Safety and Trust in Artificial Intelligence with Abstract Interpretation

Other titles in Foundations and Trends® in Programming Languages

Neurosymbolic Programming in Scallop: Principles and Practice
Ziyang Li, Jiani Huang, Jason Liu and Mayur Naik
ISBN: 978-1-63828-484-0

From Fine- to Coarse-Grained Dynamic Information Flow Control and Back
Marco Vassena, Alejandro Russo, Deepak Garg, Vineet Rajani and Deian Stefan
ISBN: 978-1-63828-218-1

Probabilistic Trace and Testing Semantics: The Importance of Being Coherent
Marco Bernardo
ISBN: 978-1-63828-074-3

Neurosymbolic Programming
Swarat Chaudhuri, Kevin Ellis, Oleksandr Polozov, Rishabh Singh, Armando Solar-Lezama and Yisong Yue
ISBN: 978-1-68083-934-0

Introduction to Neural Network Verification
Aws Albarghouthi
ISBN: 978-1-68083-910-4

Refinement Types: A Tutorial
Ranjit Jhala and Niki Vazou
ISBN: 978-1-68083-884-8

Safety and Trust in Artificial Intelligence with Abstract Interpretation

Gagandeep Singh
UIUC
ggnds@illinois.edu

Jacob Laurel
Georgia Institute of Technology
jlaurel6@gatech.edu

Sasa Misailovic
UIUC
misailo@illinois.edu

Debangshu Banerjee
UIUC
db21@illinois.edu

Avaljot Singh
UIUC
avaljot2@illinois.edu

Changming Xu
UIUC
cx23@illinois.edu

Shubham Ugare
UIUC
sugare2@illinois.edu

Huan Zhang
UIUC
huanz@illinois.edu

now
the essence of knowledge
Boston — Delft

Foundations and Trends® in Programming Languages

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

G. Singh *et al.*. *Safety and Trust in Artificial Intelligence with Abstract Interpretation.* Foundations and Trends® in Programming Languages, vol. 8, no. 3-4, pp. 250–408, 2025.

ISBN: 978-1-63828-587-8

Foundations and Trends® in Programming Languages
Volume 8, Issue 3-4, 2025

Editorial Board

Editorial Scope

Foundations and Trends® in Programming Languages publishes survey and tutorial articles in the following topics:

- Abstract Interpretation
- Compilation and Interpretation Techniques
- Domain Specific Languages
- Formal Semantics, including Lambda Calculi, Process Calculi, and Process Algebra
- Language Paradigms
- Mechanical Proof Checking
- Memory Management
- Partial Evaluation
- Program Logic
- Programming Language Implementation
- Programming Language Security
- Programming Languages for Concurrency
- Programming Languages for Parallelism
- Program Synthesis
- Program Transformations and Optimizations
- Program Verification
- Runtime Techniques for Programming Languages
- Software Model Checking
- Static and Dynamic Program Analysis
- Type Theory and Type Systems

Information for Librarians

Foundations and Trends® in Programming Languages, 2025, Volume 8, 4 issues. ISSN paper version 2325-1107. ISSN online version 2325-1131. Also available as a combined paper and online subscription.

Contents

Safety and Trust in Artificial Intelligence with Abstract Interpretation

Gagandeep Singh[1], Jacob Laurel[2], Sasa Misailovic[1], Debangshu Banerjee[1], Avaljot Singh[1], Changming Xu[1], Shubham Ugare[1] and Huan Zhang[1]

[1] *University of Illinois Urbana-Champaign, USA; {ggnds, misailo, db21, avaljot2, cx23, sugare2, huanz}@illinois.edu*
[2] *Georgia Institute of Technology, USA; jlaurel6@gatech.edu*

ABSTRACT

Deep neural networks (DNNs) now dominate the AI landscape and have shown impressive performance in diverse application domains, including vision, natural language processing (NLP), and healthcare. However, both public and private entities have been increasingly expressing significant concern about the potential of state-of-the-art AI models to cause societal and financial harm. This lack of trust arises from their black-box construction and vulnerability against natural and adversarial noise.

As a result, researchers have spent considerable time developing automated methods for building safe and trustworthy DNNs. Abstract interpretation has emerged as the most popular framework for efficiently analyzing realistic DNNs among the various approaches. However, due to fundamental differences in the computational structure (e.g., high

Gagandeep Singh, Jacob Laurel, Sasa Misailovic, Debangshu Banerjee, Avaljot Singh, Changming Xu, Shubham Ugare and Huan Zhang (2025), "Safety and Trust in Artificial Intelligence with Abstract Interpretation", Foundations and Trends® in Programming Languages: Vol. 8, No. 3-4, pp 250–408. DOI: 10.1561/2500000062.

nonlinearity) of DNNs compared to traditional programs, developing efficient DNN analyzers has required tackling significantly different research challenges than encountered for programs.

In this monograph, we describe state-of-the-art approaches based on abstract interpretation for analyzing DNNs. These approaches include the design of new abstract domains, synthesis of novel abstract transformers, abstraction refinement, and incremental analysis. We will discuss how the analysis results can be used to: (i) formally check whether a trained DNN satisfies desired output and gradient-based safety properties, (ii) guide the model updates during training towards satisfying safety properties, and (iii) reliably explain and interpret the black-box workings of DNNs.

1

Introduction

Deep neural networks (DNNs) are currently the dominant technology in artificial intelligence (AI) and have shown impressive performance in diverse applications, including autonomous driving, medical diagnosis, text generation, and logical reasoning. However, they lack transparency due to their black-box construction and are vulnerable to environmental and adversarial noise. These issues have caused concerns about their safety and trust when deployed in the real world. Although standard training optimizes the model's accuracy, it does not take into account desirable safety properties such as *robustness* (the DNN should behave similarly for similar inputs), *fairness* (the DNN output should not depend too much on some legally protected attribute, such as gender or race), and *monotonicity* (if the inputs are partially ordered, so should be the outputs). As a result, state-of-the-art models remain untrustworthy. Building trust in DNNs is essential to realizing their vast potential to positively transform society and the economy and is one of the grand challenges in computer science today.

1.1 Safety-informed DNN Deployment Cycle

Figure 1.1 presents a general safety-informed pipeline for DNN development, applicable to any application domain. Safety, accuracy, and efficiency can often conflict with each other. DNN accuracy improves with model size but that increases the inference cost. Similarly, models maximizing safety can have reduced accuracy. For example, a DNN classifier that always predicts the same class for all inputs is robust but has very low accuracy. As a result, it may not be possible to obtain DNNs that optimize all three objectives simultaneously. Depending on the target application, a developer may prioritize accuracy over trust or vice-versa. The goal of safety-informed DNN development is to ensure a sufficient balance between accuracy, safety, and efficiency.

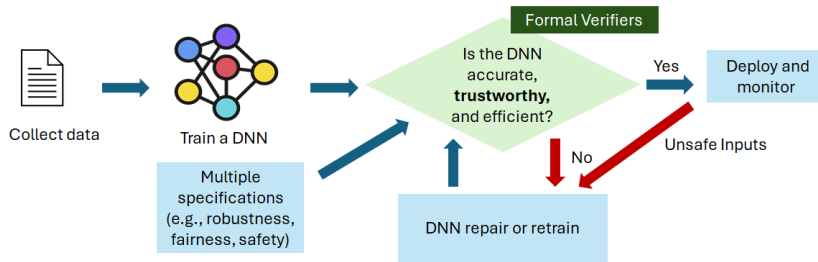


Figure 1.1: Development pipeline for building accurate, trustworthy, and efficient DNNs. Verification is used for testing model trustworthiness (green diamond).

In this pipeline, first, representative training data for the target application is collected and a DNN is trained to maximize its accuracy on test inputs from the training distribution. Next, a domain expert creates (manually or algorithmically) a set of formal safety specifications (e.g., robustness, fairness) characterizing the expected DNN behavior in different real-world scenarios. The set of inputs covered by these specifications can be infinite.

The expert then checks whether the model meets the safety standards. Since DNNs may not satisfy all the specifications, the standards can require that at least a significant fraction of all specifications be satisfied for trustworthiness. If the model meets the criteria, then the DNN is considered fit for deployment. Otherwise, it is iteratively re-

paired (e.g., by fine-tuning) until we obtain the desired balance between accuracy, safety, and efficiency.

During deployment, the DNN inputs are monitored for distribution shifts, i.e., the inputs are not from the training distribution. If the runtime system detects a distribution shift, it reports representative samples to the domain experts. They then design new specifications, and the model undergoes another round of repair (or full retraining).

How formal methods can help. For checking that the model satisfies safety specifications, the standard practice is to evaluate the DNN behavior on a finite set of inputs satisfying the specifications. However, this cannot guarantee safe and trustworthy DNN behavior on all specification inputs. The unseen set can be huge and contain inputs often seen during real-world deployment. To address these limitations, there is growing work on checking the safety of DNN models and interpreting their behavior, on an infinite set of unseen inputs from safety specifications using formal methods, which provides a more reliable metric for measuring a model's safety than standard empirical methods. For example, a repaired DNN preserving the original test set accuracy and efficiency but satisfying the trustworthy specifications more often is a better model than the unrepaired one as it is less likely to show undesirable behavior during real-world deployment. Formal methods can also be used during training to guide the model to satisfy desirable safety and trustworthiness properties. The models trained this way are more likely to satisfy safety specifications than those without.

This monograph presents a comprehensive treatment of the techniques that guarantee the safety of DNNs by formally modeling the behavior of modern DNNs and efficiently computing with abstractions that represent those behaviors. Our main focus will be on approaches that leverage a general framework for automated analysis of programming languages called abstract interpretation, the most successful formal methods for automatically reasoning about DNNs. We emphasize that the knowledge of the topics covered in this monograph is necessary not only for computer scientists but for practitioners from all areas building DNN-based applications, e.g., natural sciences, aerospace, finance, etc.

Next, we describe how safety and trustworthy properties can be formally specified for DNNs, then we will discuss the key ideas and design

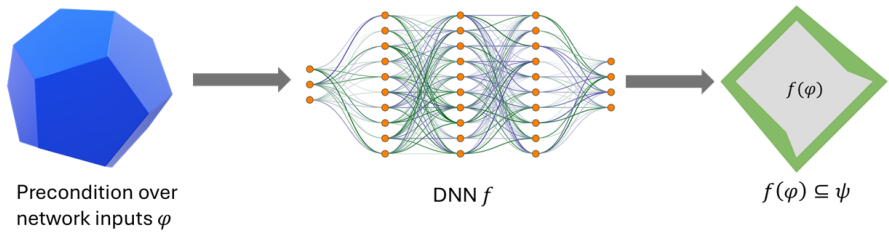
considerations in developing abstract interpretation based methods for the formal verification and training of DNNs. We will also discuss how abstract interpretation enables reliable explanations and interpretations of DNNs as well as analysis of differentiable programs.

1.2 Formal Specifications for DNNs

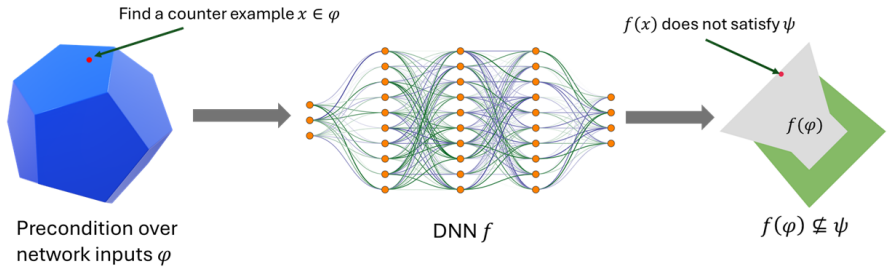
Mathematically, we model a trained DNN as a pure function f. Its input x can be images, text, videos, sensor measurements, or other data. We denote the output of the DNN as $f(x)$, which can be a classification of the input into one of the predefined classes, the regression that estimates a continuous value, or the set of tokens generated by a language model. We denote gradients of f as $f'(x)$.

For a trained DNN f, a developer specifies the property of interest using two formulas: (1) *the precondition* φ, which specifies the set of inputs on which the DNN should not misbehave and (2) *the postcondition* ψ, which specifies safe and trustworthy behaviors of the DNN for the given inputs. These behaviors are typically constraints on the DNN's outputs or its gradients. The preconditions and postconditions are domain-dependent and usually designed by DNN developers.A tool for DNN verification (*a verifier*) aims to automatically check if the postcondition on the DNN's outputs and/or gradients is satisfied for all inputs specified by the precondition.

A property specification is a tuple (φ, ψ), where φ is the precondition and ψ is the postcondition. Both formulas φ and ψ typically represent *an infinite number of inputs/outputs.* We denote the set of the results of the evaluations of the DNN on all inputs described by the precondition φ as $f(\varphi) = \{f(x) \mid x \in \varphi\}$. Similarly, we denote the set of all gradients as $f'(\varphi)$. The verifier then checks for the inclusion of the set of possible executions of the DNN into the set of outputs that satisfy the postcondition, i.e., $f(\phi) \subseteq \psi$ (or $f'(\phi) \subseteq \psi$) holds. *Single execution* specifications, as shown in Figure 1.2, require that each DNN output $f(x)$ where $x \in \varphi$ must independently satisfy ψ. *Relational* specifications require reasoning about multiple related executions of the same or different DNNs. As we will show in Section 3, a general way to represent and compute with φ and ψ in these settings is as disjunctions



(a) Verified



(b) Counterexample

Figure 1.2: Single execution specifications require that the DNN output for each input from φ must independently satisfy ψ.

of convex polyhedra within the framework of abstract interpretation. φ and ψ can also define distributions leading to *probabilistic* specifications.

Local and global properties. The set of specifications for DNNs can be broadly classified as *local* or *global*. The precondition φ for local properties defines a local neighborhood around a sample input from the test set. For example, given a test image correctly classified as a car by a DNN, the commonly used *local robustness property* specifies that if the original image was classified as a car, then all images generated by rotating the original image within $\pm d$ degrees are also classified as a car. We present many local properties in Sections 3.1 and 3.2.

In contrast, global properties are not defined with respect to a specific test input. Verifying global properties yields stronger safety guarantees compared to local properties, however, global properties are difficult to formulate for popular domains, such as vision and NLP, where the individual features processed by the DNN have no clear semantic meaning. While verifying local properties is not ideal, the local

verification results enable testing the safety of the model on an infinite set of unseen inputs, not possible with standard methods. We present several concrete global properties in Section 3.1.

1.3 Verifying Specifications over DNNs

DNN verification can be seen as an instance of program verification, since one can write a DNN as a program, i.e., there is a direct translation from the mathematical representation of the DNN as a function f into a side-effect free program. However, since it is well-known that program verification is *undecidable* (one cannot prove the correctness of an arbitrary program with respect to an arbitrary property of interest), DNN verification is also undecidable in general. Certain DNN verification problems, such as robustness verification of feedforward networks with ReLU activations, are decidable but still NP-complete in general (Katz *et al.*, 2017).

State-of-the-art verifiers are therefore incomplete in general, i.e., they can fail to prove a specification when it holds. However, when they succeed, the DNN will satisfy the specification. In this monograph, we focus on *white-box* verifiers that require access to the model parameters. Verification of closed-source models requires *black-box* verifiers. We refer the interested readers to the relevant material in this direction in Section 1.7. The white-box verifiers can be formulated using the elegant framework of abstract interpretation. The verifier is parameterized by the choice of an *abstract domain* with two main components: abstract elements and abstract transformers. Abstract elements are mathematical objects symbolically representing an infinite set of numerical points over which the verifier operates. Abstract transformers overapproximate the effect of applying the transformations inside the DNN program (e.g., affine or ReLU assignments) on abstract elements.

There is a tradeoff between the cost and overapproximation error (also known as precision) of an incomplete verifier: expensive verifiers are more precise while cheap verifiers are imprecise. Both are determined by the design of the abstract domain and transformers. The key consideration in designing an efficient verifier applicable to real-world DNNs is managing this tradeoff. The classical domains, such as Polyhe-

dra and Octagons, used for analyzing programs are not well suited for DNN verification. This is because the DNNs have a different structure compared to traditional programs. For example, DNNs have a large number of non-linear assignments but typically do not have loops. For efficient verification, researchers have developed numerous new abstract domains and transformers tailored for DNN verification. These abstract domains can scale to realistic DNNs with millions of neurons, or more than 100 layers, verifying diverse safety properties in different real-world applications. We will present them in Section 3. We will also discuss how verification can be done incrementally to improve efficiency when verifying a large number of similar DNNs and specifications as needed for the developement pipeline in Figure 1.1.

1.4 Training Provably Safe DNNs

DNNs trained with standard training often do not satisfy safety specifications as safety satisfaction is not part of their training objective. Adversarial or counter-example guided training augment the training data with violating examples during training, however the trained models still cannot be proven to be safe in most cases. To overcome these limitations, certified training methods have been developed in recent years which directly incorporate the verifier computations within the training loop and generate models with a high degree of provability, i.e., they are more likely to satisfy specifications and are relatively easier to prove than DNNs obtained with competing methods.

In certified training, if the model f does not satisfy the specification, as checked by a verifier, its weights are updated to increase the provability. The gradient updates are derived by formulating a differentiable property loss on the verifier output, which measures how far the model is from satisfying the property. Since gradient updates are derived from the verifier code, its computations must be expressible as a differentiable function of model weights and parallelizable on GPUs for scalability. Overall, certified training can be seen as training f where the model updates are derived by differentiating the surrogate approximation of the DNN within φ, computed by the verifier.

While certified training improves the provability, safety specifications can be in conflict with accuracy. Using an imprecise verifier during training can result in overregularization and a significant reduction in the standard accuracy. However, precise verifiers often have complicated code which makes the optimization problem too complicated to solve during training, yielding suboptimal results. Also, employing a verifier during training is more expensive than when used for checking specifications on an already trained DNN, as now the verifier is called during every training iteration. Balancing the provability, accuracy, and cost is therefore the main challenge when developing state-of-the-art methods. Researchers have developed a variety of abstractions, refinements, and loss formulations to enable efficient training. We will cover these methods in detail in Section 4.

1.5 Explaining and Interpreting DNNs

Popular methods for explaining DNN predictions identify relevant input features that influence the DNN output the most. However, they do not give guarantees about the robustness of the generated explanations. Relying on non-robust explanations can lead to a false sense of confidence in an untrustworthy model. We will discuss how abstract interpretation can be leveraged to generate explanations with robustness guarantees in Section 5, reliably improving DNN transparency.

Abstract interpretation-based DNN verifiers generate high-dimensional abstract elements at different layers capturing complex relationships between neurons and DNN inputs to prove DNN safety. However, the individual neurons and inputs in the DNN do not have any semantic meaning, unlike the variables in programs, therefore it is not clear whether the safety proofs are based on any meaningful features learned by the DNN. If the DNN is proven to be safe but the proof is based on meaningless features not aligned with human intuition, then the DNN behavior cannot be considered trustworthy. While there has been a lot of work on interpreting black-box DNNs, standard methods can only explain the DNN behavior on individual inputs and cannot interpret the complex invariants encoded by the abstract elements capturing DNN behavior on an infinite set of inputs.

The main challenge in interpreting DNN proofs is mapping the complex abstract elements to human-understandable interpretations.

Section 5 presents ProFIt, the first method for interpreting robustness proofs computed by DNN verifiers. The technique can interpret proofs computed by different verifiers. It builds upon the novel concept of *proof features* computed by projecting the high-dimensional abstract elements onto individual neurons. The proof features can be analyzed independently by generating the corresponding interpretations. Since certain proof features can be more important for the proof than others, a priority function over the proof features that signifies the importance of each proof feature in the complete proof is defined. The method extracts a set of proof features by retaining only the more important parts of the proof that preserve the property.

A comparison of proof interpretations for DNNs trained with standard and robust training methods shows that the proof features corresponding to the standard networks rely on spurious input features that are not aligned with human intuition. The proofs of adversarially trained DNNs filter out some of these spurious features. In contrast, the networks trained with certifiable training produce proofs that do not rely on any spurious features but they also miss out on some meaningful features. Proofs for training methods that combine both empirical and certified robustness not only preserve meaningful features but also selectively filter out spurious ones. These insights suggest that DNNs can satisfy safety properties but their behavior can still be untrustworthy.

1.6 Analyzing and Verifying Differentiable Programs

Differentiable programming, which includes automatic differentiation (AD), is the backbone of machine learning. AD computes the gradients alongside the values of the program's output variables. AD computations generalize many machine learning and signal processing applications. Thus, generalized abstractions for AD analysis can be deployed across applications: a neural network, an image filter, and a differential equation solver can be expressed and analyzed in the same language, even when combined in complex programs. Despite AD's ubiquity, automated formal reasoning of derivatives that AD computes has lagged.

Analyzing gradient properties is important for today's trustworthy AI: for instance, the sensitivity of DNN's output to input noise can be expressed as finding bound for the absolute gradients values. The same bounds can help with selecting low precision data types in machine learning algorithms to prevent overflows. Fairness can be formalized as a monotonicity property on a specific attribute, which is satisfied when all derivatives are strictly positive.

To answer these questions, it is not sufficient to reason about the output of a function (e.g., DNN) f for all inputs in φ. Instead one has to reason about f', the derivative of f. For instance, to prove the monotonicity of f, one should ensure that its derivative f' is strictly positive or negative for all inputs in φ. Figure 1.3 presents an intuition of this workflow.

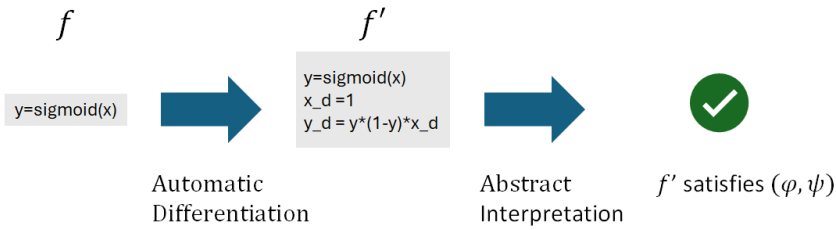


Figure 1.3: Verifying derivative properties requires first computing the derivative of a function f (given as a piece of code) using automatic differentiation. The derivative program is then analyzed with abstract interpretation to prove the desired property (φ, ψ) holds for the derivative f' instead of f itself.

Section 6 will present a general framework for precise analysis of AD computations. This approach leverages ideas from abstract interpretation of DNNs and generalizes them to find precise abstract transformers of gradient computation. It overcomes the limitation of standard program analysis, which treats the gradient computation as any other code, and leads to significant imprecision, and is in some cases ill-defined. We will present the advantage of the AD-specific abstract transformers on the case study for monotonicity analysis for a decision-making DNN.

1.7 Sources and Further Reading

Many recent studies demonstrate the power of modern DNNs, e.g., Bojarski *et al.* (2016) for autonomous driving, Amato *et al.* (2013) for medical diagnosis, Brown *et al.* (2020) for text generation, and Pan *et al.* (2023) for logical reasoning. Many domains have standard datasets for training and inference, e.g., in vision MNIST (LeCun *et al.*, 1989), CIFAR10 (Krizhevsky, 2009), and ImageNet (Deng *et al.*, 2009).

At the same time, recent research also points out key concerns: Ribeiro *et al.* (2016) discusses the issues of black-box model construction and non-interpretability; Szegedy *et al.* (2014) and Kurakin *et al.* (2017) discuss vulnerability against environmental and adversarial noise. Works that pointed out problems with standard training include Shafique *et al.* (2020) for robustness, Dwork *et al.* (2012) for fairness, and Sill (1997) for monotonicity. Tsipras *et al.* (2019) and Wong *et al.* (2021) point out problems with using a finite set of test inputs to ensure DNN safety during deployment. Many recent works, identify classes of slight adversarial perturbations that impact the DNN decisions (Madry *et al.*, 2017; Goodfellow *et al.*, 2014; Heo *et al.*, 2019).

For examples of local robustness to image rotations and its classification see, e.g., Balunovic *et al.* (2019). For examples of global properties in air traffic collision avoidance systems see, e.g., Katz *et al.* (2017), and in security vulnerability classification see, e.g., Chen *et al.* (2021). Beyond manual design, there is a growing line of work on automatically generating formal specifications for DNNs. These include Geng *et al.* (2022), Chaudhary *et al.* (2024b), Geng *et al.* (2024), and Jin *et al.* (2024).

Checking the safety of DNNs has been a very active area of research with many publications, primarily during inference and relying on white-box access to the model, such as Balunovic *et al.* (2019), Singh *et al.* (2019b), Zhang *et al.* (2018a), Singh *et al.* (2018), Singh *et al.* (2019d), Paulsen *et al.* (2020), Xu *et al.* (2021), Tran *et al.* (2019b), Wu *et al.* (2022b), Anderson *et al.* (2019), Katz *et al.* (2019), Singh *et al.* (2019a), Wong and Kolter (2018), Lan *et al.* (2022), Wang *et al.* (2018), Bunel *et al.* (2020), Wang *et al.* (2021), Ugare *et al.* (2022), Kabaha and Drachsler-Cohen (2022), Palma *et al.* (2021a), Dathathri *et al.* (2020),

Munakata *et al.* (2023), Ranzato *et al.* (2021), Banerjee *et al.* (2024b), Banerjee *et al.* (2024a), and Zhou *et al.* (2024). Black-box DNN verifiers are based on collecting DNN output for inputs from φ and providing probabilistic guarantees. These include Baluta *et al.* (2021), Webb *et al.* (2019), Chaudhary *et al.* (2024a), and Chaudhary *et al.* (2025).

Certified training leverages DNN verifiers during training obtaining models that have higher provability than those with standard training. Examples include Gowal *et al.* (2019), Mirman *et al.* (2018), Xu *et al.* (2020), Zhang *et al.* (2020), Shi *et al.* (2021), Yang *et al.* (2023), Müller *et al.* (2023a), Balunovic and Vechev (2020), and Hu *et al.* (2023b).

Numerous methods aim to provide transparency of DNNs. Standard methods include Ribeiro *et al.* (2016) and Wu *et al.* (2023) and Wong *et al.* (2021). Marques-Silva and Ignatiev (2022), Malfa *et al.* (2021), Ignatiev *et al.* (2019), Darwiche and Hirth (2020), and Wu *et al.* (2023) generate explanations with formal guarantees. The work of Banerjee *et al.* (2024a) presents ProFIt, the first method for interpreting robustness proofs computed by DNN verifiers.

Various uses of automatic differentiation are presented (Hückelheim *et al.*, 2018). Static analysis of AD computations is introduced by Laurel *et al.* (2022a), Laurel *et al.* (2022b), and Laurel *et al.* (2023). Verification of properties involving gradients and Jacobians are discussed by Zhang *et al.* (2019), Fazlyab *et al.* (2019b), and Shi *et al.* (2022)

Abstract interpretation was introduced in the seminal work by Cousot and Cousot (1977). Over the past almost 50 years, this approach to program analysis has flourished and demonstrated many uses. There are numerous books, monographs, and tutorials describing the foundations of abstract interpretations, for instance Cousot (2021), Nielson *et al.* (2005), Miné (2017), and Rival and Yi (2020).

Examples of abstract domains for neural networks include DeepPoly/CROWN (Singh *et al.*, 2019b; Zhang *et al.*, 2018a), DeepZ/FastLin (Singh *et al.*, 2018; Weng *et al.*, 2018), Star sets (Tran *et al.*, 2019b), and DeepJ (Laurel *et al.*, 2022a). These custom solutions can scale to realistic DNNs with up to a million neurons (Müller *et al.*, 2021a), or more than 100 layers (Wu *et al.*, 2022b), verifying diverse safety properties in different real-world applications. Examples include autonomous driving (Yang *et al.*, 2023), job-scheduling (Wu *et al.*, 2022b), data

center management (Chakravarthy *et al.*, 2022), biology (Mohr *et al.*, 2021), aerospace (Cohen *et al.*, 2024), and financial modeling (Laurel *et al.*, 2023). For examples of refinements of abstract domains used in machine learning see e.g., Wang *et al.* (2018), Singh *et al.* (2019d), Müller *et al.* (2021b), Ryou *et al.* (2021), Wang *et al.* (2021), Wu *et al.* (2022b), and Yang *et al.* (2021).

References

Allamigeon, X., S. Gaubert, and É. Goubault. (2008). "Inferring Min and Max Invariants Using Max-Plus Polyhedra". In: *Static Analysis*. Ed. by M. Alpuente and G. Vidal. Berlin, Heidelberg: Springer Berlin Heidelberg. 189–204.

Alvarez-Melis, D. and T. S. Jaakkola. (2018). "Towards robust interpretability with self-explaining neural networks". In: *Neural Information Processing Systems*.

Amato, F., A. López, E. M. Peña-Méndez, P. Vaňhara, A. Hampl, and J. Havel. (2013). "Artificial neural networks in medical diagnosis". *Journal of Applied Biomedicine*. 11(2).

Anderson, G., S. Pailoor, I. Dillig, and S. Chaudhuri. (2019). "Optimization and Abstraction: A Synergistic Approach for Analyzing Neural Network Robustness". In: *Proc. Programming Language Design and Implementation (PLDI)*. 731–744.

Anderson, R., J. Huchette, W. Ma, C. Tjandraatmadja, and J. P. Vielma. (2020). "Strong mixed-integer programming formulations for trained neural networks". *Mathematical Programming*. 183(1): 3–39.

Anil, C., J. Lucas, and R. B. Grosse. (2019). "Sorting Out Lipschitz Function Approximation". In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by K. Chaudhuri and R. Salakhutdinov. Vol. 97. *Proceedings of Machine Learning Research*. PMLR. 291–301. URL: http://proceedings.mlr.press/v97/anil19a.html.

Baader, M., M. Mirman, and M. Vechev. (2020). "Universal Approximation with Certified Networks". In: *International Conference on Learning Representations*.

Bak, S. (2021). "nnenum: Verification of ReLU Neural Networks with Optimized Abstraction Refinement". In: *NASA Formal Methods - 13th International Symposium, NFM 2021, Virtual Event, May 24-28, 2021, Proceedings*. Ed. by A. Dutle, M. M. Moscato, L. Titolo, C. A. Muñoz, and I. Perez. Vol. 12673. *Lecture Notes in Computer Science*. Springer. 19–36. DOI: 10.1007/978-3-030-76384-8_2.

Bak, S., T. Dohmen, K. Subramani, A. Trivedi, A. Velasquez, and P. Wojciechowski. (2023). "The Octatope Abstract Domain for Verification of Neural Networks". In: *Formal Methods - 25th International Symposium, FM 2023, Lübeck, Germany, March 6-10, 2023, Proceedings*. Ed. by M. Chechik, J. Katoen, and M. Leucker. Vol. 14000. *Lecture Notes in Computer Science*. Springer. 454–472. DOI: 10.1007/978-3-031-27481-7_26.

Bak, S., T. Dohmen, K. Subramani, A. Trivedi, A. Velasquez, and P. Wojciechowski. (2024). "The hexatope and octatope abstract domains for neural network verification". *Form. Methods Syst. Des.* 64(1): 178–199. DOI: 10.1007/s10703-024-00457-y.

Bak, S., C. Liu, and T. Johnson. (2021). "The second international verification of neural networks competition (vnn-comp 2021): Summary and results". *arXiv preprint arXiv:2109.00498*.

Bak, S., H. Tran, K. Hobbs, and T. T. Johnson. (2020). "Improved Geometric Path Enumeration for Verifying ReLU Neural Networks". In: *Computer Aided Verification - 32nd International Conference, CAV 2020, Los Angeles, CA, USA, July 21-24, 2020, Proceedings, Part I*. Ed. by S. K. Lahiri and C. Wang. Vol. 12224. *Lecture Notes in Computer Science*. Springer. 66–96. DOI: 10.1007/978-3-030-53288-8_4.

Balunovic, M., M. Baader, G. Singh, T. Gehr, and M. Vechev. (2019). "Certifying Geometric Robustness of Neural Networks". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc.

Balunovic, M. and M. T. Vechev. (2020). "Adversarial Training and Provable Defenses: Bridging the Gap". In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Baluta, T., Z. L. Chua, K. S. Meel, and P. Saxena. (2021). "Scalable Quantitative Verification For Deep Neural Networks". In: *43rd IEEE/ACM International Conference on Software Engineering, ICSE 2021, Madrid, Spain, 22-30 May 2021*. IEEE. 312–323. DOI: 10.1109/ICSE43902.2021.00039.

Banerjee, D., A. Singh, and G. Singh. (2024a). "Interpreting Robustness Proofs of Deep Neural Networks". In: *The Twelfth International Conference on Learning Representations*. URL: https://openreview.net/forum?id=Ev10F9TWML.

Banerjee, D. and G. Singh. (2024). "Relational DNN Verification With Cross Executional Bound Refinement". In: *Forty-first International Conference on Machine Learning*. URL: https://openreview.net/forum?id=HOG80Yk4Gw.

Banerjee, D., C. Xu, and G. Singh. (2024b). "Input-Relational Verification of Deep Neural Networks". *Proc. ACM Program. Lang.* 8(PLDI). DOI: 10.1145/3656377.

Barrett, B., A. Camuto, M. Willetts, and T. Rainforth. (2022). "Certifiably robust variational autoencoders". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 3663–3683.

Batten, B., M. Hosseini, and A. Lomuscio. (2024). "Tight Verification of Probabilistic Robustness in Bayesian Neural Networks". In: *International Conference on Artificial Intelligence and Statistics, 2-4 May 2024, Palau de Congressos, Valencia, Spain*. Ed. by S. Dasgupta, S. Mandt, and Y. Li. Vol. 238. *Proceedings of Machine Learning Research*. PMLR. 4906–4914. URL: https://proceedings.mlr.press/v238/batten24a.html.

Becker, B. and R. Kohavi. (1996). "Adult". UCI Machine Learning Repository.

Bender, E. M., T. Gebru, A. McMillan-Major, and S. Shmitchell. (2021). "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" In: *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*. Ed. by M. C. Elish, W. Isaac, and R. S. Zemel. ACM. 610–623.

Bendtsen, C. and O. Stauning. (1996). "FADBAD, a flexible C++ package for automatic differentiation".

Berrada, L., S. Dathathri, K. Dvijotham, R. Stanforth, R. Bunel, J. Uesato, S. Gowal, and M. P. Kumar. (2021). "Make Sure You're Unsure: A Framework for Verifying Probabilistic Specifications". In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Ed. by M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan. 11136–11147. URL: https://proceedings.neurips.cc/paper/2021/hash/5c5bc7df3d37b2a7ea29e1b47b2bd4ab-Abstract.html.

Blalock, D. W., J. J. G. Ortiz, J. Frankle, and J. V. Guttag. (2020). "What is the State of Neural Network Pruning?" In: *MLSys*. mlsys.org.

Blondel, M., Q. Berthet, M. Cuturi, R. Frostig, S. Hoyer, F. Llinares-López, F. Pedregosa, and J.-P. Vert. (2022). "Efficient and modular implicit differentiation". *Advances in neural information processing systems*. 35: 5230–5242.

Bojarski, M., D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, *et al.* (2016). "End to end learning for self-driving cars". *arXiv preprint arXiv:1604.07316*.

Bonaert, G., D. I. Dimitrov, M. Baader, and M. T. Vechev. (2021). "Fast and precise certification of transformers". In: *PLDI '21: 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation, Virtual Event, Canada, June 20-25, 2021*. Ed. by S. N. Freund and E. Yahav. ACM. 466–481. DOI: 10.1145/3453483.3454056.

Boopathy, A., T.-W. Weng, P.-Y. Chen, S. Liu, and L. Daniel. (2019). "Cnn-cert: An efficient framework for certifying robustness of convolutional neural networks". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 3240–3247.

Bouissou, O., E. Goubault, J. Goubault-Larrecq, and S. Putot. (2012). "A generalization of p-boxes to affine arithmetic". *Computing*. 94(2-4): 189–201. DOI: 10.1007/S00607-011-0182-8.

Bouissou, O., E. Goubault, S. Putot, A. Chakarov, and S. Sankaranarayanan. (2016). "Uncertainty Propagation Using Probabilistic Affine Forms and Concentration of Measure Inequalities". In: *Tools and Algorithms for the Construction and Analysis of Systems - 22nd International Conference, TACAS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*. Ed. by M. Chechik and J. Raskin. Vol. 9636. *Lecture Notes in Computer Science*. Springer. 225–243.

Brix, C., S. Bak, T. T. Johnson, and H. Wu. (2024a). "The Fifth International Verification of Neural Networks Competition (VNN-COMP 2024): Summary and Results". *arXiv preprint arXiv:2412.19985*.

Brix, C., S. Bak, T. T. Johnson, and H. Wu. (2024b). "The Fifth International Verification of Neural Networks Competition (VNN-COMP 2024): Summary and Results". URL: https://arxiv.org/abs/2412.19985.

Brix, C., S. Bak, C. Liu, and T. T. Johnson. (2023). "The fourth international verification of neural networks competition (VNN-COMP 2023): Summary and results". *arXiv preprint arXiv:2312.16760*.

Brown, T. B., B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. (2020). "Language Models are Few-Shot Learners". In: *Proc. Advances in Neural Information Processing Systems (NeurIPS)*.

Brückner, B. and A. Lomuscio. (2024). "Verification of Neural Networks against Convolutional Perturbations via Parameterised Kernels". URL: https://arxiv.org/abs/2411.04594.

Bunel, R., J. Lu, I. Turkaslan, P. Kohli, P. Torr, and P. Mudigonda. (2020). "Branch and bound for piecewise linear neural network verification". *Journal of Machine Learning Research*. 21(2020).

Camuto, A., M. Willetts, S. Roberts, C. Holmes, and T. Rainforth. (2021). "Towards a theoretical understanding of the robustness of variational autoencoders". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 3565–3573.

Chakravarthy, A., N. Narodytska, A. Rathis, M. Vilcu, M. Sharif, and G. Singh. (2022). "Property-Driven Evaluation of RL-Controllers in Self-Driving Datacenters". In: *Workshop on Challenges in Deploying and Monitoring Machine Learning Systems (DMML)*.

Chang, Y.-C., N. Roohi, and S. Gao. (2019). "Neural lyapunov control". *Advances in Neural Information Processing Systems*. 32.

Chaudhary, I., Q. Hu, M. Kumar, M. Ziyadi, R. Gupta, and G. Singh. (2025). "Certifying Counterfactual Bias in LLMs". In: *The Thirteenth International Conference on Learning Representations*. URL: https://openreview.net/forum?id=HQHnhVQznF.

Chaudhary, I., V. V. Jain, and G. Singh. (2024a). "Decoding Intelligence: A Framework for Certifying Knowledge Comprehension in LLMs". URL: https://arxiv.org/abs/2402.15929.

Chaudhary, I., S. Lin, C. Tan, and G. Singh. (2024b). "Specification Generation for Neural Networks in Systems". *CoRR*. abs/2412.03028. DOI: 10.48550/ARXIV.2412.03028.

Chen, Y., S. Wang, Y. Qin, X. Liao, S. Jana, and D. A. Wagner. (2021). "Learning Security Classifiers with Verified Global Robustness Properties". In: *Proc. Conference on Computer and Communications Security (CCS)*. ACM. 477–494.

Chevalier, S., I. Murzakhanov, and S. Chatzivasileiadis. (2023). "GPU-Accelerated Verification of Machine Learning Models for Power Systems". *arXiv preprint arXiv:2306.10617*.

Chevalier, S., D. Starkenburg, and K. Dvijotham. (2024). "Achieving the Tightest Relaxation of Sigmoids for Formal Verification". URL: https://arxiv.org/abs/2408.10491.

Chiang, P., R. Ni, A. Abdelkader, C. Zhu, C. Studer, and T. Goldstein. (2020). "Certified Defenses for Adversarial Patches". In: *Proc. International Conference on Learning Representations (ICLR)*.

Chugh, U., A. Mitra, A. Deshwal, N. P. Swaroop, A. Saluja, S. Lee, and J. Song. (2021). "An Automated Approach to Accelerate DNNs on Edge Devices". In: *ISCAS*. IEEE. 1–5.

Cohen, N., M. Ducoffe, R. Boumazouza, C. Gabreau, C. Pagetti, X. Pucel, and A. Galametz. (2024). "Verification for Object Detection – IBP IoU". URL: https://arxiv.org/abs/2403.08788.

Corliss, G. F. and L. B. Rall. (1991). "Computing the range of derivatives". *IMACS Annals on Computing and Applied Mathematics,(to appear)*.

Cousot, P. (2021). *Principles of abstract interpretation*. MIT Press.

Cousot, P. and R. Cousot. (1977). "Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints". In: *Conference Record of the Fourth ACM Symposium on Principles of Programming Languages, Los Angeles, California, USA, January 1977*. ACM. 238–252.

Cousot, P. and N. Halbwachs. (1978). "Automatic Discovery of Linear Restraints Among Variables of a Program". In: *Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages, Tucson, Arizona, USA, January 1978*. ACM Press. 84–96.

Cousot, P. and M. Monerau. (2012). "Probabilistic abstract interpretation". In: *European Symposium on Programming*. Springer. 169–193.

Darwiche, A. and A. Hirth. (2020). "On the Reasons Behind Decisions". In: *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*. Ed. by G. D. Giacomo, A. Catalá, B. Dilkina, M. Milano, S. Barro, A. Bugarín, and J. Lang. Vol. 325. *Frontiers in Artificial Intelligence and Applications*. IOS Press. 712–720. DOI: 10.3233/FAIA200158.

Dathathri, S., K. Dvijotham, A. Kurakin, A. Raghunathan, J. Uesato, R. Bunel, S. Shankar, J. Steinhardt, I. J. Goodfellow, P. Liang, and P. Kohli. (2020). "Enabling certification of verification-agnostic networks via memory-efficient semidefinite programming". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Demarchi, S., D. Guidotti, L. Pulina, A. Tacchella, N. Narodytska, G. Amir, G. Katz, and O. Isac. (2023). "Supporting Standardization of Neural Networks Verification with VNNLIB and CoCoNet." In: *FoMLAS@ CAV*. 47–58.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 248–255.

Deussen, J. (2021). "Global Derivatives". *PhD thesis*.

Dimitrov, D. I., G. Singh, T. Gehr, and M. T. Vechev. (2022). "Provably Robust Adversarial Examples". In: *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net. URL: https://openreview.net/forum?id=UMfhoMtIaP5.

Duong, H., L. Li, T. Nguyen, and M. B. Dwyer. (2023). "A DPLL(T) Framework for Verifying Deep Neural Networks". *CoRR*. abs/2307.10266. DOI: 10.48550/ARXIV.2307.10266.

Dvijotham, K., M. Garnelo, A. Fawzi, and P. Kohli. (2018a). "Verification of deep probabilistic models". URL: https://arxiv.org/abs/1812.02795.

Dvijotham, K., R. Stanforth, S. Gowal, T. A. Mann, and P. Kohli. (2018b). "A Dual Approach to Scalable Verification of Deep Networks". In: *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*. Ed. by A. Globerson and R. Silva. AUAI Press. 550–559. URL: http://auai.org/uai2018/proceedings/papers/204.pdf.

Dwork, C., M. Hardt, T. Pitassi, O. Reingold, and R. S. Zemel. (2012). "Fairness through awareness". In: *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*. ACM. 214–226.

Dwork, C., F. McSherry, K. Nissim, and A. Smith. (2006). "Calibrating noise to sensitivity in private data analysis". In: *Theory of cryptography conference*.

Eiras, F., A. Bibi, R. R. Bunel, K. D. Dvijotham, P. Torr, and M. P. Kumar. (2023). "Efficient Error Certification for Physics-Informed Neural Networks". In: *Forty-first International Conference on Machine Learning*.

Fan, J. and W. Li. (2020). "Adversarial Training and Provable Robustness: A Tale of Two Objectives". *CoRR*. abs/2008.06081. URL: https://arxiv.org/abs/2008.06081.

Fazlyab, M., M. Morari, and G. J. Pappas. (2019a). "Probabilistic Verification and Reachability Analysis of Neural Networks via Semidefinite Programming". URL: https://arxiv.org/abs/1910.04249.

Fazlyab, M., A. Robey, H. Hassani, M. Morari, and G. Pappas. (2019b). "Efficient and accurate estimation of lipschitz constants for deep neural networks". *Advances in neural information processing systems*. 32.

Ferrari, C., M. N. Mueller, N. Jovanović, and M. Vechev. (2022). "Complete Verification via Multi-Neuron Relaxation Guided Branch-and-Bound". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=l_amHf1oaK.

Ferson, S., V. Kreinovich, L. Grinzburg, D. Myers, and K. Sentz. (2015). "Constructing probability boxes and Dempster-Shafer structures". *Sandia journal manuscript; Not yet accepted for publication*. May. URL: https://www.osti.gov/biblio/1427258.

Fischer, M., C. Sprecher, D. I. Dimitrov, G. Singh, and M. T. Vechev. (2022). "Shared Certificates for Neural Network Verification". In: *Computer Aided Verification - 34th International Conference, CAV 2022, Haifa, Israel, August 7-10, 2022, Proceedings, Part I*. Vol. 13371. *Lecture Notes in Computer Science*. Springer. 127–148.

Gehr, T., M. Mirman, D. Drachsler-Cohen, P. Tsankov, S. Chaudhuri, and M. T. Vechev. (2018). "AI2: Safety and Robustness Certification of Neural Networks with Abstract Interpretation". In: *2018 IEEE Symposium on Security and Privacy, SP 2018, Proceedings, 21-23 May 2018, San Francisco, California, USA*. 3–18. DOI: 10.1109/SP.2018.00058.

Geng, C., N. Le, X. Xu, Z. Wang, A. Gurfinkel, and X. Si. (2022). "Towards Reliable Neural Specifications". In: *International Conference on Machine Learning*. URL: https://api.semanticscholar.org/CorpusID:253224120.

Geng, C., Z. Wang, H. Ye, S. Liao, and X. Si. (2024). "Learning Minimal NAP Specifications for Neural Network Verification". *CoRR*. abs/2404.04662. DOI: 10.48550/ARXIV.2404.04662.

Gholami, A., S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer. (2022). "A survey of quantization methods for efficient neural network inference". In: *Low-Power Computer Vision*. Chapman and Hall/CRC. 291–326.

Gholami, A., S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer. (2021). "A Survey of Quantization Methods for Efficient Neural Network Inference". *CoRR*. abs/2103.13630.

Ghorbal, K., E. Goubault, and S. Putot. (2009). "The Zonotope Abstract Domain Taylor1+". In: *Computer Aided Verification, 21st International Conference, CAV 2009, Grenoble, France, June 26 - July 2, 2009. Proceedings*. Ed. by A. Bouajjani and O. Maler. Vol. 5643. *Lecture Notes in Computer Science*. Springer. 627–633.

Goan, E. and C. Fookes. (2020). "Bayesian Neural Networks: An Introduction and Survey". In: *Case Studies in Applied Bayesian Data Science*. Springer International Publishing. 45–87. DOI: 10.1007/978-3-030-42553-1_3.

Goldberg, A. V. and R. E. Tarjan. (1989). "Finding minimum-cost circulations by canceling negative cycles". *J. ACM*. 36(4): 873–886. DOI: 10.1145/76359.76368.

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

Goodman, N., V. Mansinghka, D. M. Roy, K. Bonawitz, and J. Tenenbaum. (2008). "Church: a language for generative models with non-parametric memoization and approximate inference". In: *Uncertainty in Artificial Intelligence*. 165.

Gopinath, D., H. Converse, C. S. Pasareanu, and A. Taly. (2020). "Property Inference for Deep Neural Networks". URL: https://arxiv.org/abs/1904.13215.

Goubault, E., S. Palumby, S. Putot, L. Rustenholz, and S. Sankaranarayanan. (2021). "Static Analysis of ReLU Neural Networks with Tropical Polyhedra". In: *Static Analysis*. Ed. by C. Drăgoi, S. Mukherjee, and K. Namjoshi. Cham: Springer International Publishing. 166–190.

Goubault, E. and S. Putot. (2022). "Rino: Robust inner and outer approximated reachability of neural networks controlled systems". In: *International Conference on Computer Aided Verification*. Springer. 511–523.

Goubault, E. and S. Putot. (2024). "A Zonotopic Dempster-Shafer Approach to the Quantitative Verification of Neural Networks". In: *International Symposium on Formal Methods*. Springer. 324–342.

Goubault, E. and S. Putot. (2025). "A Zonotopic Dempster-Shafer Approach to the Quantitative Verification of Neural Networks". In: *Formal Methods*. Ed. by A. Platzer, K. Y. Rozier, M. Pradella, and M. Rossi. Cham: Springer Nature Switzerland. 324–342.

Gowal, S., K. Dvijotham, R. Stanforth, R. Bunel, C. Qin, J. Uesato, R. Arandjelovic, T. A. Mann, and P. Kohli. (2018). "On the Effectiveness of Interval Bound Propagation for Training Verifiably Robust Models". *CoRR*. abs/1810.12715.

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

Griewank, A. and A. Walther. (2008). *Evaluating derivatives: principles and techniques of algorithmic differentiation*. SIAM.

Gupta, A., L. Marla, R. Sun, N. Shukla, and A. Kolbeinsson. (2021). "Pender: Incorporating shape constraints via penalized derivatives". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. No. 13. 11536–11544.

Gurobi Optimization, LLC. (2018). "Gurobi Optimizer Reference Manual".

Habeeb, P., D. D'Souza, K. Lodaya, and P. Prabhakar. (2024). "Interval Image Abstraction for Verification of Camera-Based Autonomous Systems". *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* 43(11): 4310–4321. DOI: 10.1109/TCAD.2024.3448306.

Henriksen, P. and A. Lomuscio. (2021). "DEEPSPLIT: An Efficient Splitting Method for Neural Network Verification via Indirect Effect Analysis". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. Ed. by Z.-H. Zhou. International Joint Conferences on Artificial Intelligence Organization. 2549–2555. DOI: 10.24963/ijcai.2021/351.

Henriksen, P. and A. R. Lomuscio. (2020). "Efficient Neural Network Verification via Adaptive Refinement and Adversarial Search". In: *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*. Ed. by G. D. Giacomo, A. Catalá, B. Dilkina, M. Milano, S. Barro, A. Bugarín, and J. Lang. Vol. 325. *Frontiers in Artificial Intelligence and Applications*. IOS Press. 2513–2520. DOI: 10.3233/FAIA200385.

Heo, J., S. Joo, and T. Moon. (2019). "Fooling Neural Network Interpretations via Adversarial Model Manipulation". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2921–2932.

Hladík, M. (2018). "Testing pseudoconvexity via interval computation". *Journal of Global Optimization*. 71(3): 443–455.

Hladík, M., L. V. Kolev, and I. Skalna. (2021). "Linear interval parametric approach to testing pseudoconvexity". *Journal of Global Optimization*. 79: 351–368.

Hovland, P. D., B. Norris, M. M. Strout, S. Bhowmick, and J. Utke. (2005). "Sensitivity analysis and design optimization through automatic differentiation". In: *Journal of Physics: Conference Series*.

Hu, K., A. Zou, Z. Wang, K. Leino, and M. Fredrikson. (2023a). "Unlocking Deterministic Robustness Certification on ImageNet". In: *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*. Ed. by A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine. URL: http://papers.nips.cc/paper%5C_files/paper/2023/hash/863da9d40547f1d1b18859519ce2dee4-Abstract-Conference.html.

Hu, K., A. Zou, Z. Wang, K. Leino, and M. Fredrikson. (2023b). "Unlocking deterministic robustness certification on imagenet". *Advances in Neural Information Processing Systems*. 36: 42993–43011.

Huang, Z., S. Dutta, and S. Misailovic. (2021). "Aqua: Automated quantized inference for probabilistic programs". In: *Automated Technology for Verification and Analysis: 19th International Symposium, ATVA 2021, Gold Coast, QLD, Australia, October 18–22, 2021, Proceedings 19*. Springer. 229–246.

Hückelheim, J., Z. Luo, S. H. K. Narayanan, S. Siegel, and P. D. Hovland. (2018). "Verifying properties of differentiable programs". In: *Static Analysis: 25th International Symposium, SAS 2018, Freiburg, Germany, August 29–31, 2018, Proceedings 25*. 205–222.

Ignatiev, A., N. Narodytska, and J. Marques-Silva. (2019). "Abduction-based explanations for Machine Learning models". In: *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence. AAAI'19/IAAI'19/EAAI'19*. Honolulu, Hawaii, USA: AAAI Press. DOI: 10.1609/aaai.v33i01.33011511.

Ivanov, R., J. Weimer, R. Alur, G. J. Pappas, and I. Lee. (2019). "Verisig: Verifying Safety Properties of Hybrid Systems with Neural Network Controllers". In: *Proc. Hybrid Systems: Computation and Control (HSCC)*. 169–178.

Jaderberg, M., K. Simonyan, A. Zisserman, and K. Kavukcuoglu. (2015). "Spatial Transformer Networks". In: *Proc. Neural Information Processing Systems (NeurIPS)*. 2017–2025.

Jia, R., A. Raghunathan, K. Göksel, and P. Liang. (2019). "Certified Robustness to Adversarial Word Substitutions". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*. Ed. by K. Inui, J. Jiang, V. Ng, and X. Wan. Association for Computational Linguistics. 4127–4140.

Jiang, E. and G. Singh. (2024). "Towards Universal Certified Robustness with Multi-Norm Training". URL: https://arxiv.org/abs/2410.03000.

Jin, S., F. Y. Yan, C. Tan, A. Kalia, X. Foukas, and Z. M. Mao. (2024). "AutoSpec: Automated Generation of Neural Network Specifications". URL: https://arxiv.org/abs/2409.10897.

Jordan, M. and A. Dimakis. (2021). "Provable Lipschitz certification for generative models". In: *International Conference on Machine Learning*. PMLR. 5118–5126.

Jordan, M. and A. G. Dimakis. (2020). "Exactly Computing the Local Lipschitz Constant of ReLU Networks". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin. URL: https://proceedings.neurips.cc/paper/2020/hash/5227fa9a19dce7ba113f50a405dcaf09-Abstract.html.

Jovanovic, N., M. Balunovic, M. Baader, and M. T. Vechev. (2022). "On the Paradox of Certified Training". *Trans. Mach. Learn. Res.* 2022.

Kabaha, A. and D. Drachsler-Cohen. (2022). "Boosting Robustness Verification of Semantic Feature Neighborhoods". In: *Static Analysis - 29th International Symposium, SAS 2022, Auckland, New Zealand, December 5-7, 2022, Proceedings*. Vol. 13790. *Lecture Notes in Computer Science*. Springer. 299–324.

Kabaha, A. and D. Drachsler-Cohen. (2024). "Verification of Neural Networks' Global Robustness". *Proc. ACM Program. Lang.* 8(OOPSLA1): 1010–1039. DOI: 10.1145/3649847.

Katz, G., C. Barrett, D. Dill, K. Julian, and M. Kochenderfer. (2017). "Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks". In: *Proc. 29th Int. Conf. on Computer Aided Verification (CAV)*. 97–117.

Katz, G., D. Huang, D. Ibeling, K. Julian, C. Lazarus, R. Lim, P. Shah, S. Thakoor, H. Wu, A. Zeljić, D. Dill, M. Kochenderfer, and C. Barrett. (2019). "The Marabou Framework for Verification and Analysis of Deep Neural Networks". In: 443–452.

Kingma, D. P. and J. Ba. (2015). "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kingma, D. P. and M. Welling. (2019). "An Introduction to Variational Autoencoders". *Foundations and Trends® in Machine Learning*. 12(4): 307–392. DOI: 10.1561/2200000056.

Ko, C., Z. Lyu, L. Weng, L. Daniel, N. Wong, and D. Lin. (2019). "POPQORN: Quantifying Robustness of Recurrent Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by K. Chaudhuri and R. Salakhutdinov. Vol. 97. *Proceedings of Machine Learning Research*. PMLR. 3468–3477. URL: http://proceedings.mlr.press/v97/ko19a.html.

Kochdumper, N. and M. Althoff. (2021). "Sparse Polynomial Zonotopes: A Novel Set Representation for Reachability Analysis". *IEEE Transactions on Automatic Control*. 66(9): 4043–4058. DOI: 10.1109/TAC.2020.3024348.

Kochdumper, N., C. Schilling, M. Althoff, and S. Bak. (2023). "Open- and Closed-Loop Neural Network Verification Using Polynomial Zonotopes". In: *NASA Formal Methods*. Springer Nature Switzerland. 16–36. DOI: 10.1007/978-3-031-33170-1_2.

König, M., A. W. Bosman, H. H. Hoos, and J. N. van Rijn. (2024a). "Critically Assessing the State of the Art in Neural Network Verification". *Journal of Machine Learning Research*. 25(12): 1–53. URL: http://jmlr.org/papers/v25/23-0119.html.

König, M., X. Zhang, H. H. Hoos, M. Kwiatkowska, and J. N. van Rijn. (2024b). "Automated Design of Linear Bounding Functions for Sigmoidal Nonlinearities in Neural Networks". In: *Machine Learning and Knowledge Discovery in Databases. Research Track - European Conference, ECML PKDD 2024, Vilnius, Lithuania, September 9-13, 2024, Proceedings, Part VII*. Ed. by A. Bifet, J. Davis, T. Krilavicius, M. Kull, E. Ntoutsi, and I. Zliobaite. Vol. 14947. *Lecture Notes in Computer Science*. Springer. 383–398. DOI: 10.1007/978-3-031-70368-3_23.

Kos, J., I. Fischer, and D. Song. (2018). "Adversarial examples for generative models". In: *2018 ieee security and privacy workshops (spw)*. IEEE. 36–42.

Kotha, S., C. Brix, Z. Kolter, K. Dvijotham, and H. Zhang. (2023). "Provably Bounding Neural Network Preimages". *CoRR*. abs/2302.01404. DOI: 10.48550/ARXIV.2302.01404.

Krizhevsky, A. (2009). "Learning Multiple Layers of Features from Tiny Images".

Kurakin, A., I. J. Goodfellow, and S. Bengio. (2017). "Adversarial examples in the physical world". In: *ICLR (Workshop)*. OpenReview.net.

Kurin, V., A. D. Palma, I. Kostrikov, S. Whiteson, and M. P. Kumar. (2022). "In Defense of the Unitary Scalarization for Deep Multi-Task Learning". In: *Advances in Neural Information Processing Systems*. Ed. by A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho. URL: https://openreview.net/forum?id=wmwgLEPjL9.

Ladner, T. and M. Althoff. (2023). "Automatic Abstraction Refinement in Neural Network Verification using Sensitivity Analysis". In: *Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control. HSCC '23*. San Antonio, TX, USA: Association for Computing Machinery. DOI: 10.1145/3575870.3587129.

Ladner, T. and M. Althoff. (2024). "Exponent Relaxation of Polynomial Zonotopes and Its Applications in Formal Neural Network Verification". In: *AAAI*. 21304–21311. URL: https://doi.org/10.1609/aaai.v38i19.30125.

Ladner, T., M. Eichelbeck, and M. Althoff. (2024). "Formal Verification of Graph Convolutional Networks with Uncertain Node Features and Uncertain Graph Structure". URL: https://arxiv.org/abs/2404.15065.

Lan, J., Y. Zheng, and A. Lomuscio. (2022). "Tight Neural Network Verification via Semidefinite Relaxations and Linear Reformulations". In: *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022,* AAAI Press. 7272–7280.

Laurel, J. (2024). "Static Analysis of Differentiable Programs". *PhD thesis*. University of Illinois at Urbana-Champaign.

Laurel, J., S. B. Qian, G. Singh, and S. Misailovic. (2023). "Synthesizing Precise Static Analyzers for Automatic Differentiation". *Proc. ACM Program. Lang.* (OOPSLA2).

Laurel, J., S. B. Qian, G. Singh, and S. Misailovic. (2024). "Abstract Interpretation of Automatic Differentiation". In: *Languages for Inference Workshop (LAFI)*.

Laurel, J., R. Yang, G. Singh, and S. Misailovic. (2022a). "A dual number abstraction for static analysis of Clarke Jacobians". *Proc. ACM Program. Lang.* 6(POPL): 1–30.

Laurel, J., R. Yang, S. Ugare, R. Nagel, G. Singh, and S. Misailovic. (2022b). "A general construction for abstract interpretation of higher-order automatic differentiation". *Proc. ACM Program. Lang.* 6(OOPSLA2): 1007–1035.

LeCun, Y., B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel. (1989). "Handwritten Digit Recognition with a Back-Propagation Network". In: *NIPS*. 396–404.

Leino, K., Z. Wang, and M. Fredrikson. (2021). "Globally-Robust Neural Networks". *CoRR*. abs/2102.08452. URL: https://arxiv.org/abs/2102.08452.

Lemesle, A., J. Lehmann, and T. L. Gall. (2024). "Neural Network Verification with PyRAT". URL: https://arxiv.org/abs/2410.23903.

Lerman, S., C. Venuto, H. Kautz, and C. Xu. (2021). "Explaining local, global, and higher-order interactions in deep learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1224–1233.

Li, J., S. Qu, X. Li, J. Szurley, J. Z. Kolter, and F. Metze. (2019a). "Adversarial Music: Real world Audio Adversary against Wake-word Detection System". In: *Proc. Neural Information Processing Systems (NeurIPS)*. 11908–11918.

Li, J., F. R. Schmidt, and J. Z. Kolter. (2019b). "Adversarial camera stickers: A physical camera-based attack on deep learning systems". In: *Proc. International Conference on Machine Learning, ICML*. Vol. 97. 3896–3904.

Li, L., X. Qi, T. Xie, and B. Li. (2020). "SoK: Certified Robustness for Deep Neural Networks". *CoRR*. abs/2009.04131. URL: https://arxiv.org/abs/2009.04131.

Liang, T., J. Glossner, L. Wang, S. Shi, and X. Zhang. (2021). "Pruning and quantization for deep neural network acceleration: A survey". *Neurocomputing*. 461: 370–403.

Liao, Z. and M. Cheung. (2022). "Automated Invariance Testing for Machine Learning Models Using Sparse Linear Layers". In: *ICML 2022: Workshop on Spurious Correlations, Invariance and Stability*. URL: https://openreview.net/forum?id=VP8ATzLGyQx.

Lin, J., C. Gan, and S. Han. (2019). "Defensive Quantization: When Efficiency Meets Robustness". In: *International Conference on Learning Representations*.

Lin, S., H. He, T. Wei, K. Xu, H. Zhang, G. Singh, C. Liu, and C. Tan. (2024). "NN4SysBench: Characterizing Neural Network Verification for Computer Systems". *Advances in Neural Information Processing Systems*. 37: 91390–91404.

Liu, Z., G. Singh, C. Xu, and D. Vasisht. (2021). "FIRE: enabling reciprocity for FDD MIMO systems". In: *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking. MobiCom '21*. New Orleans, Louisiana: Association for Computing Machinery. 628–641. DOI: 10.1145/3447993.3483275.

Liu, Z., C. Xu, Y. Xie, E. Sie, F. Yang, K. Karwaski, G. Singh, Z. L. Li, Y. Zhou, D. Vasisht, *et al.* (2023). "Exploring practical vulnerabilities of machine learning-based wireless systems". In: *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*. 1801–1817.

Lu, J. and M. P. Kumar. (2019). "Neural Network Branching for Neural Network Verification". URL: https://arxiv.org/abs/1912.01329.

Lundberg, S. M. and S. Lee. (2017). "A unified approach to interpreting model predictions". *CoRR*. abs/1705.07874. URL: http://arxiv.org/abs/1705.07874.

Lyu, Z., M. Guo, T. Wu, G. Xu, K. Zhang, and D. Lin. (2021). "Towards Evaluating and Training Verifiably Robust Neural Networks". URL: https://arxiv.org/abs/2104.00447.

Lyu, Z., C.-Y. Ko, Z. Kong, N. Wong, D. Lin, and L. Daniel. (2020). "Fastened crown: Tightened neural network robustness certificates". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 04. 5037–5044.

Ma, Y., V. Dixit, M. J. Innes, X. Guo, and C. Rackauckas. (2021). "A comparison of automatic differentiation and continuous sensitivity analysis for derivatives of differential equation solutions". In: *2021 IEEE High Performance Extreme Computing Conference (HPEC)*. IEEE. 1–9.

Ma, Z. (2023). "Verifying Neural Networks by Approximating Convex Hulls". In: *Formal Methods and Software Engineering - 24th International Conference on Formal Engineering Methods, ICFEM 2023, Brisbane, QLD, Australia, November 21-24, 2023, Proceedings*. Ed. by Y. Li and S. Tahar. Vol. 14308. *Lecture Notes in Computer Science*. Springer. 261–266. DOI: 10.1007/978-981-99-7584-6_17.

Ma, Z., J. Li, and G. Bai. (2024). "ReLU Hull Approximation". *Proc. ACM Program. Lang.* 8(POPL). DOI: 10.1145/3632917.

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

Malfa, E. L., R. Michelmore, A. M. Zbrzezny, N. Paoletti, and M. Kwiatkowska. (2021). "On Guaranteed Optimal Robust Explanations for NLP Models". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*. Ed. by Z. Zhou. ijcai.org. 2658–2665. DOI: 10.24963/IJCAI.2021/366.

Mametjanov, A., B. Norris, X. Zeng, B. Drewniak, J. Utke, M. Anitescu, and P. Hovland. (2012). "Applying automatic differentiation to the Community Land Model". In: *Recent Advances in Algorithmic Differentiation*.

Mangal, R., K. Sarangmath, A. V. Nori, and A. Orso. (2020). "Probabilistic Lipschitz Analysis of Neural Networks". In: *International Static Analysis Symposium*.

Mao, Y., S. Balauca, and M. Vechev. (2024). "CTBENCH: A Library and Benchmark for Certified Training". URL: https://arxiv.org/abs/2406.04848.

Mao, Y., M. N. Mueller, M. Fischer, and M. Vechev. (2023). "Connecting Certified and Adversarial Training". In: *Thirty-seventh Conference on Neural Information Processing Systems*. URL: https://openreview.net/forum?id=T2lM4ohRwb.

Mardziel, P., S. Magill, M. Hicks, and M. Srivatsa. (2013). "Dynamic enforcement of knowledge-based security policies using probabilistic abstract interpretation". *J. Comput. Secur.* 21(4): 463–532.

Marques-Silva, J. and A. Ignatiev. (2022). "Delivering Trustworthy AI through Formal XAI". In: *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*. AAAI Press. 12342–12350. DOI: 10.1609/AAAI.V36I11.21499.

Miné, A. (2002). "A Few Graph-Based Relational Numerical Abstract Domains". In: *Static Analysis, 9th International Symposium, SAS 2002, Madrid, Spain, September 17-20, 2002, Proceedings*. Ed. by M. V. Hermenegildo and G. Puebla. Vol. 2477. *Lecture Notes in Computer Science*. Springer. 117–132. DOI: 10.1007/3-540-45789-5_11.

Miné, A. (2006). "The octagon abstract domain". *High. Order Symb. Comput.* 19(1): 31–100.

Miné, A. (2017). "Tutorial on static inference of numeric invariants by abstract interpretation". *Foundations and Trends® in Programming Languages*. 4(3-4): 120–372.

Mirman, M., T. Gehr, and M. Vechev. (2018). "Differentiable abstract interpretation for provably robust neural networks". In: *Proc. International Conference on Machine Learning (ICML)*. 3578–3586.

Mirman, M., A. Hägele, P. Bielik, T. Gehr, and M. Vechev. (2021). "Robustness certification with generative models". In: *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation. PLDI 2021*. Virtual, Canada: Association for Computing Machinery. 1141–1154. DOI: 10.1145/3453483.3454100.

Mirman, M., G. Singh, and M. Vechev. (2020). "A Provable Defense for Deep Residual Networks". URL: https://arxiv.org/abs/1903.12519.

Misailovic, S., M. Carbin, S. Achour, Z. Qi, and M. C. Rinard. (2014). "Chisel: Reliability-and accuracy-aware optimization of approximate computational kernels". *ACM Sigplan Notices*. 49(10): 309–328.

Misra, A., J. Laurel, and S. Misailovic. (2023). "ViX: Analysis-driven Compiler for Efficient Low-Precision Variational Inference". In: *Design, Automation & Test in Europe Conference & Exhibition (DATE)*.

Mitchell, D. and P. Hanrahan. (1992). "Illumination from curved reflectors". *SIGGRAPH Comput. Graph.* 26(2). DOI: 10.1145/142920.134082.

Mitra, S., C. S. Pasareanu, P. Prabhakar, S. A. Seshia, R. Mangal, Y. Li, C. Watson, D. Gopinath, and H. Yu. (2024). "Formal Verification Techniques for Vision-Based Autonomous Systems - A Survey". In: *Principles of Verification: Cycling the Probabilistic Landscape - Essays Dedicated to Joost-Pieter Katoen on the Occasion of His 60th Birthday, Part III*. Ed. by N. Jansen, S. Junges, B. L. Kaminski, C. Matheja, T. Noll, T. Quatmann, M. Stoelinga, and M. Volk. Vol. 15262. *Lecture Notes in Computer Science*. Springer. 89–108. DOI: 10.1007/978-3-031-75778-5_5.

Mohapatra, J., T. Weng, P. Chen, S. Liu, and L. Daniel. (2020). "Towards Verifying Robustness of Neural Networks Against A Family of Semantic Perturbations". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE. 241–249.

Mohr, S., K. Drainas, and J. Geist. (2021). "Assessment of Neural Networks for Stream-Water-Temperature Prediction". *CoRR*. abs/2110.04254. URL: https://arxiv.org/abs/2110.04254.

Müller, C., F. Serre, G. Singh, M. Püschel, and M. T. Vechev. (2021a). "Scaling Polyhedral Neural Network Verification on GPUs". In: *Proceedings of Machine Learning and Systems 2021, MLSys 2021, virtual, April 5-9, 2021*. mlsys.org.

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

Müller, M. N., F. Eckert, M. Fischer, and M. T. Vechev. (2023a). "Certified Training: Small Boxes are All You Need". In: *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Müller, M. N., M. Fischer, R. Staab, and M. T. Vechev. (2023b). "Abstract Interpretation of Fixpoint Iterators with Applications to Neural Networks". *Proc. ACM Program. Lang.* 7(PLDI): 786–810.

Müller, M. N., G. Makarchuk, G. Singh, M. Püschel, and M. Vechev. (2021b). "Precise Multi-Neuron Abstractions for Neural Network Certification". *arXiv preprint arXiv:2103.03638*.

Munakata, S., C. Urban, H. Yokoyama, K. Yamamoto, and K. Munakata. (2023). "Verifying Attention Robustness of Deep Neural Networks Against Semantic Perturbations". In: *NASA Formal Methods - 15th International Symposium, NFM 2023, Houston, TX, USA, May 16-18, 2023, Proceedings*. Vol. 13903. *Lecture Notes in Computer Science*. Springer. 37–61.

Nielson, F., H. R. Nielson, and C. Hankin. (2005). *Principles of program analysis*. Springer.

Palma, A. D., H. S. Behl, R. Bunel, P. H. S. Torr, and M. P. Kumar. (2021a). "Scaling the Convex Barrier with Active Sets". In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Palma, A. D., R. Bunel, A. Desmaison, K. Dvijotham, P. Kohli, P. H. S. Torr, and M. P. Kumar. (2021b). "Improved Branch and Bound for Neural Network Verification via Lagrangian Decomposition". *CoRR*. abs/2104.06718. URL: https://arxiv.org/abs/2104.06718.

Palma, A. D., R. Bunel, K. Dvijotham, M. P. Kumar, and R. Stanforth. (2022). "IBP Regularization for Verified Adversarial Robustness via Branch-and-Bound". *CoRR*. abs/2206.14772. DOI: 10.48550/ARXIV.2206.14772.

Palma, A. D., R. Bunel, K. (Dvijotham, M. P. Kumar, R. Stanforth, and A. Lomuscio. (2024). "Expressive Losses for Verified Robustness via Convex Combinations". In: *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net. URL: https://openreview.net/forum?id=mzyZ4wzKlM.

Pan, L., A. Albalak, X. Wang, and W. Y. Wang. (2023). "Logic-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning". *CoRR*. abs/2305.12295. DOI: 10.48550/ARXIV.2305.12295.

Păsăreanu, C., H. Converse, A. Filieri, and D. Gopinath. (2020). "On the probabilistic analysis of neural networks". In: *Proceedings of the IEEE/ACM 15th International Symposium on Software Engineering for Adaptive and Self-Managing Systems. SEAMS '20*. Seoul, Republic of Korea: Association for Computing Machinery. 5–8. DOI: 10.1145/3387939.3391594.

Paulsen, B. and C. Wang. (2022). "LinSyn: Synthesizing Tight Linear Bounds for Arbitrary Neural Network Activation Functions". In: *Tools and Algorithms for the Construction and Analysis of Systems - 28th International Conference, TACAS 2022, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2022, Munich, Germany, April 2-7, 2022, Proceedings, Part I*. Ed. by D. Fisman and G. Rosu. Vol. 13243. *Lecture Notes in Computer Science*. Springer. 357–376. DOI: 10.1007/978-3-030-99524-9_19.

Paulsen, B., J. Wang, and C. Wang. (2020). "ReluDiff: differential verification of deep neural networks". In: *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering. ICSE '20*. Seoul, South Korea: Association for Computing Machinery. 714–726. DOI: 10.1145/3377811.3380337.

Pei, K., Y. Cao, J. Yang, and S. Jana. (2017). "DeepXplore: Automated Whitebox Testing of Deep Learning Systems". In: *Proc. Symposium on Operating Systems Principles (SOSP)*. 1–18.

Pilipovsky, J., V. Sivaramakrishnan, M. Oishi, and P. Tsiotras. (2023). "Probabilistic Verification of ReLU Neural Networks via Characteristic Functions". In: *Proceedings of The 5th Annual Learning for Dynamics and Control Conference*. Ed. by N. Matni, M. Morari, and G. J. Pappas. Vol. 211. *Proceedings of Machine Learning Research*. PMLR. 966–979. URL: https://proceedings.mlr.press/v211/pilipovsky23a.html.

Prabhakar, P. and Z. Rahimi Afzal. (2019). "Abstraction based Output Range Analysis for Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/5df0385cba256a135be596dbe28fa7aa-Paper.pdf.

Pulina, L. and A. Tacchella. (2010). "An Abstraction-Refinement Approach to Verification of Artificial Neural Networks". In: *Computer Aided Verification, 22nd International Conference, CAV 2010, Edinburgh, UK, July 15-19, 2010. Proceedings*. Ed. by T. Touili, B. Cook, and P. B. Jackson. Vol. 6174. *Lecture Notes in Computer Science*. Springer. 243–257. DOI: 10.1007/978-3-642-14295-6_24.

Qi, Z., S. Khorram, and F. Li. (2020). "Visualizing Deep Networks by Optimizing with Integrated Gradients". In: *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press. 11890–11898. DOI: 10.1609/AAAI.V34I07.6863.

Qin, Z., T.-W. Weng, and S. Gao. (2022). "Quantifying safety of learning-based self-driving control using almost-barrier functions". In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 12903–12910.

Rackauckas, C., Y. Ma, J. Martensen, C. Warner, K. Zubov, R. Supekar, D. Skinner, A. Ramadhan, and A. Edelman. (2020). "Universal differential equations for scientific machine learning". *arXiv preprint arXiv:2001.04385*.

Ranzato, F., C. Urban, and M. Zanella. (2021). "Fairness-Aware Training of Decision Trees by Abstract Interpretation". In: *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*. ACM. 1508–1517.

Räuker, T., A. Ho, S. Casper, and D. Hadfield-Menell. (2023). "Toward Transparent AI: A Survey on Interpreting the Inner Structures of Deep Neural Networks". URL: https://arxiv.org/abs/2207.13243.

Ribeiro, M. T., S. Singh, and C. Guestrin. (2016). ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". In: *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 1135–1144.

Ribeiro, M. T., S. Singh, and C. Guestrin. (2018). "Anchors: High-Precision Model-Agnostic Explanations". In: *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. Ed. by S. A. McIlraith and K. Q. Weinberger. AAAI Press. 1527–1535. DOI: 10.1609/AAAI.V32I1.11491.

Rival, X. and K. Yi. (2020). *Introduction to static analysis: an abstract interpretation perspective*. MIT Press.

Rober, N., S. M. Katz, C. Sidrane, E. Yel, M. Everett, M. J. Kochenderfer, and J. P. How. (2023). "Backward reachability analysis of neural feedback loops: Techniques for linear and nonlinear systems". *IEEE Open Journal of Control Systems*. 2: 108–124.

Ruan, W., M. Wu, Y. Sun, X. Huang, D. Kroening, and M. Kwiatkowska. (2019). "Global Robustness Evaluation of Deep Neural Networks with Provable Guarantees for the Hamming Distance". In: *Proc. International Joint Conference on Artificial Intelligence, IJCAI*. Ed. by S. Kraus. 5944–5952.

Ryou, W., J. Chen, M. Balunovic, G. Singh, A. Dan, and M. Vechev. (2021). "Scalable Polyhedral Verification of Recurrent Neural Networks". In: *International Conference on Computer Aided Verification*. Springer. 225–248.

Sadraddini, S. and R. Tedrake. (2019). "Linear Encodings for Polytope Containment Problems". In: *Proc. Conference on Decision and Control (CDC)*.

Salman, H., G. Yang, H. Zhang, C.-J. Hsieh, and P. Zhang. (2019). "A Convex Relaxation Barrier to Tight Robustness Verification of Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/246a3c5544feb054f3ea718f61adfa16-Paper.pdf.

Samek, W., G. Montavon, S. Lapuschkin, C. J. Anders, and K. Müller. (2021). "Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications". *Proc. IEEE*. 109(3): 247–278.

Sankaranarayanan, S., A. Chakarov, and S. Gulwani. (2013). "Static analysis for probabilistic programs: inferring whole program properties from finitely many paths". In: *ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI '13*. Ed. by H. Boehm and C. Flanagan. 447–458.

Shafique, M., M. Naseer, T. Theocharides, C. Kyrkou, O. Mutlu, L. Orosa, and J. Choi. (2020). "Robust Machine Learning Systems: Challenges,Current Trends, Perspectives, and the Road Ahead". *IEEE Design Test*. 37(2): 30–57.

Sherman, B., J. Michel, and M. Carbin. (2021). "LambdaS: computable semantics for differentiable programming with higher-order functions and datatypes". *Proceedings of the ACM on Programming Languages*. 5(POPL): 1–31.

Shi, Z., Q. Jin, Z. Kolter, S. Jana, C.-J. Hsieh, and H. Zhang. (2024). "Neural Network Verification with Branch-and-Bound for General Nonlinearities". URL: https://arxiv.org/abs/2405.21063.

Shi, Z., Y. Wang, H. Zhang, Z. Kolter, and C.-J. Hsieh. (2022). "Efficiently Computing Local Lipschitz Constants of Neural Networks via Bound Propagation". In: *Advances in Neural Information Processing Systems*.

Shi, Z., Y. Wang, H. Zhang, J. Yi, and C.-J. Hsieh. (2021). "Fast Certified Robust Training with Short Warmup". URL: https://arxiv.org/abs/2103.17268.

Sidrane, C., A. Maleki, A. Irfan, and M. J. Kochenderfer. (2022). "Overt: An algorithm for safety verification of neural network control policies for nonlinear systems". *Journal of Machine Learning Research*. 23(117): 1–45.

Sill, J. (1997). "Monotonic Networks". In: *Advances in Neural Information Processing Systems 10, [NIPS Conference, Denver, Colorado, USA, 1997]*. The MIT Press. 661–667.

Simon, A. and A. King. (2010). "The two variable per inequality abstract domain". *High. Order Symb. Comput.* 23(1): 87–143. URL: https://doi.org/10.1007/s10990-010-9062-8.

Simonyan, K., A. Vedaldi, and A. Zisserman. (2013). "Deep inside convolutional networks: Visualising image classification models and saliency maps". *arXiv preprint arXiv:1312.6034*.

Singh, A., Y. Sarita, C. Mendis, and G. Singh. (2024). "ConstraintFlow: A DSL for Specification and Verification of Neural Network Analyses". *CoRR*. abs/2403.18729. DOI: 10.48550/ARXIV.2403.18729.

Singh, A., Y. C. Sarita, C. Mendis, and G. Singh. (2025). "Automated Verification of Soundness of DNN Certifiers". *Proc. ACM Program. Lang.* 9(OOPSLA1). DOI: 10.1145/3720509.

Singh, G., R. Ganvir, M. Püschel, and M. Vechev. (2019a). "Beyond the single neuron convex barrier for neural network certification". In: *Advances in Neural Information Processing Systems*.

Singh, G., T. Gehr, M. Mirman, M. Püschel, and M. Vechev. (2018). "Fast and effective robustness certification". *Advances in Neural Information Processing Systems*. 31.

Singh, G., T. Gehr, M. Püschel, and M. Vechev. (2019b). "An abstract domain for certifying neural networks". *Proceedings of the ACM on Programming Languages*. 3(POPL).

Singh, G., T. Gehr, M. Püschel, and M. Vechev. (2019c). "Boosting Robustness Certification of Neural Networks". In: *International Conference on Learning Representations*.

Singh, G., T. Gehr, M. Püschel, and M. Vechev. (2019d). "Robustness Certification with Refinement". In: *International Conference on Learning Representations*.

Singh, G., M. Püschel, and M. T. Vechev. (2017). "Fast polyhedra abstract domain". In: *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, POPL 2017, Paris, France, January 18-20, 2017*. ACM. 46–59.

Singla, S. and S. Feizi. (2021). "Skew Orthogonal Convolutions". URL: https://arxiv.org/abs/2105.11417.

Sivaraman, A., G. Farnadi, T. Millstein, and G. Van den Broeck. (2020). "Counterexample-guided learning of monotonic neural networks". *Neural Information Processing Systems*.

Smilkov, D., N. Thorat, B. Kim, F. Viégas, and M. Wattenberg. (2017). "Smoothgrad: removing noise by adding noise". *arXiv preprint arXiv:1706.03825*.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. (2014). "Dropout: a simple way to prevent neural networks from overfitting". *J. Mach. Learn. Res.* 15(1): 1929–1958.

Suresh, T., D. Banerjee, and G. Singh. (2024). "Relational Verification Leaps Forward with RABBit". In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. URL: https://openreview.net/forum?id=W5U3XB1C11.

Szegedy, C., W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. (2014). "Intriguing properties of neural networks". In: *ICLR (Poster)*.

Tang, X. (2024). "Improved Incremental Verification for Neural Networks". In: *Theoretical Aspects of Software Engineering*. Ed. by W.-N. Chin and Z. Xu. Cham: Springer Nature Switzerland. 392–409.

Tang, X., Y. Zheng, and J. Liu. (2023). "Boosting Multi-neuron Convex Relaxation for Neural Network Verification". In: *Static Analysis - 30th International Symposium, SAS 2023, Cascais, Portugal, October 22-24, 2023, Proceedings*. Ed. by M. V. Hermenegildo and J. F. Morales. Vol. 14284. *Lecture Notes in Computer Science*. Springer. 540–563. DOI: 10.1007/978-3-031-44245-2_23.

Tjandraatmadja, C., R. Anderson, J. Huchette, W. Ma, K. K. Patel, and J. P. Vielma. (2020). "The convex relaxation barrier, revisited: Tightened single-neuron relaxations for neural network verification". *Advances in Neural Information Processing Systems*. 33: 21675–21686.

Tran, H.-D., S. Bak, W. Xiang, and T. T. Johnson. (2020a). "Verification of Deep Convolutional Neural Networks Using ImageStars". In: *Computer Aided Verification: 32nd International Conference, CAV 2020, Los Angeles, CA, USA, July 21–24, 2020, Proceedings, Part I*. 18–42.

Tran, H.-D., S. Choi, H. Okamoto, B. Hoxha, G. Fainekos, and D. Prokhorov. (2023). "Quantitative Verification for Neural Networks using ProbStars". In: *Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control. HSCC '23*. San Antonio, TX, USA: Association for Computing Machinery. DOI: 10.1145/3575870.3587112.

Tran, H., D. M. Lopez, P. Musau, X. Yang, L. V. Nguyen, W. Xiang, and T. T. Johnson. (2019a). "Star-Based Reachability Analysis of Deep Neural Networks". In: *Formal Methods - The Next 30 Years - Third World Congress, FM 2019, Porto, Portugal, October 7-11, 2019, Proceedings*. Ed. by M. H. ter Beek, A. McIver, and J. N. Oliveira. Vol. 11800. *Lecture Notes in Computer Science*. Springer. 670–686. DOI: 10.1007/978-3-030-30942-8_39.

Tran, H.-D., D. Manzanas Lopez, P. Musau, X. Yang, L. V. Nguyen, W. Xiang, and T. T. Johnson. (2019b). "Star-Based Reachability Analysis of Deep Neural Networks". In: *Formal Methods – The Next 30 Years*. Cham: Springer International Publishing. 670–686.

Tran, H., N. Pal, D. M. Lopez, P. Musau, X. Yang, L. V. Nguyen, W. Xiang, S. Bak, and T. T. Johnson. (2021). "Verification of piecewise deep neural networks: a star set approach with zonotope pre-filter". *Formal Aspects Comput.* 33(4-5): 519–545.

Tran, H., X. Yang, D. M. Lopez, P. Musau, L. V. Nguyen, W. Xiang, S. Bak, and T. T. Johnson. (2020b). "NNV: The Neural Network Verification Tool for Deep Neural Networks and Learning-Enabled Cyber-Physical Systems". In: *Computer Aided Verification - 32nd International Conference, CAV 2020, Los Angeles, CA, USA, July 21-24, 2020, Proceedings, Part I*. Ed. by S. K. Lahiri and C. Wang. Vol. 12224. *Lecture Notes in Computer Science*. Springer. 3–17.

Trockman, A. and J. Z. Kolter. (2021). "Orthogonalizing Convolutional Layers with the Cayley Transform". *CoRR*. abs/2104.07167. URL: https://arxiv.org/abs/2104.07167.

Tsipras, D., S. Santurkar, L. Engstrom, A. Turner, and A. Madry. (2019). "Robustness May Be at Odds with Accuracy". In: *proc. International Conference on Learning Representations, ICLR*. OpenReview.net.

Tsuzuku, Y., I. Sato, and M. Sugiyama. (2018). "Lipschitz-margin training: scalable certification of perturbation invariance for deep neural networks". In: *Neural Information Processing Systems*.

Ugare, S., D. Banerjee, S. Misailovic, and G. Singh. (2023). "Incremental Verification of Neural Networks". *Proc. ACM Program. Lang.* 7(PLDI).

Ugare, S., G. Singh, and S. Misailovic. (2022). "Proof transfer for fast certification of multiple approximate neural networks". *Proc. ACM Program. Lang.* 6(OOPSLA1): 1–29.

Urban, C., M. Christakis, V. Wüstholz, and F. Zhang. (2020a). "Perfectly parallel fairness certification of neural networks". *Proc. ACM Program. Lang.* 4(OOPSLA): 185:1–185:30. DOI: 10.1145/3428253.

Urban, C., M. Christakis, V. Wüstholz, and F. Zhang. (2020b). "Perfectly parallel fairness certification of neural networks". *Proc. ACM Program. Lang.* 4(OOPSLA). DOI: 10.1145/3428253.

Vassiliadis, V., J. Riehme, J. Deussen, K. Parasyris, C. D. Antonopoulos, N. Bellas, S. Lalis, and U. Naumann. (2016). "Towards automatic significance analysis for approximate computing". In: *2016 IEEE/ACM International Symposium on Code Generation and Optimization*. No. CGO.

Wang, S., K. Pei, J. Whitehouse, J. Yang, and S. Jana. (2018). "Efficient formal safety analysis of neural networks". In: *Advances in Neural Information Processing Systems*.

Wang, S., H. Zhang, K. Xu, X. Lin, S. Jana, C.-J. Hsieh, and J. Z. Kolter. (2021). "Beta-crown: Efficient bound propagation with per-neuron split constraints for complete and incomplete neural network verification". *arXiv preprint arXiv:2103.06624*.

Wang, X., M. Hersche, B. Tömekce, B. Kaya, M. Magno, and L. Benini. (2020). "An Accurate EEGNet-based Motor-Imagery Brain-Computer Interface for Low-Power Edge Computing". In: *IEEE International Symposium on Medical Measurements and Applications, (MeMeA)*. IEEE. 1–6.

Wang, Z., C. Huang, and Q. Zhu. (2022a). "Efficient global robustness certification of neural networks via interleaving twin-network encoding". In: *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE. 1087–1092.

Wang, Z., A. Albarghouthi, G. Prakriya, and S. Jha. (2022b). "Interval universal approximation for neural networks". *Proc. ACM Program. Lang.* 6(POPL): 1–29. DOI: 10.1145/3498675.

Webb, S., T. Rainforth, Y. W. Teh, and M. P. Kumar. (2019). "A Statistical Approach to Assessing Neural Network Robustness". In: *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. URL: https://openreview.net/forum?id=S1xcx3C5FX.

Wei, T., Z. Jia, C. Liu, and C. Tan. (2023). "Building Verified Neural Networks for Computer Systems with Ouroboros". In: *Sixth Conference on Machine Learning and Systems*. Sixth Conference on Machine Learning and Systems.

Weng, L., H. Zhang, H. Chen, Z. Song, C.-J. Hsieh, L. Daniel, D. Boning, and I. Dhillon. (2018). "Towards fast computation of certified robustness for relu networks". In: *International Conference on Machine Learning*. PMLR. 5276–5285.

Wengert, R. E. (1964). "A simple automatic derivative evaluation program". *Communications of the ACM*. 7(8): 463–464.

Wicker, M., L. Laurenti, A. Patane, and M. Kwiatkowska. (2020). "Probabilistic Safety for Bayesian Neural Networks". In: *Proceedings of the Thirty-Sixth Conference on Uncertainty in Artificial Intelligence, UAI 2020, virtual online, August 3-6, 2020*. Ed. by R. P. Adams and V. Gogate. Vol. 124. *Proceedings of Machine Learning Research*. AUAI Press. 1198–1207. URL: http://proceedings.mlr.press/v124/wicker20a.html.

Wicker, M., A. Patane, L. Laurenti, and M. Kwiatkowska. (2023). "Adversarial Robustness Certification for Bayesian Neural Networks". *CoRR*. abs/2306.13614. DOI: 10.48550/ARXIV.2306.13614. arXiv: 2306.13614.

Wong, E. and J. Z. Kolter. (2018). "Provable Defenses against Adversarial Examples via the Convex Outer Adversarial Polytope". In: *Proc. International Conference on Machine Learning, ICML*. Vol. 80. *Proceedings of Machine Learning Research*. PMLR. 5283–5292.

Wong, E., S. Santurkar, and A. Madry. (2021). "Leveraging Sparse Linear Layers for Debuggable Deep Networks". In: *Proceedings of the 38th International Conference on Machine Learning, ICML*. Vol. 139. *Proceedings of Machine Learning Research*. PMLR. 11205–11216.

Wong, E., F. R. Schmidt, J. H. Metzen, and J. Z. Kolter. (2018). "Scaling provable adversarial defenses". In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. 8410–8419. URL: https://proceedings.neurips.cc/paper/2018/hash/358f9e7be09177c17d0d17ff73584307-Abstract.html.

Wu, C., R. Raghavendra, U. Gupta, B. Acun, N. Ardalani, K. Maeng, G. Chang, F. A. Behram, J. Huang, C. Bai, M. Gschwind, A. Gupta, M. Ott, A. Melnikov, S. Candido, D. Brooks, G. Chauhan, B. Lee, H. S. Lee, B. Akyildiz, M. Balandat, J. Spisak, R. Jain, M. Rabbat, and K. Hazelwood. (2022a). "Sustainable AI: Environmental Implications, Challenges and Opportunities". In: *MLSys*. mlsys.org.

Wu, H., C. Barrett, M. Sharif, N. Narodytska, and G. Singh. (2022b). "Scalable Verification of GNN-Based Job Schedulers". *Proc. ACM Program. Lang.* 6(OOPSLA2).

Wu, M., X. Li, H. Wu, and C. Barrett. (2024). "Better Verified Explanations with Applications to Incorrectness and Out-of-Distribution Detection". URL: https://arxiv.org/abs/2409.03060.

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

Xu, C., D. Banerjee, D. Vasisht, and G. Singh. (2024). "Support is All You Need for Certified VAE Training". In: *The Thirteenth International Conference on Learning Representations*.

Xu, C. and G. Singh. (2024). "Cross-Input Certified Training for Universal Perturbations". In: *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXV*. Ed. by A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol. Vol. 15123. *Lecture Notes in Computer Science*. Springer. 233–250. DOI: 10.1007/978-3-031-73650-6_14.

Xu, K., Z. Shi, H. Zhang, Y. Wang, K.-W. Chang, M. Huang, B. Kailkhura, X. Lin, and C.-J. Hsieh. (2020). "Automatic perturbation analysis for scalable certified robustness and beyond". In: *Proc. Neural Information Processing Systems (NeurIPS)*. 1129–1141.

Xu, K., H. Zhang, S. Wang, Y. Wang, S. Jana, X. Lin, and C.-J. Hsieh. (2021). "Fast and Complete: Enabling Complete Neural Network Verification with Rapid and Massively Parallel Incomplete Verifiers". In: *International Conference on Learning Representations*.

Xue, X. and M. Sun. (2024). "Optimal Solution Guided Branching Strategy for Neural Network Branch and Bound Verification". In: *Engineering of Complex Computer Systems - 28th International Conference, ICECCS 2024, Limassol, Cyprus, June 19-21, 2024, Proceedings*. Vol. 14784. *Lecture Notes in Computer Science*. Springer. 67–87. DOI: 10.1007/978-3-031-66456-4_4.

Yang, C. and S. Chaudhuri. (2022). "Safe Neurosymbolic Learning with Differentiable Symbolic Execution". In: *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net. URL: https://openreview.net/forum?id=NYBmJN4MyZ.

Yang, C., D. Saxena, R. Dwivedula, K. Mahajan, S. Chaudhuri, and A. Akella. (2024a). "C3: Learning Congestion Controllers with Formal Certificates". *CoRR*. abs/2412.10915. DOI: 10.48550/ARXIV.2412.10915.

Yang, L., H. Dai, Z. Shi, C.-J. Hsieh, R. Tedrake, and H. Zhang. (2024b). "Lyapunov-stable neural control for state and output feedback: A novel formulation for efficient synthesis and verification". *arXiv preprint arXiv:2404.07956*.

Yang, P., R. Li, J. Li, C. Huang, J. Wang, J. Sun, B. Xue, and L. Zhang. (2021). "Improving Neural Network Verification through Spurious Region Guided Refinement". In: *Tools and Algorithms for the Construction and Analysis of Systems TACAS*. Vol. 12651. *Lecture Notes in Computer Science*. Springer. 389–408.

Yang, R., J. Laurel, S. Misailovic, and G. Singh. (2023). "Provable Defense Against Geometric Transformations". In: *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yang, Y. and M. C. Rinard. (2019). "Correctness Verification of Neural Networks". *CoRR*. abs/1906.01030. URL: http://arxiv.org/abs/1906.01030.

Yang, Z., K. Xu, B. Li, and H. Zhang. (2024c). "Improving Branching in Neural Network Verification with Bound Implication Graph". URL: https://openreview.net/forum?id=mMh4W72Hhe.

Yin, B., L. Chen, J. Liu, and J. Wang. (2022). "Efficient Complete Verification of Neural Networks via Layerwised Splitting and Refinement". *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* 41(11): 3898–3909. DOI: 10.1109/TCAD.2022.3197534.

Zelazny, T., H. Wu, C. Barrett, and G. Katz. (2022). "On Optimizing Back-Substitution Methods for Neural Network Verification". URL: https://arxiv.org/abs/2208.07669.

Zeng, Y., Z. Shi, M. Jin, F. Kang, L. Lyu, C.-J. Hsieh, and R. Jia. (2023). "Towards Robustness Certification Against Universal Perturbations". In: *The Eleventh International Conference on Learning Representations*. URL: https://openreview.net/forum?id=7GEvPKxjtt.

Zhang, B., T. Cai, Z. Lu, D. He, and L. Wang. (2021a). "Towards Certifying L-infinity Robustness using Neural Networks with L-inf-dist Neurons". In: *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*. Ed. by M. Meila and T. Zhang. Vol. 139. *Proceedings of Machine Learning Research*. PMLR. 12368–12379. URL: http://proceedings.mlr.press/v139/zhang21b.html.

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

Zhang, H., S. Wang, K. Xu, L. Li, B. Li, S. Jana, C.-J. Hsieh, and J. Z. Kolter. (2022). "General Cutting Planes for Bound-Propagation-Based Neural Network Verification". In: *Advances in Neural Information Processing Systems*. Ed. by A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho. URL: https://openreview.net/forum?id=5haAJAcofjc.

Zhang, H., T.-W. Weng, P.-Y. Chen, C.-J. Hsieh, and L. Daniel. (2018a). "Efficient Neural Network Robustness Certification with General Activation Functions". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Vol. 31. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2018/file/d04863f100d59b3eb688a11f95b0ae60-Paper.pdf.

Zhang, H., P. Zhang, and C. Hsieh. (2019). "RecurJac: An Efficient Recursive Algorithm for Bounding Jacobian Matrix of Neural Networks and Its Applications". In: *The 33rd AAAI Conference on Artificial Intelligence, (AAAI)*.

Zhang, X., A. Solar-Lezama, and R. Singh. (2018b). "Interpreting Neural Network Judgments via Minimal, Stable, and Symbolic Corrections". In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. 4879–4890. URL: https://proceedings.neurips.cc/paper/2018/hash/300891a62162b960cf02ce3827bb363c-Abstract.html.

Zhang, X., B. Wang, M. Kwiatkowska, and H. Zhang. (2024). "PREMAP: A Unifying PREiMage APproximation Framework for Neural Networks". URL: https://arxiv.org/abs/2408.09262.

Zhang, Y., A. Albarghouthi, and L. D'Antoni. (2021b). "Certified Robustness to Programmable Transformations in LSTMs". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Ed. by M. Moens, X. Huang, L. Specia, and S. W. Yih. Association for Computational Linguistics. 1068–1083.

Zhou, D., C. Brix, G. A. Hanasusanto, and H. Zhang. (2024). "Scalable Neural Network Verification with Branch-and-bound Inferred Cutting Planes". In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. URL: https://openreview.net/forum?id=FwhM1Zpyft.

Zhou, Z., Z. Huang, and S. Misailovic. (2023). "Aquasense: Automated sensitivity analysis of probabilistic programs via quantized inference". In: *International Symposium on Automated Technology for Verification and Analysis*. Springer. 288–301.