introduced **FAME** (Formal Abstract Minimal Explanations), a novel framework for computing abductive explanations that effectively scales to large neural networks. By leveraging a hybrid strategy grounded in abstract interpretation and dedicated perturbation domains, we successfully addressed the long-standing sequential bottleneck of traditional formal explanation methods.

Our main contribution is a new approach that eliminates the need for traversal order by progressively shrinking dedicated perturbation domains and using LiRPA-based bounds to efficiently discard irrelevant features. The core of our method relies on a greedy heuristic for batch freeing that, as our analysis shows, is significantly faster than an exact MILP solver while yielding comparable explanation sizes.

Our experimental results demonstrate that the full hybrid FAME pipeline outperforms the current state-of-the-art VERIX+ baseline, providing a superior trade-off between computation time and explanation quality. We consistently observed significant reductions in runtime while producing explanations that are close to true minimality. This success highlights the feasibility of computing formal explanations for larger models and validates the effectiveness of our hybrid strategy.

Beyond its performance benefits, the FAME framework is highly generalizable. Although our evaluation focused on classification tasks, the framework can be extended to other machine learning applications, such as regression, and can support a variety of properties beyond local robustness, including local stability. Additionally, FAME can be configured to use exact solvers for the final refinement step, ensuring its adaptability and robustness for various use cases.

References

[1] Audemard, G., Bellart, S., Bounia, L., Koriche, F., Lagniez, J.M., Marquis, P.: Trading complexity for sparsity in random forest explanations. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 5461–5469 (2022)

[2] Bassan, S., Elboher, Y.Y., Ladner, T., Althoff, M., Katz, G.: Explaining, fast and slow: Abstraction and refinement of provable explanations. arXiv preprint arXiv:2506.08505 (2025)

[3] Bassan, S., Eliav, R., Gur, S.: Explain yourself, briefly! self-explaining neural networks with concise sufficient reasons. arXiv preprint arXiv:2502.03391 (2025)

[4] Bassan, S., Katz, G.: Towards formal xai: formally approximate minimal explanations of neural networks. In: International Conference on Tools and Algorithms for the Construction and Analysis of Systems. pp. 187–207. Springer (2023)

[5] Boumazouza, R., Cheikh-Alili, F., Mazure, B., Tabia, K.: Asteryx: A model-agnostic sat-based approach for symbolic and score-based explanations. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 120–129 (2021)

[6] Brix, C., Bak, S., Liu, C., Johnson, T.T.: The fourth international verification of neural networks competition (vnn-comp 2023): Summary and results. arXiv preprint arXiv:2312.16760 (2023)

[7] Chinneck, J.W., Dravnieks, E.W.: Locating minimal infeasible constraint sets in linear programs. ORSA Journal on Computing **3**(2), 157–168 (1991)

[8] Darwiche, A., Ji, C.: On the computation of necessary and sufficient explanations. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 5582–5591 (2022)

[9] De Palma, A., Durand, S., Chihani, Z., Terrier, F., Urban, C.: On using certified training towards empirical robustness. arXiv preprint arXiv:2410.01617 (2024)

[10] Ducoffe, M., Povéda, G., Galametz, A., Boumazouza, R., Martin, M.C., Baris, J., Daverschot, D., O'Higgins, E.: Surrogate neural networks local stability for aircraft predictive maintenance. In: International Conference on Formal Methods for Industrial Critical Systems. pp. 245–258. Springer (2024)

[11] Fel, T., Ducoffe, M., Vigouroux, D., Cadène, R., Capelle, M., Nicodème, C., Serre, T.: Don't lie to me! robust and efficient explainability with verified perturbation analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16153–16163 (June 2023)

[12] Fel, T., Hervier, L., Vigouroux, D., Poche, A., Plakoo, J., Cadene, R., Chalvidal, M., Colin, J., Boissin, T., Bethune, L., et al.: Xplique: A deep learning explainability toolbox. arXiv preprint arXiv:2206.04394 (2022)

[13] Huang, X., Izza, Y., Ignatiev, A., Cooper, M., Asher, N., Marques-Silva, J.: Tractable explanations for d-dnnf classifiers. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 5719–5728 (2022)

[14] Huang, X., Izza, Y., Ignatiev, A., Marques-Silva, J.: On efficiently explaining graph-based classifiers. arXiv preprint arXiv:2106.01350 (2021)

[15] Huang, X., Marques-Silva, J.: From robustness to explainability and back again. arXiv preprint arXiv:2306.03048 (2023)

[16] Ignatiev, A.: Towards trustable explainable ai. In: International Joint Conference on Artificial Intelligence-Pacific Rim International Conference on Artificial Intelligence 2020. pp. 5154–5158. Association for the Advancement of Artificial Intelligence (AAAI) (2020)

[17] Ignatiev, A., Izza, Y., Stuckey, P.J., Marques-Silva, J.: Using maxsat for efficient explanations of tree ensembles. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 3776–3785 (2022)

[18] Ignatiev, A., Marques-Silva, J.: Sat-based rigorous explanations for decision lists. In: International Conference on Theory and Applications of Satisfiability Testing. pp. 251–269. Springer (2021)

[19] Ignatiev, A., Morgado, A., Marques-Silva, J.: Propositional abduction with implicit hitting sets. In: ECAI 2016, pp. 1327–1335. IOS Press (2016)

[20] Ignatiev, A., Narodytska, N., Marques-Silva, J.: Abduction-based explanations for machine learning models. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 1511–1519 (2019)

[21] Izza, Y., Huang, X., Morgado, A., Planes, J., Ignatiev, A., Marques-Silva, J.: Distance-restricted explanations: theoretical underpinnings & efficient implementation. arXiv preprint arXiv:2405.08297 (2024)

[22] Izza, Y., Ignatiev, A., Marques-Silva, J.: On explaining decision trees. arXiv preprint arXiv:2010.11034 (2020)

[23] Izza, Y., Marques-Silva, J.: On explaining random forests with sat. arXiv preprint arXiv:2105.10278 (2021)

[24] Junker, U.: Quickxplain: Preferred explanations and relaxations for over-constrained problems. In: Proceedings of the 19th national conference on Artifical intelligence. pp. 167–172 (2004)

[25] Katz, G., Barrett, C., Dill, D.L., Julian, K., Kochenderfer, M.J.: Reluplex: An efficient smt solver for verifying deep neural networks. In: International conference on computer aided verification. pp. 97–117. Springer (2017)

[26] Katz, G., Huang, D.A., Ibeling, D., Julian, K., Lazarus, C., Lim, R., Shah, P., Thakoor, S., Wu, H., Zeljić, A., et al.: The marabou framework for verification and analysis of deep neural networks. In: International conference on computer aided verification. pp. 443–452. Springer (2019)

[27] La Malfa, E., Zbrzezny, A., Michelmore, R., Paoletti, N., Kwiatkowska, M.: On guaranteed optimal robust explanations for nlp models. arXiv preprint arXiv:2105.03640 (2021)

[28] Lomuscio, A.: Venus: Formal verification for neural systems. Tech. rep. (2023)

[29] Marques-Silva, J.: Minimal unsatisfiability: Models, algorithms and applications (invited paper). In: 40th IEEE International Symposium on Multiple-Valued Logic, ISMVL 2010, Barcelona, Spain, 26-28 May 2010. pp. 9–14. IEEE Computer Society (2010). https://doi.org/10.1109/ISMVL.2010.11, https://doi.org/10.1109/ISMVL.2010.11

[30] Marques-Silva, J.: Disproving xai myths with formal methods–initial results. In: 2023 27th International Conference on Engineering of Complex Computer Systems (ICECCS). pp. 12–21. IEEE (2023)

[31] Marques-Silva, J.: Logic-based explainability in machine learning. In: Reasoning Web. Causality, Explanations and Declarative Knowledge: 18th International Summer School 2022, Berlin, Germany, September 27–30, 2022, Tutorial Lectures, pp. 24–104. Springer (2023)

[32] Marques-Silva, J., Gerspacher, T., Cooper, M., Ignatiev, A., Narodytska, N.: Explaining naive bayes and other linear classifiers with polynomial time and delay. Advances in Neural Information Processing Systems **33**, 20590–20600 (2020)

[33] Marques-Silva, J., Gerspacher, T., Cooper, M.C., Ignatiev, A., Narodytska, N.: Explanations for monotonic classifiers. In: International Conference on Machine Learning. pp. 7469–7479. PMLR (2021)

[34] Serrurier, M., Mamalet, F., González-Sanz, A., Boissin, T., Loubes, J.M., Del Barrio, E.: Achieving robustness in classification using optimal transport with hinge regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2021)

[35] Shih, A., Choi, A., Darwiche, A.: A symbolic approach to explaining bayesian network classifiers. arXiv preprint arXiv:1805.03364 (2018)

[36] Singh, G., Gehr, T., Püschel, M., Vechev, M.: An abstract domain for certifying neural networks. Proceedings of the ACM on Programming Languages **3**(POPL), 1–30 (2019)

[37] de Siqueira, N.: Jl, and puget, j.-f. 1988. explanationbased generalisation of failures. In: Proceedings of the Eighth European Conference on Artificial Intelligence (ECAI'88). pp. 339–344 (1988)

[38] Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C.: Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. Neural networks **32**, 323–332 (2012)

[39] Urban, C., Miné, A.: A review of formal methods applied to machine learning. arXiv preprint arXiv:2104.02466 (2021)

[40] Wu, M., Li, X., Wu, H., Barrett, C.: Better verified explanations with applications to incorrectness and out-of-distribution detection. arXiv preprint arXiv:2409.03060 (2024)

[41] Wu, M., Wu, H., Barrett, C.: Verix: towards verified explainability of deep neural networks. Advances in neural information processing systems **36**, 22247–22268 (2023)

[42] Xu, K., Shi, Z., Zhang, H., Wang, Y., Chang, K.W., Huang, M., Kailkhura, B., Lin, X., Hsieh, C.J.: Automatic perturbation analysis for scalable certified robustness and beyond. Advances in Neural Information Processing Systems **33**, 1129–1141 (2020)

[43] Yann, L.: Mnist handwritten digit database. ATT Labs. (2010)

[44] Zhang, H., Weng, T.W., Chen, P.Y., Hsieh, C.J., Daniel, L.: Efficient neural network robustness certification with general activation functions. Advances in neural information processing systems **31** (2018)

[45] Zhao, Z., Zhang, Y., Chen, G., Song, F., Chen, T., Liu, J.: Cleverest: accelerating cegar-based neural network verification via adversarial attacks. In: International Static Analysis Symposium. pp. 449–473. Springer (2022)