On the δ-decidability of decision problems for neural network questions

Olivier Bournez[1] Johanne Cohen[2] Valentin Dardilhac[3]

[1] LIX, Ecole Polytechnique, IP Paris, France
[2] Université Paris-Saclay, CNRS, LISN, 91405, Orsay, France.
[3] ENS Paris-Saclay, CNRS, LISN, Université Paris Saclay

July 26, 2023

A extended abstract with more details is available on:
https://www.lix.polytechnique.fr/~bournez/load/tmp/ccc-2023-more-details.pdf

The current success of deep learning applications renewed interest in various models of computation over the reals and their associated complexity classes. This phenomenon mirrors what happened during the second wave of neural networks, which led to the development and funding of several research projects related to computational models over the reals such as the Blum Shub Smale model [3, 6, 9]. Notably, the complexity class $\exists\mathbb{R}$, popularized in [12], corresponds to the constant-free Boolean part of the class $\mathrm{NP}_{\mathbb{R}}$ from [3, 9]. The currently popular real RAM model [4] can be viewed as an extension or formalization of models such as the real Turing machine model considered in [6, 9].

One reason for $\exists\mathbb{R}$'s popularity is its natural appearance in discussions about the complexity of decision questions for neural networks. Formally, let $\mathrm{Th}_{\exists}(\mathbb{R})$ be the set of all true sentences over $\mathbb{R}$ of the form $\exists x_1, \ldots, x_n \in \mathbb{R} : \phi(x_1, \ldots, x_n)$, where ϕ is a quantifier-free Boolean formula of equalities and inequalities of polynomials with integer coefficients. Then, $\exists\mathbb{R}$ is defined as the closure of $\mathrm{Th}_{\exists}(\mathbb{R})$ under polynomial-time many-one reductions [12]. Indeed, the training of neural networks has then been shown to be $\exists\mathbb{R}$ complete in [1]. Later, the authors of [2] proved the hardness of training for fully connected two-layer neural networks, even in the basic case of fully connected two-layer ReLU neural networks with exactly two input and output dimensions. In the context of verification of neural networks, it is established that reachability is NP-complete even for the simplest neural networks using the ReLU activation function [11]. This result has been further extended in [14] to prove that verification of neural networks is $\exists\mathbb{R}$ complete for every non-linear polynomial activation function.

However, in all these statements, completeness is established at the price of assuming that activation functions are either the $\mathrm{ReLU}(\mathrm{x}) = \max(0, \mathrm{x})$ function or, more generally, piecewise algebraic functions. This assumption allows them to demonstrate that the model can be effectively simulated by the real RAM model of computation [4, 3, 9]. Otherwise, it is not clear that we have $\exists\mathbb{R}$-membership as the function is not supported by the real RAM model of computation [4, 3, 9]. Notably, works like [10] state that computations involving analytic functions may be undecidable.

The question of what happens when the activation function is not piecewise algebraic, such as the widely used sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$, or tanh, is however very relevant. Let us define the complexity class $\exists\mathbb{R}_{\mathcal{F}}$ as the collection of decision problems that have a polynomial-time many-one reduction to closed formulae of existential real arithmetic with additional functions from $\mathcal{F}$. Here, $\mathcal{F}$ is a set of unary real-valued functions intended to capture the allowed activation functions. In that more general case, the neural network training problem $\mathrm{NN}_{\mathcal{F}}$-TRAINING is $\exists\mathbb{R}_{\mathcal{F}}$-complete, as proved in [7]. Specifically, NN_{σ}-TRAINING is $\exists\mathbb{R}_{\exp}$-complete [7]. However, the question of whether $\exists\mathbb{R}_{\exp}$ is decidable is a well-known open problem, initially posed by Tarksi, and is known to be related to some open questions in number theory [13]. In other words, while we have some completeness results, the hardness with respect to classical complexity, and even the decidability of neural network training or verification remain open questions.

Our first contribution is to present this unified framework that encompasses the previous results. This helps to better understand the connections between various models and complexity classes. A second main contribution is to demonstrate that the potential undecidability arises from the fact that exact tests are assumed in these models and classes. To address this, we consider a more robust notion of decision, called δ-decidability, proposed by [5]. We prove this leads to the definition of more natural and provably tractable robust complexity classes.

Namely, δ-decision corresponds to the following: Fix a collection $\mathcal{F}$ of computable functions over the reals. The associated logic might be undecidable, in the general case: there might be no algorithm that decides whether a first-order sentence φ over the corresponding signature is true. However, using arguments from computable analysis, the authors of [5] prove that there always exists an algorithm that, given any first-order sentence φ, containing only bounded quantifiers, and a positive rational number δ, decides either "φ is true", or "a δ-strengthening of φ is false". The proof from [5] consists in transforming the question into the question the sign of a computable function, expressed using functions from $\mathcal{F}$, and max and min functions. Then, using arguments from computable analysis, one can determine whether such a function is positive or negative, if we authorize an arbitrary answer (possibly incorrect) when close to 0.

In relation with this approach, we introduce the concept of the non-binary decision problem: the idea is that an algorithm may answer incorrectly for some inputs, i.e. may be incorrect in some "grey" zone. As an application, we establish that while the (exact) complexity of training neural networks remains open (in particular $\mathrm{NN}_{\tanh} - \mathrm{TRAINING}$), its (bounded) non-binary decision lies between NP and PSPACE. Our results follow from the previous arguments and an improved complexity analysis. The analysis of the approach in [5] is primarily based on results from Ko in [8] about the complexity of minimisation. We improve their analysis, focusing on the case of Lipschitz functions. This is the case of the tanh or the sigmoid σ functions, proving that the obtained algorithm is polynomial with oracle in NP.

In particular, as algorithms over neural networks are often implemented using non-exact arithmetic and bounded coefficients, we believe that the approach of δ-decision instead of decision is very natural and highly relevant. Fundamentally, our approach connects previous statements, primarily derived using the real RAM model, with essential questions and concepts in the realm of computable analysis.

References

[1] Mikkel Abrahamsen, Linda Kleist, and Tillmann Miltzow. Training neural networks is er-complete. *Advances in Neural Information Processing Systems*, 34:18293–18306, 2021.

[2] Daniel Bertschinger, Christoph Hertrich, Paul Jungeblut, Tillmann Miltzow, and Simon Weber. Training fully connected neural networks is∃ r-complete. 2022. *Preprint, https://doi. org/10.48550/arXiv*, 2204.

[3] Lenore Blum, Mike Shub, and Steve Smale. On a theory of computation and complexity over the real numbers; NP completeness, recursive functions and universal machines. *Bulletin of the American Mathematical Society*, 21(1):1–46, July 1989.

[4] Jeff Erickson, Ivor Van Der Hoog, and Tillmann Miltzow. Smoothing the gap between np and er. *SIAM Journal on Computing*, (0):FOCS20–102, 2022.

[5] Sicun Gao, Jeremy Avigad, and Edmund M Clarke. Delta-decidability over the reals. In *Logic in Computer Science (LICS), 2012 27th Annual IEEE Symposium on*, pages 305–314. IEEE, 2012.

[6] J. B. Goode. Accessible telephone directories. *The Journal of Symbolic Logic*, 59(1):92–105, March 1994.

[7] Teemu Hankala, Miika Hannula, Juha Kontinen, and Jonni Virtema. Complexity of neural network training and etr: Extensions with effectively continuous functions. *arXiv preprint arXiv:2305.11833*, 2023.

[8] Ker-I Ko. *Complexity Theory of Real Functions.* Progress in Theoretical Computer Science. Birkhaüser, Boston, 1991.

[9] Bruno Poizat. *Les petits cailloux: Une approche modèle-théorique de l'Algorithmie.* Aléas Editeur, 1995.

[10] Dan Richardson and John Fitch. The identity problem for elementary functions and constants. In *Proceedings of the international symposium on Symbolic and algebraic computation*, pages 285–290, 1994.

[11] Marco Sälzer and Martin Lange. Reachability is np-complete even for the simplest neural networks. In *International Conference on Reachability Problems*, pages 149–164, 2021.

[12] Marcus Schaefer and Daniel Štefankovič. Fixed points, nash equilibria, and the existential theory of the reals. *Theory of Computing Systems*, 60(2):172–193, 2017.

[13] Alex J Wilkie. Schanuel's conjecture and the decidability of the real exponential field. *NATO ASI Series C Mathematical and Physical Sciences-Advanced Study Institute*, 496:223–230, 1997.

[14] Adrian Wurm. Complexity of reachability problems in neural networks. *arXiv preprint arXiv:2306.05818*, 2023.