Deep Learning as a Convex Paradigm of Computation: Minimizing Circuit Size with ResNets

Arthur Jacot
Courant Institute, NYU
`arthur.jacot@nyu.edu`

November 27, 2025

Abstract

This paper argues that DNNs implement a computational Occam's razor - finding the 'simplest' algorithm that fits the data - and that this could explain their incredible and wide-ranging success over more traditional statistical methods. We start with the discovery that the set of real-valued function f that can be ϵ-approximated with a binary circuit of size at most $c\epsilon^{-\gamma}$ becomes convex in the 'Harder than Monte Carlo' (HTMC) regime, when $\gamma > 2$, allowing for the definition of a HTMC norm on functions. In parallel one can define a complexity measure on the parameters of a ResNets (a weighted ℓ_1 norm of the parameters), which induce a 'ResNet norm' on functions. The HTMC and ResNet norms can then be related by an almost matching sandwich bound. Thus minimizing this ResNet norm is equivalent to finding a circuit that fits the data with an almost minimal number of nodes (within a power of 2 of being optimal). ResNets thus appear as an alternative model for computation of real functions, better adapted to the HTMC regime and its convexity.

1 Introduction

Technically speaking, no statistical model can be strictly better than another, and each model can be optimal under certain assumptions. However, it has been argued that there is a statistical model that outmatches all others up to constants: finding the minimal Kolmogorov complexity function, i.e. the program with the minimal description length, that fits the data [42, 43]. Such an ideal statistical model would be at least as good as any model with short descriptions [20] which basically includes all models that have been described within a single scientific paper. But this perfect statistical model appears unattainable because we know that Kolmogorov complexity is undecidable in general, and even less optimizable.

In this paper, we prove results that suggest that Residual Networks (ResNets) training is implementing a weaker version of this ideal model: greedily minimizing circuit complexity in the so-called Harder than Monte Carlo (HTMC) regime. These simplifications can be summarized as follows:

- Minimal Circuit Size Problem (MCSP): searching for an interpolating circuit with minimal minimal number of operations is "only" NP. Minimal circuit size interpolators share a lot of the statistical power of minimal Kolmogorov complexity ones, in particular

their ability to learn hierarchical structures thanks to its compositionality. Roughly speaking, the tasks where minimal circuit size interpolators are suboptimal are those where the runtime of the interpolator is significantly larger than its description length, or in other terms its effective number of parameters.

- Real-valued function computation: DNNs are real-valued, so it makes sense to focus on the circuit complexity $C(f, \epsilon)$ of real-valued functions which is the smallest circuit size required to approximate f within an ϵ error. A central quantity is the rate $\gamma > 0$ of approximation (i.e. $C(f, \epsilon) = O(\epsilon^{-\gamma})$), which also determines the rate of the generalization error, e.g. for the MSE on N datapoints, one has $R_{test} = O(N^{-\frac{2}{2+\gamma}})$.

- Harder than Monte Carlo (HTMC) regime ($\gamma > 2$): A surprising discovery is that the set of functions that are approximable with a rate γ becomes convex when $\gamma > 2$, which we call the HTMC regime, thus allowing us to define a HTMC norm $\|f\|_{H^\gamma}$. This paper focuses on the minimization of this norm. Empirical computations of the test error rate of Large Language Models (LLMs) have identified rates such as $N^{-0.095}$ [29] which corresponds to a rate $\gamma \approx 18$ that is far into the HTMC regime.

1.1 Related Works

Universal Prior: The idea of searching for the interpolator with minimal Kolmogorov complexity is actually 5 years older [42, 41, 43] than Kolmogorov complexity itself [30]. However, the uncomputability of this task has led this idea to be considered more as a thought experiment than a concrete model [14, 20], although some computable relaxations have been proposed [45], which can be interpreted as optimizing for a notion of complexity that is somewhat closer to circuit complexity.

Minimum Circuit Size Problem (MCSP): The problem of minimizing circuit size has been known to be NP for a long time, but it remains unknown whether it is NP complete or not [18, 22]. The interest in the MCSP started in the 1950s in Russia as a typical example of a problem that requires brute force search [44], and later Leonid Levin actually wanted to prove NP-completeness of the MCSP in his seminal work on P=NP [31] but ultimately failed [32]. To this day, "no known algorithm significantly improves over brute-force search" [21]. The idea that there is a setting where MCSP becomes a convex problem, and that neural networks can then be used to (even greedily or approximately) solve it, is a stark departure from previous approaches.

Relating DNNs to Circuits: The analogy between DNNs and circuits is quite self-evident and has guided a lot of empirical research [16]. More formally, the space of functions representable by DNNs with a bounded width or number of non-zero parameters can be related to computational spaces, e.g. the space of compositionally sparse functions [38, 6, 40, 37, 13] which resemble computational graphs of Hölder functions where each function has low input dimension. The ability of these spaces to represent essentially any algorithm have been used to guarantee strong generalization bounds in other settings too [9, 35]. However, the number of non-zero parameters remains a discrete quantity that is not clearly optimizable, so this line of work has generally failed to capture the training dynamics.

Parameter norm minimization: Another source of inspiration is the literature DNNs whose parameters have bounded L_2 norm, for example as a result of L_2 regularization. This type of analysis has been very successful in simple models such as linear networks [12, 47], or in shallow networks [3, 2, 15, 39, 36], where norms on functions have been identified that describe exactly the space of functions that can be represented with a bounded L_2 norm of

2

the parameters. In these simpler networks, the convexity of those norms in function space translates into an absence of barriers between local and global minima in function spaces [1, 27], and more generally played an important role in the development of convergence proofs [11].

For deep fully-connected networks it is instead a notion of rank (on non-linear functions) which captures the space of representable functions [24, 23], at least at first order. My first conclusion from this line of work was that taking large depth limits gives compositionality at the cost of losing convexity, as illustrated by the resulting norm vs rank notions, which did not bode well for the possibility of proving convergence results. Thankfully, this present paper shows that (approximate) sub-additivity under both addition and composition is actually possible, though it requires switching to another notion of parameter norm.

Generalization Bounds: Looking for other notions of parameter complexity that avoid the pathologies of the L_2-norm, I have taken inspiration from the diverse literature on generalization bounds for DNNs, based on different notions of parameter which naturally avoid the issues of the L_2-norm.

A first line of work, generally based on Rademacher complexity, is based on taking products rather than sums of Frobenius norms over the layers [10, 34] or the related path norms [4, 15]. These can exhibit different degrees of approximate convexity, but they are generally sub-multiplicative rather than sub-additive under composition, thus leading to loose bounds on "deep" computational graphs.

Another line of work relies on covering number arguments to achieve some form of compositional sub-additivity [5, 46, 19], leading to almost tight rates on compositions of Sobolev functions [25]. A key idea is that the contribution of a connection to the parameter norm should be weighted by how much this connection affects the outputs, which is one of the key ideas that help obtain a much stronger approximate convexity in this paper.

Still, in practice these generalization bounds tend to be very loose [28] and it is unclear what makes them suboptimal. This paper started from the realization that these covering number arguments can be translated into circuit size upper bounds, and then providing an almost matching lower bound which could be key in understanding how to improve these types of bounds and compare them.

2   Setup and Main Results

The main result of this paper is a sandwich bound that relates the so-called HTMC norm $\|f\|_{H^\gamma}$ and ResNet (pseudo-)norm $\|f\|_{R^\omega}$, which measure the computability of f in with respect to two radically different notions of computation: the first being the traditional notion of "real-to-bin" computation where a binary circuit operates on a binary representation of the inputs, while the second proposes a new notion of computability based on being approximable by a ResNet f_θ with a small parameter complexity $R(\theta)$. The first is a discrete notion of computation while the second is a continuous one and our main result shows that these two paradigm can be almost unified, which implies that DNN optimization is almost equivalent to the MCSP.

2.1 Real-to-bin computation

We consider binary circuits C made up of binary AND, OR and NOT nodes. We write $|C|$ for the number of nodes, and we write

$$|C|_p = \min_{n_1,\ldots,n_L} \left(\sum_{\ell=1}^{L} n_\ell^p \right)^{\frac{1}{p}}$$

where the minimum is over all ways in which to organize the nodes of the circuit into L layers, each with $n_1,\ldots,n_L$ nodes (so that $|C| = \sum n_\ell = |C|_{p=1}$). We will be only interested in the case $p = \frac{2}{3}$, which can be interpreted as favoring parallel circuit that can be organized in only a few layers with many nodes per layer. We have the bounds $|C| \leq |C|_{\frac{2}{3}} \leq L^{\frac{1}{3}} |C|$, for L the optimal depth (since $(\sum n_\ell)^{\frac{2}{3}} \leq \sum n_\ell^{\frac{2}{3}}$ and by concavity $\frac{1}{L} \sum n_\ell^{\frac{2}{3}} \leq \left(\frac{1}{L} \sum n_\ell \right)^{\frac{2}{3}}$).

A real-valued function $f : \mathbb{R}^{d_{in}} \to \mathbb{R}^{d_{out}}$ cannot be computed exactly, but it can be approximated by a circuit which takes the binary representation of the inputs x and returns a binary representation of the outputs. Given a norm $\|\cdot\|$ on functions supported over a bounded hyper-rectangle (e.g. the $L_2(\pi)$ norm $\|\cdot\|_\pi$ over a distribution π or the L_∞ norm $\|\cdot\|_\infty$), we define

$$C_p(f,\epsilon; \|\cdot\|) = \min_{C : \|C-f\| \leq \epsilon} |C|_p$$

where the minimum is over all circuits that ϵ-approximate f. We will mainly focus on $C_{p=1}(f,\epsilon; L_2(\pi))$ which we will write as $C(f,\epsilon)$, but other cases will appear as well, such as $C_{p=\frac{2}{3}}(f,\epsilon; L_\infty)$.

We can classify computable real-valued functions in terms of their rate γ and prefactor c of computability: $C_p(f,\epsilon; \|\cdot\|) \leq c\epsilon^{-\gamma}$. For any γ, we define

$$\|f\|_{H_p^\gamma(\|\cdot\|)}^\gamma = \max_\epsilon \epsilon^\gamma C_p(f,\epsilon; \|\cdot\|)$$

which is the smallest value such that $C_p(f,\epsilon; \|\cdot\|) \leq \|f\|_{H_p^\gamma(\|\cdot\|)}^\gamma \epsilon^{-\gamma}$. For simplicity, we will denote H^γ for $H_{p=1}^\gamma(L_2(\pi))$.

The homogeneity of $\|f\|_{H_p^\gamma(\|\cdot\|)}$ follows directly from its definition, however it is not subadditive in general, thus failing to be a norm. Surprisingly, when $\gamma > 2$, which we call the the Harder than Monte Carlo (HTMC) regime, $\|f\|_{H^\gamma}$ becomes convex up to a constant:

Theorem 1. *For $\gamma > 2$, there is a constant c_γ such that for all $m \geq 1$, $\|\sum_{i=1}^m f_i\|_{H^\gamma} \leq c_\gamma \sum_{i=1}^m \|f_i\|_{H^\gamma}$.*

Sketch of proof. This follows quite directly from Multi-level Monte Carlo (MLMC) [17]. See Theorem 4 in Section 3.1 for the complete proof. □

This does not quite make H^γ a norm, but there is a norm $Conv(H^\gamma)$ (obtained by replacing the unit ball of H^γ by its convex hull) such that $c_\gamma^{-1} \|f\|_{H^\gamma} \leq \|f\|_{Conv(H^\gamma)} \leq \|f\|_{H^\gamma}$, thanks to the fact that the constant c_γ does not depend on m.

The convexity of the HTMC norm suggests that one could use convex optimization methods to minimize circuit size, but it remains unclear how to optimize over the infinite sequence of circuits $A_1, A_2, \ldots$ that underlies the definition of the HTMC norm. The issue is that the usual notion of real-to-bin computation is fundamentally discrete, and thus ill adapted to leverage the convex/continuous structure of computation in the HTMC regime.

This could be solved by switching to a "fully real" notion of computation: computational graphs with nodes that take real values and apply continuous operations rather than binary ones, and with connections between nodes that take any value instead of being 0 or 1 only. This pretty much describes a neural network, and so we turn our attention to ResNets and how they can be connected to the HTMC norm.

2.2 ResNet computation

For an input $x \in \mathbb{R}^{d_{in}}$, we define the activations $\alpha_0(x), \ldots, \alpha_L(x) \in \mathbb{R}^{d+1}$ of a Residual Network (ResNet) of depth L by first fixing their last coordinate to $\alpha_{\ell,d+1}(x) = 1$ (this will allow us to combine the usual weight matrix and bias vector into a single weight matrix) and then defining the first d coordinates $\alpha_{\ell,1:d}(x)$ recursively:

$$\alpha_{0,1:d}(x) = W_{in}x$$
$$\alpha_{\ell,1:d}(x) = \alpha_{\ell-1,1:d}(x) + W_\ell \sigma \left(V_\ell \alpha_{\ell-1}(x) \right),$$

where $\sigma(x) = \max x, 0$ is the ReLU, and the matrices W_ℓ, V_ℓ are of dimensions $d \times w$ and $w \times d+1$ respectively. We call w the width of the network and d the hidden dimension. Finally, the outputs are defined as $f_\theta(x) = W_{out}\alpha_L(x)$, where θ is the vector of parameters, obtained by concatenating the entries of W_{in}, W_{out} and all W_ℓ, V_ℓs. We will focus on inputs contained within a hyper-rectangle with sides $s_1, \ldots, s_{d_{in}}$ which we combine into a $d_{in} \times d_{in}$ diagonal matrix S.

Our strategy is to define a notion of parameter complexity $R(\theta)$ that describes how "prunable" the network f_θ in the sense that one can remove all but $O(R(\theta)^2\epsilon^{-2})$ connections of the network without changing the outputs by more than an ϵ. This in turns implies that the function f_θ can be approximated by a circuit of size $O(R(\theta)^2\epsilon^{-2})$ (up to $\log \epsilon^{-1}$ terms) and therefore implies a bound of the type "$\|f_\theta\|_{H^{\gamma=2}} \lesssim R(\theta)$".

Before we define $R(\theta)$, we give a simple heuristic argument that leads to another $R_{lin}(\theta)$ that is both simpler and more easily interpretable. Sadly, to obtain rigorous results we need to use the more complex $R(\theta)$. The intuition goes as follows: if we prune each parameter θ_i randomly, by multiplying it with independent rescaled Poisson random variables $\tilde{\theta}_i = \theta_i \frac{P_i}{\lambda_i}$ where $P_i \sim Poisson(\lambda_i)$, then by a simple Taylor approximation, we obtain

$$\mathbb{E} \left\| f_\theta - f_{\tilde{\theta}} \right\|_\pi^2 \approx \sum \lambda_i^{-1} \theta_i^2 \left\| \partial_{\theta_i} f_\theta \right\|_\pi^2$$

with an expected number of non-zero parameters of $\mathbb{E} \left\| \tilde{\theta} \right\|_0 = \sum 1 - e^{-\lambda_i} \leq \sum \lambda_i$. With the optimal choice λ_is, one obtains an error ϵ with $\mathbb{E} \|\theta\|_0 \leq R_{lin}(\theta)^2\epsilon^{-2}$ for

$$R_{lin}(\theta) = \sum |\theta_i| \left\| \partial_{\theta_i} f_\theta \right\|_\pi.$$

But to control the other terms in the Taylor expansion, we will use the following complexity measure:

$$R(\theta)^{\frac{2}{3}} = \min_{D_\ell} \sum_{\ell=1}^{L} \|D_\ell |W_\ell| |V_\ell| C_{\ell-1}\|_1^{\frac{2}{3}} + \|D_0 W_{in}S\|_1^{\frac{2}{3}} + \|W_{out}C_L\|_1^{\frac{2}{3}}$$

where C_ℓ is diagonal with $C_{\ell,ii} = \|\alpha_{\ell,i}\|_\infty$, and the minimum is over all diagonal D_ℓs such that $Lip((\alpha_\ell \to f_\theta) \circ D_\ell^{-1}) \leq 1$ (i.e. the map from the ℓ-th layer activations α_ℓ to the

5

outputs precomposed with D_ℓ^{-1} is 1-Lipschitz), and we apply the absolute value entrywise to the matrices W_ℓ, V_ℓ.

One can verify that $2R(\theta)$ is an upper bound on the 'ideal' parameter complexity $R_{lin}(\theta)$. Note that both $R(\theta)$ and $R_{lin}(\theta)$ are weighted ℓ_1 norms of the parameters, but since the weights depend on the parameter themselves, $R(\theta)$ and $R'(\theta)$ are generally not convex in parameter space, hence why we call them 'complexities' and do not use a norm notation.

Convexity does reveal itself in the function space, mirroring the HTMC convexity. We define the ω-ResNet norm $\|f\|_{R^\omega}$ that captures functions f that can be ϵ-approximated by a ResNet with a complexity measure $R(\theta)$ of order $\epsilon^{1-\omega}$:

$$\|f\|_{R^\omega(\|\cdot\|)}^\omega = \max_\epsilon \epsilon^{\omega-1} \min_{\theta:\|f_\theta-f\|\leq\epsilon} R(\theta),$$

which implies that for all ϵ, there are parameters θ_ϵ such that $\|f_{\theta_\epsilon} - f\| \leq \epsilon$ and $R(\theta_\epsilon) \leq \|f\|_{R^\omega(\|\cdot\|)}^\omega \epsilon^{1-\omega}$. The choice $\omega = 1$ corresponds to functions that can be represented exactly with a finite complexity measure $R(\theta)$, and we will sometimes use the notation $\|\cdot\|_R = \|\cdot\|_{R^{\omega=1}}$. One can easily check that $\|f\|_{R_{lin}^\omega}$ is convex because it is subadditive under putting two networks in parallel, while $\|f\|_{R^\omega}$ only satisfies

$$\|f + g\|_{R^\omega} \leq \left(\|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} + \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} \right)^{\frac{1+2\omega}{2\omega}}.$$

One of the goal of this paper is to prove that the circuit size required to approximate a ResNet can be bounded in terms of the parameter complexity measure $R(\theta)$ independently from the size of the network. For this reason, we will generally assume that the depth L, width w, and hidden dimension d are arbitrarily large, and obtain results that are independent of these quantities (as long as they are large enough). For example, in the definition of the R^ω norm, the minimization is over all possible network sizes and parameters.

2.3   Sandwich bound

We are now able to state our main result:

Theorem 2. *For all and $\omega > 1$ and $\delta > 0$, one has (up to constants that are polynomial in the input and output dimensions):*

$$\|f\|_{H^{\gamma=2\omega+\delta}} + \|f\|_{C^{\alpha=\frac{1}{\omega}}} \lesssim \|f\|_{R^\omega} \lesssim \|f\|_{H_{p=\frac{2}{3}}^{\gamma=\omega-\delta}(L_\infty)} + \|f\|_{C^{\alpha=\frac{1}{\omega-\delta}}},$$

where $\|\cdot\|_{C^\alpha}$ is the Hölder (semi-)norm for $\alpha \in (0,1]$, and where $\lesssim$ denotes an inequality up to a constant that depends $d_{in}, d_{out}, \gamma, \alpha, \delta$.

Structure of the proof. The lower bound is a pruning bound, where the majority of the weights of the ResNet are removed to obtain a small circuit, and the upper bound is a construction bound, where an infinite sequence of circuits that approximate a function f are combined into a single ResNet. The LHS follows from Theorem 11 from Section 4.2 and point 2 of Proposition 10 from Section 4.1. The RHS follows from Theorem 14 in Section 4.3. □

The convexity of the HTMC regime hints at the potential to use of convex optimization to solve the MCSP, but it is unclear how one could differentiate the HTMC norm because it is

defined in terms of a discrete structure: infinite sequences of binary circuits. In contrast the linearized complexity measure $R_{lin}(\theta)$ is both differentiable and convex in function space, which could be leveraged to prove global convergence of gradient descent (in the infinite width limit). The $R(\theta)$ norm is less practically differentiable because of its reliance on Lipschitz constants and L_∞-norms off the activations, it is also not exactly convex (though it could easily be "convexified" by taking ensembles of ResNets). A future goal is to find a differentiable, convex complexity measure that admits a similar sandwich bound, and use it to prove global convergence of gradient descent to an approximate solution of the MCSP, opening the door to a whole new family of convex algorithms for this fundamental problem.

The quality of this approximate solution directly depends on the tightness of the sandwich bound. The δ reflects the presence of logarithmic terms, some of which could be removed, and the $p = \frac{2}{3}$ and L_∞ norm on the RHS are likely artifacts of the current proof and/or of the definition of $R(\theta)$. There are however two distinctions that might be representative of something more fundamental:

Hölder continuity: The presence of the Hölder-continuity norm on both sides implies that the R-norm captures a mix of computability and regularity. This is interesting, because especially in high-dimension, these two notions are essentially "orthogonal": there are computable functions, such as the heavy-side, that are γ-computable for all $\gamma > 0$ but not Hölder, and while all Hölder functions are computable, the worst-case and typical rate suffers from the curse of dimensionality because the number of equal frequency Fourier modes is exponential in d_{in} (computing only the lowest Fourier frequencies, one obtains $\|f\|_{H^{\gamma = \frac{d_{in}}{\alpha}}} \lesssim \|f\|_{C^\alpha}$). The HTMC unit ball is also much "smaller" than the Hölder unit ball in high dimension, as illustrated by the fact that uniform generalization bound over these balls are of order $N^{-\frac{2}{2+\gamma}}$ and $N^{-\frac{2\alpha}{2\alpha+d_{in}}}$ respectively for the Mean squared Error (MSE), where only the second one suffers from the curse of dimensionality.

HTMC rate gap: Up to δ terms, the HTMC rate of the LFS is $\gamma = 2\omega$ while the RHS is $\gamma = \omega$, which implies that if one translates a circuit into a ResNet and back into a circuit, the number nodes can be squared in the worst case. This also implies that if one minimizes $R(\theta)$ as a proxy for the circuit size, one can only guarantee recovery of a circuit whose size is at most the square of the optimal one. A possible explanation for this gap is that while the Hölder continuity guarantees regularity of the map from input to output, the complexity measure $R(\theta)$ also controls the regularity of the intermediate steps in the middle of the network. It is possible that this worst-case gap only applies to functions whose optimal circuit approximation has intermediate representations that are in some sense highly irregular (e.g. the approximating circuits for different ϵ are completely different, so that the intermediate binary values cannot be interpreted as converging to an intermediate real-valued representation as $\epsilon \searrow 0$). In fact if we restrict ourselves to compositions $f = g_L \circ \cdots \circ g_1$ of Sobolev and Lipschitz functions $g_1, \ldots, g_L$ (or more general computational graphs of Sobolev functions, similar to notion the compositional sparsity [38]), then we can guarantee a rate of $\gamma = \max_\ell \frac{d_\ell + 3}{k_\ell}$, where k_ℓ is the differentiability of g_ℓ and d_ℓ its input dimension, which is much closer to the optimal $\gamma^* = \max_\ell \frac{d_\ell}{k_\ell}$ than $2\gamma^*$ (this follows from Theorem 5 in [25], together with the fact that the complexity of this paper upper bounds the one of this paper up to constants in the intermediate dimensions).

3 Harder than Monte Carlo (HTMC) Regime

This section delves deeper into the HTMC regime and related HTMC norm. We first prove a few basic properties in Section 3.1, in particular the convexity of the HTMC norm. We then prove a probably approximately correct (PAC) generalization bound based on the HTMC norm in Section 3.2. Finally in Section 3.3 we introduce a family of functions, the Tetrakis functions, which we interpret as being an approximation for the vertices or extrema points of the HTMC unit ball. These basic properties and Tetrakis functions which will play a central role in Section 4 where we will prove Theorem 2.

3.1 Basic Properties

It follows directly from the definition of the HTMC norm that it is homogeneous and bounds the underlying norm $\|\cdot\|$:

Proposition 3. *We have:*

1. $\|\lambda f\|_{H_p^\gamma(\|\cdot\|)} = \lambda \|f\|_{H_p^\gamma(\|\cdot\|)}$,

2. $\|f\| \leq \|f\|_{H_p^\gamma(\|\cdot\|)}$.

Proof. (1) If the algorithm A approximates f within an ϵ, then λA approximates λf within an $\lambda\epsilon$ error and therefore

$$C_p(\lambda f, \lambda\epsilon, \|\cdot\|) = C_p(f, \epsilon, \|\cdot\|).$$

In our setting, A and λA have the same size because we are only changing the decoding step, i.e. how the final binary representation gets mapped to a real value, which is not counted in the size.

We then have

$$\begin{aligned}
\|\lambda f\|_{H_p^\gamma(\|\cdot\|)}^\gamma &= \min_\epsilon \epsilon^\gamma C_p(\lambda f, \epsilon; \|\cdot\|) \\
&= \min_\epsilon (\lambda\epsilon)^\gamma C_p(\lambda f, \lambda\epsilon; \|\cdot\|) \\
&= \lambda^\gamma \min_\epsilon C_p(f, \epsilon; \|\cdot\|) \\
&= \lambda^\gamma \|f\|_{H_p^\gamma(\|\cdot\|)}^\gamma.
\end{aligned}$$

(2) For all $\epsilon > \|f\|_{H_p^\gamma(\|\cdot\|)}$, there is a circuit that ϵ-approximates f with a circuit size of zero, because $\|f\|_{H_p^\gamma(\|\cdot\|)}^\gamma \epsilon^{-\gamma} < 1$. Since the only circuit with zero nodes is the constant 0 function, we obtain our bound. $\qquad\square$

However, subadditivity fails in general, because if we approximate $f + g$ by the sum of the approximations of f and g, we obtain (disregarding the cost of the final summation which is of order $\log \epsilon$) for all $p \geq 1$

$$C_p(f + g, \epsilon; \|\cdot\|) \leq \min_{\epsilon=\epsilon_1+\epsilon_2} C_p(f, \epsilon_1; \|\cdot\|) + C_p(g, \epsilon_2; \|\cdot\|).$$

Optimizing over ϵ_1 and ϵ_2, we only obtain a weaker notion of subadditivity: $\|f + g\|_{H_p^\gamma(\|\cdot\|)}^{\frac{\gamma}{\gamma+1}} \leq \|f\|_{H_p^\gamma(\|\cdot\|)}^{\frac{\gamma}{\gamma+1}} + \|g\|_{H_p^\gamma(\|\cdot\|)}^{\frac{\gamma}{\gamma+1}}$, and the case $p < 1$ is even worse.

But we can do better than this naive bound, by leveraging the fact that we have access to multiple approximations of f across any accuracy ϵ, and by using a Multi Level Monte Carlo (MLMC) we can prove the convexity of H^γ up to a universal constant:

Theorem 4. *For all $\gamma > 2$, there is a constant c_γ such that for all $m \geq 1$,*

$$\left\| \sum_{i=1}^m f \right\|_{H^\gamma} \leq c_\gamma \sum_{i=1}^m \|f\|_{H^\gamma}.$$

Proof. The proof relies on the probabilistic method, we define a random approximator and bound its expected squared error, which implies the existence of an approximator.

Thanks to the homogeneity of the HTMC norm, we may assume that $\sum_{j=1}^m \|f_j\|_{H^\gamma} = 1$ and write $p_j = \|f_j\|_{H^\gamma}$. For an integer k we sample n_k iid random functions $f_{k,1}, \ldots, f_{k,n_k}$, each equal to $p_j^{-1} f_j$ with probability p_j so that $\mathbb{E}[f_{k,i}] = \sum_{j=1}^m f_j$ and $\|f_{k,i}\|_{H^\gamma} = 1$ almost surely. We then consider the pairs of circuits $A_{k,i}, B_{k,i}$ that are 2^{-k} and 2^{-k+1} approximators of $f_{k,i}$, so that $\|A_{k,i} - B_{k,i}\|_\pi \leq \|A_{k,i} - f_{k,i}\|_\pi + \|B_{k,i} - f_{k,i}\|_\pi \leq 3 \cdot 2^{-k}$ and $A_{k,i}, B_{k,i}$ have circuit size at most $2^{\gamma k}$ and $2^{\gamma(k-1)}$ respectively. We then approximate $f = \sum_{j=1}^m f_j$ using MLMC

$$\tilde{f} = \sum_{k=0}^{k_{max}} \frac{1}{n_k} \sum_{i=1}^{n_k} A_{k,i} - B_{k,i}.$$

Note that we may assume that $B_{0,i} = 0$ since it approximates $f_{0,i}$ within $2^{-0} = 1$ because $\|f\|_\pi \leq \|f\|_{H^\gamma} = 1$. We have

$$\left\| \mathbb{E}\tilde{f} - f \right\|_\pi = \|\mathbb{E}A_{k_{max},i} - f\|_\pi \leq 2^{-k_{max}}$$

$$\mathbb{E}\left\| \tilde{f} - \mathbb{E}\tilde{f} \right\|_\pi^2 \leq \sum_{k=0}^{k_{max}} \frac{1}{n_k^2} \sum_{i=1}^{n_k} \|A_{k,i} - B_{k,i}\|_\pi^2 \leq \sum_{k=0}^{k_{max}} \frac{3}{n_k} 2^{-2k}.$$

And since the expected circuit size of $A_{k,i}$ and $B_{k,i}$ are at most $2^{\gamma k}$ and $2^{\gamma(k-1)}$ respectively, the expected circuit size is at most

$$10 \sum_{k=0}^{k_{max}} |A_{k,i}| + |B_{k,i}| \leq 10(1 + 2^{-\gamma}) \sum_{k=0}^{k_{max}} n_k 2^{\gamma k}$$

where the prefactor of 10 allows us to also capture the cost of adding up the circuits, because we need to perform at most one sum/substraction per $A_{k,i}/B_{k,i}$ and the number of significant bits that need to be summed is bounded by the circuit size of $A_{k,i}$ and $B_{k,i}$. Since a 'full adder' requires two XOR (which themselves require two AND and one OR), two AND and one OR, we need $c = 1 + 2 \cdot 3 + 2 + 1 = 10$. this is obviously very loose in general.

For any $\epsilon > 0$, we choose $k_{max} = \lceil -\log_2 \epsilon/\sqrt{2} \rceil$ and $n_k = \left\lceil \frac{3}{1 - 2^{-\frac{\gamma-2}{2}}} (\epsilon/\sqrt{2})^{-\frac{\gamma+2}{2}} 2^{-\frac{\gamma+2}{2}k} \right\rceil$

to obtain $\mathbb{E}\left\|\tilde{f} - f\right\|_\pi^2 \le \epsilon^2$ at a computational cost of at most

$$10(1 + 2^{-\gamma}) \sum_{k=0}^{k_{max}} \frac{3}{1 - 2^{-\frac{\gamma-2}{2}}} (\epsilon/\sqrt{2})^{-\frac{\gamma+2}{2}} 2^{\frac{\gamma-2}{2}k} + 2^{\gamma k}$$

$$\le 10(1 + 2^{-\gamma}) \left[\frac{3 \cdot 2^{\frac{\gamma+2}{4}}}{\left(1 - 2^{-\frac{\gamma-2}{2}}\right)^2} + \frac{1}{1 - 2^{-\gamma}} \right] \epsilon^{-\gamma}.$$

$\square$

The behavior of the HTMC norm under composition is close but weaker than subadditivity. For functions $f_1, \ldots, f_L$ and their ϵ_ℓ-approximating circuits $A_1, \ldots, A_L$ we can approximate $f_{L:1}$ by $A_{L:1}$ and a telescopic sum argument gives us

$$C(f_l \circ \cdots \circ f_1, \sum_\ell Lip(f_{L:\ell+1})\epsilon_\ell) \le \sum_{\ell=1}^L C(f_\ell, \epsilon_\ell; L_2(A_{\ell-1:1}\#\pi)), \tag{1}$$

where $A_{\ell-1:1}\#\pi$ is the pushforward of π under $A_{\ell-1:1}$, i.e. the distribution of $A_{\ell-1:1}(x)$ when $x \sim \pi$. Two things stand out: first we need to control the Lipschitz constants of $f_{L:\ell+1}$ to control how the approximation error in the intermediate steps propagates to the outputs, and second the approximations are made on the distribution $A_{\ell-1:1}\#\pi$ rather than $f_{\ell-1:1}\#\pi$. To handle the second problem, one could leverage the fact that $A_{\ell-1:1}$ is an approximation of $f_{\ell-1:1}$ and therefore $f_{\ell-1:1}\#\pi$ and $A_{\ell-1:1}\#\pi$ are close in Wasserstein distance. We will use a simpler strategy in this paper: our bound on $C(f_\ell, \epsilon_\ell; L_2(\pi_\ell))$ only depends on the constants c_i such that $|x_i| \le c_i$ over the support of π_ℓ, we therefore can simply project the i-th output of $A_{\ell-1:1}$ to the range $[-c_i, c_i]$. This idea is formalized in the following statement:

Proposition 5. *Let $f_1, \ldots, f_L$ be functions $f_\ell : \mathbb{R}^{d_{\ell-1}} \to \mathbb{R}^{d_\ell}$ such that $|f_{\ell:1,i}(x)| \le c_{\ell,i}$ over the support of π, and assume $\|f_\ell\|_{H^\gamma(\pi_{\ell-1})} \le r_\ell$ for all distributions $\pi_{\ell-1}$ supported in the hyper-rectangle $[-c_{\ell-1,1}, c_{\ell-1,1}] \times \cdots \times [-c_{\ell-1,d_\ell}, c_{\ell-1,d_\ell}]$, then*

$$\|f_{L:1}\|_{H^\gamma(\pi)}^{\frac{\gamma}{\gamma+1}} \le c^{\frac{1}{\gamma+1}} \sum_{\ell=1}^L (Lip(f_{L:\ell+1})r_\ell)^{\frac{\gamma}{\gamma+1}}.$$

Proof. Between each approximation A_ℓ of f_ℓ, we add a projection step, that clamps each coordinate to the range $[-c_{\ell,i}, c_{\ell,i}]$. This only reduces the approximation error (since the output f_ℓ is by assumption already inside this range), and increases the circuit size by at most a constant prefactor since at each layer a finite number of bits of the representations are changed, and only those need to be projected. Combining Equation 1 together with our assumption on the HTMC norm of f_ℓ, we obtain

$$C(f_l \circ \cdots \circ f_1, \sum_\ell Lip(f_{L:\ell+1})\epsilon_\ell) \le c \sum_{\ell=1}^L r_\ell^\gamma \epsilon_\ell^{-\gamma}.$$

Optimizing over the choice of ϵ_ℓ that result in the same total error ϵ, we obtain

$$C(f_l \circ \cdots \circ f_1, \epsilon) \le c \left(\sum_{\ell=1}^L Lip(f_{L:\ell+1})^{\frac{\gamma}{\gamma+1}} r_\ell^{\frac{\gamma}{\gamma+1}} \right)^{\gamma+1} \epsilon^{-\gamma}$$

10

which implies $\|f_{L:1}\|_{H^\gamma(\pi)}^{\frac{\gamma}{\gamma+1}} \le c^{\frac{1}{\gamma+1}} \sum_{\ell=1}^{L} (Lip(f_{L:\ell+1})r_\ell)^{\frac{\gamma}{\gamma+1}}$ as needed. $\qquad\square$

3.2 Generalization bound

One can rely on the HTMC norm to prove a probably approximately correct (PAC) generalization bound for the (noiseless) MSE loss, i.e. bound the gap between the risk

$$R(f) = \mathbb{E}_{X \sim \pi}[\|f(x) - f^*(x)\|^2] = \|f - f^*\|_\pi^2,$$

and empirical risk

$$\tilde{R}_N(f) = \frac{1}{N} \sum_{i=1}^{N} \|f(x-i) - f^*(x_i)\|^2] = \|f - f^*\|_{\tilde{\pi}_N}^2,$$

for the 'true function' f^*, the 'true distribution' π, and the empirical measure $\tilde{\pi}_N$ over the dataset $x_1, \ldots, x_N$.

Theorem 6. *Given a true function f^* that is bounded $\|f^*\|_\infty \le B$, then for any function f with bounded L_∞ norm $\|f\|_\infty \le B$ and bounded γ-HTMC norm $\|f\|_{H^\gamma_{p=1}(L_2(\frac{\pi+\tilde{\pi}_N}{2}))}$ over the mixed distribution $\frac{\pi+\tilde{\pi}_N}{2}$, we have that with probability p*

$$\sqrt{R(f)} - \sqrt{\tilde{R}_N(f)} \le 4 \|f\|_{H^\gamma} \left(\frac{B^2}{\|f\|_{H^\gamma}^2 N} \right)^{\frac{1}{2+\gamma}} \left(1 + \sqrt{4 \log \left(2 \frac{\|f\|_{H^\gamma}^2}{B^2} N \right)} \right) + 3B \sqrt{\frac{\log \frac{1}{p}}{N}}.$$

Proof. Let us first consider a generic circuit A such that $\|A\|_\infty \le B$. Since the function $h(x) = \|f^*(x) - A(x)\|^2$ is bounded by $2B^2$, we may apply Bernstein inequality to obtain that with probability $1 - p$

$$R(A) - \tilde{R}_N(A) \le \sqrt{\mathrm{Var}(h(x_i)) \frac{-2 \log p}{N}} + B^2 \frac{-4 \log p}{3N} \le \sqrt{R(A) B^2 \frac{-2 \log p}{N}} + B^2 \frac{-4 \log p}{3N}$$

where we used $\mathrm{Var}(h(x_i)) \le \mathbb{E}_\pi [h(x)^2] \le 2B^2 \mathbb{E}_\pi [h(x)] = 2B^2 R(A)$. Using the quadratic formula, we obtain

$$\sqrt{R(A)} \le \sqrt{B^2 \frac{-\log p}{2N}} + \sqrt{B^2 \frac{-\log p}{2N} + \tilde{R}_N(A) + B^2 \frac{-4 \log p}{3N}} \le \sqrt{\tilde{R}_N(A)} + 3B \sqrt{\frac{\log \frac{1}{p}}{N}},$$

where we used $\frac{1}{\sqrt{2}} + \sqrt{\frac{3+8}{6}} \approx 2.0611 \le 3$.

Since there are at most $(3T^2)^T$ circuits with T nodes (three types of nodes each connected to at most 2 other nodes, of which there are T), we can take a union bound over all such circuits and obtain that with probability $1 - p$ we have simultaneously for all circuits of size T

$$\sqrt{R(A)} - \sqrt{\tilde{R}_N(A)} \le 3B \sqrt{\frac{T \log 3T^2 + \log \frac{1}{p}}{N}} \le 3B \sqrt{\frac{T \log 3T^2}{N}} + 3B \sqrt{\frac{\log \frac{1}{p}}{N}}.$$

Given a function f with finite HTMC norm $\|f\|_{H^\gamma}$, there is for all ϵ a circuit A_ϵ that ϵ-approximates it over the sample S over the mixed distribution $\frac{\pi+\tilde{\pi}_N}{2}$, where $\tilde{\pi}_N$ is the

11

empirical measure over the training data. We therefore have $\sqrt{R(f)} - \sqrt{\tilde{R}_N(f)} \leq \sqrt{R(A_\epsilon)} - \sqrt{\tilde{R}_N(A_\epsilon)} + 4\epsilon$ so that with probability $1 - p$:

$$\sqrt{R(f)} - \sqrt{\tilde{R}_N(f)} \leq \min_\epsilon 4\epsilon + 3B\sqrt{\frac{\|f\|_{H^\gamma}^\gamma \, \epsilon^{-\gamma} \log(3\|f\|_{H^\gamma}^{2\gamma} \, \epsilon^{-2\gamma})}{N}} + 3B\sqrt{\frac{\log \frac{1}{p}}{N}}.$$

and with $\epsilon = \left(B \frac{\|f\|_{H^\gamma}^{\frac{\gamma}{2}}}{\sqrt{N}} \right)^{\frac{2}{2+\gamma}}$, we obtain that $\sqrt{R(f)} - \sqrt{\tilde{R}_N(f)}$ is upper bounded by

$$4B^{\frac{2}{2+\gamma}} \left(\frac{\|f\|_{H^\gamma}^\gamma}{N} \right)^{\frac{1}{2+\gamma}} \left(1 + \sqrt{\log\left(3 \left(\frac{\|f\|_{H^\gamma}}{B} \right)^{\frac{4\gamma}{2+\gamma}} N^{\frac{2\gamma}{2+\gamma}} \right)} \right) + 3B\sqrt{\frac{\log \frac{1}{p}}{N}}$$

$$\leq 4\|f\|_{H^\gamma} \left(\frac{B^2}{\|f\|_{H^\gamma}^2 \, N} \right)^{\frac{1}{2+\gamma}} \left(1 + \sqrt{2\log\left(3 \frac{\|f\|_{H^\gamma}^2}{B^2} N \right)} \right) + 3B\sqrt{\frac{\log \frac{1}{p}}{N}}.$$

$\square$

Note that if we assume that the train error is zero (or small enough), this implies a bound on the risk $R(f)$ of order $\|f\|_{H^\gamma} \left(\frac{B^2}{\|f\|_{H^\gamma}^2 N} \right)^{\frac{2}{2+\gamma}}$ (up to log terms). The LHS of the sandwich bound (Theorem 11) allows us to the HTMC norm in terms of th parameter complexity $R(\theta)$, and since this upper bound only depends on the support of the input distribution, it can easily be applied to the mixed distribution $\frac{\pi + \tilde{\pi}_N}{2}$.

The proof is analogous to bounds based on covering number arguments [7, 25], where the circuits play the role of the center of the balls of the covering. However the big advantage is that the set of circuits with bounded size is independent of the empirical measure $\hat{\pi}_N$, which allows us to apply Bennett's inequality (a type of Chernoff bound) directly, whereas with covering numbers (or Rademacher complexity) a symmetrization argument is required.

Actually, for many models (described by a family $\mathcal{F}$ of functions) the ϵ-covering number $\mathcal{N}(\mathcal{F}, \epsilon)$ is upper bounded (up to log terms) by the max circuit size $\max_{f \in \mathcal{F}} C(f, \epsilon)$. And for many models, e.g. Sobolev functions, this bound is tight, which means that one would recover the same rate with a covering number argument or with a circuit size argument.

One could go even further and consider the ϵ-Kolmogorov complexity $K(f, \epsilon)$ and define a similar Kolmogorov HTMC norm $\|f\|_{K^\gamma}$. A similar generalization bound could then be provenand it would yield rates that match covering arguments on all finite description length models.

3.3 Vertices of the HTMC ball: Tetrakis functions

Taking a more convex geometric point of view, the previous section proves that the HTMC unit ball B_{H^γ} is in some sense pretty small, because we can efficiently apply a uniform bound over it. This could be interpreted as the HTMC ball being 'pointy' like the ℓ_1 ball rather than 'boxy' like the ℓ_∞ ball.

To make this a little bit more formal, we can try to identify the vertices or extrema points of the HTMC ball, i.e. the smallest subset $V \subset B_{H^\gamma}$ whose convex hull equals the
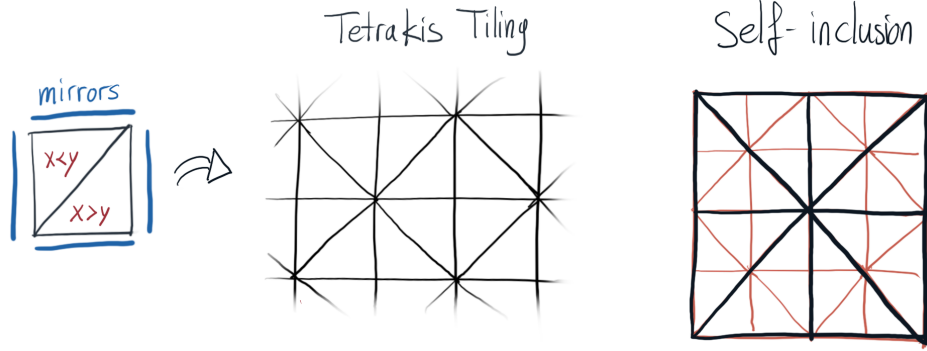
Figure 1: The Tetrakis triangulation in 2D.

full ball $B_{H^\gamma} = Conv(V)$. Pointy balls have much less vertices than boxy ones, e.g. the ℓ_1 ball has $2d$ vertices, while the ℓ_∞ ball has 2^d vertices.

In the context of the HTMC ball we should think of these vertices as *atomic circuits*: circuits that cannot be represented as the average of other simpler circuits. In this section we will identify a set of functions, the Tetrakis functions, that are an approximation for these vertices, in the sense that it is a smaller subset (a countable set, in comparison to the HTMC functions, which are defined as infinite sequence of circuits, and therefore potentially uncountable) whose convex hull is close to the full ball.

And this convex geometric detour will pay off, as these Tetrakis functions will end up playing a central role in our construction bound for the RHS of Theorem 2.

We start from the so-called Tetrakis triangulation of the hyper-grid, which matches the Tetrakis tiling in 2D. We start from a triangulation of the hyper-cube based on permutations, where a permutation $\pi \in S_{d_{in}}$ on d_{in} elements corresponds to the simplex

$$\{x \in [0,1]^{d_{in}} : x_{\pi^{-1}(1)} \leq \cdots \leq x_{\pi^{-1}(d_{in})}\}$$

which has $d_{in} + 1$ vertices $\pi(0, \ldots, 0), \pi(0, \ldots, 0, 1), \ldots, \pi(1, \ldots, 1)$. Conversely, any point $x \in [0,1]^{d_{in}}$ is contained in the simplices corresponding to permutations π that sort x, i.e. such that $\pi(x)$ is non-decreasing.

This triangulation is then extended to the whole grid by mirroring along the faces of the hypercube, leading to a triangulation of the grid that is invariant under translation by even integer (not odd ones). We write $V(x)$ for the sequence of $d_{in} + 1$ tuples (v_i, p_i) of vertices v of the simplex that contains x along with the weights $p \in [0,1]$ such that $x = \sum_{i=1}^{d_{in}+1} p_i v_i$. Note that this decomposition may not be unique, but the non-uniqueness is only in the vertices v_i whose weight p_i is 0.

One of the main parts of the construction will be defining a neural network that maps a point x to the weighted binary representations $(p_i Bin(v_i))_{i=1,\ldots,d_{in}+1}(p_i Bin(v_i))_{i=1,\ldots,d_{in}+1}$. Note that this map is now not only unique/well-defined but also continuous.

Given a circuit C taking $d_{in}M$ binary inputs, i.e. the binary representations of d_{in} integers in the range $\{0, \ldots, 2^M - 1\}$, and returns $d_{out}M_{out}$ binary outputs, we write $TK_M[C] : [0,1]^{d_{in}} \to [0,1]^{d_{out}}$ for the extension of the values of C (which are defined

on the grid $\{n2^{-\frac{M}{2}} : n \in \{0, \ldots, 2^{\frac{M}{2}}\}\}$) to the hyper-cube using the Tetrakis triangulation:

$$TK_M[C](x) = \sum_{(v,p) \in V_M(x)} p \sum_{m=1}^{M_{out}} 2^{-m} C_{\cdot,m}(Bin(v)).$$

The set $\mathcal{TK} = \{TK_M[C] : \forall \text{circuits } C, M \in \mathbb{N}\}$ of Tetrakis function is countable since the circuits are themselves countable. Furthermore, these functions are HTMC computable:

Proposition 7. *There is a universal constant c such that for all circuits C, we have*

$$C_p(TK_M[C], \epsilon; L_\infty) \leq \begin{cases} 0 & \epsilon \geq 1 \\ cd_{in}\left(|C|_p + M - \log_2 \epsilon\right) & \epsilon < 1. \end{cases}$$

and therefore

$$\|TK_M[C]\|_{H_p^\gamma(L_\infty)} \leq \left[cd_{in}\left(|C|_p + M\right)\right]^{\frac{1}{\gamma}}.$$

Proof. Without loss of generality, we may assume that the grid points have integer value, because one could always rescale the inputs x_i to $\frac{x_i - a_i}{w} 2^M$.

We start by identifying the cube that contains x: we compute the closest even and odd grid coordinates

$$x_i^e = 2\left\lfloor \frac{x_i}{2} \right\rfloor + 1, \qquad\qquad x_i^o = 2\left\lfloor \frac{x_i - 1}{2} \right\rfloor$$

as well as the cube coordinates of x_i:

$$s_i = |x_i - x_i^e| \in [0, 1]$$

which equals 1 when x_i equals x_i^o and 0 when it equals x_i^e, and interpolates linearly between them. These can be bomputes can be done in $O(d_{in}(M + \log_2 \epsilon))$ steps.

We now look for the simplex that contains x, by sorting the s_i, yielding a non-decreasing list $t_1, \ldots, t_{d_{in}}$ to which we append $t_0 = 0$ and $t_{d_{in}+1} = 1$. Given the permutation π that sorts the s_i, the $d_{in} + 1$ vertices of the simplex are then

$$(y_0, \ldots, y_{d_{in}}) = x^{\pi^{-1}(e,\ldots,e)}, x^{\pi^{-1}(o,e,\ldots,e)}, \ldots, x^{\pi^{-1}(o,\ldots,o,e)}, x^{\pi^{-1}(o,\ldots,o)},$$

This can rewritten as

$$y_{k,i} = \begin{cases} x_i^e & \pi(i) > k \\ x_i^o & \pi(i) \leq k \end{cases} = \begin{cases} x_i^e & s_i \geq t_{k+1} \\ x_i^o & s_i \leq t_k \end{cases}$$

because $\pi(i) \geq k + 1$ implies $s_i = t_{\pi(i)} \geq t_{k+1}$ because the t_ks are non-decreasing, and similarly $\pi(i) \leq k$ implies $s_i \leq t_k$. The weight of the k-th vertex are then simpy $p_k = t_{k+1} - t_k$. This can all be done with a circuit of size $O(d_{in}(\log d_{in})^2 \log_2 \epsilon)$: the main cost is a sorting network with $d_{in}(\log d_{in})^2$ comparison each of cost $\log_2 \epsilon$.

We then evaluate the circuit C on the vertices and compute the weighted sum

$$TK_M[C](x) = \sum p_k C(y_k).$$

14

which requires $O(d_{in} |C|_p + \log_2 \epsilon)$ circuit size. There is therefore a constant c such that

$$C_p(TK_M[C], \epsilon; L_\infty) \leq \begin{cases} 0 & \epsilon \geq 1 \\ cd_{in}\left(|C|_p + M - \log_2 \epsilon\right) & \epsilon < 1. \end{cases}$$

This also implies that

$$\|TK_M[C]\|^\gamma_{H_p^\gamma(L_\infty)} \leq \max_{\epsilon \geq 1} cd_{in}\left(|C|_p + M + \log_2 \epsilon\right) \epsilon^{-\gamma} = cd_{in}\left(|C| + M\right)$$

since the maximum is always attained at $\epsilon = 1$. $\qquad\square$

Now that we have shown that the Tetrakis functions are included in the HTMC ball, we want to show that their convex hull contains the HTMC ball, by showing that HTMC functions can be written as sums of Tetrakis functions. This turns out to be possible if the underlying norm is the L_∞ norm and if we also assume some Hölder continuity:

Proposition 8. *Consider a function $f : [0,1]^{d_{in}} \to \mathbb{R}^{d_{out}}$ with bounded HTMC norm $\|f\|_{H_p^\gamma(L_\infty)}$ for some $\gamma > 0$ and bounded Hölder norm $\|f\|_{C^\alpha}$ for some $\alpha \in (0,1]$, then there are circuits C_k for $k \geq k_{min} = -\lceil\log_2 \|f\|_\infty\rceil$ such that $|C_k|_p \leq c_1 \|f\|^\gamma_{H_p^\gamma(L_\infty)} 2^{\gamma k}$ and*

$$\left\| f - \sum_{k=k_{min}}^{K} 2^{-k+3} TK_{M(k)}[C_k] \right\|_\infty \leq 2^{-K}$$

for $M(k) = \left\lceil \frac{1}{\alpha}\log_2 \|f\|_{C^\alpha} + \frac{1}{2}\log_2 d_{in} + \frac{1}{\alpha}k \right\rceil$.

Proof. Let $A_1, A_2, \ldots$ be the circuits that approximate f in L_∞-norm $\|f - A_k\|_\infty \leq 2^{-k}$ over the $2^{-M(k)}$-grid, the Tetrakis extension $TK_{M(k)}[A_k]$ for

$$M(k) = \left\lceil \frac{1}{\alpha}\log_2 \|f\|_{C^\alpha} + \frac{1}{2}\log_2 d_{in} + \frac{1}{\alpha}k \right\rceil$$

then satisfies

$$\|f(x) - TK_{k'}[A_k](x)\| \leq \|f(x) - TK_{M(k)}[f](x)\| + \|TK_{M(k)}[f](x) - TK_{M(k)}[A_k](x)\|$$

$$\leq \sum_{v,p \in V(x)} p\left(\|f(x) - f(v)\| + \|f(v) - A_k(v)\|\right)$$

$$\leq \|f\|_{C^\alpha} d_{in}^{\frac{\alpha}{2}} 2^{-\alpha M(k)} + 2^{-k} \leq 2^{-k+1}.$$

Thanks to the self-inclusion property of the Tetrakis triangulation, there is a circuit B_k such that $TK_{M(k)+1}[B_k] = TK_{M(k)}[A_k]$ and with $|B_k|_p \leq c_p |A_k|_p$ for some constant c_p: given a gridpoint x of the finer grid, if the i-th coordinate x_i is odd, we map it to the closest even and odd coordinates x_i^e, x_i^o of the coarser grid, and if it is even, it already lies on the coarser grid and we set $x_i^e = x_i^o = x_i/2$, we then evaluate A_k on x_i^e and x_i^o and average the outputs.

We can therefore approximate f with a telescopic sum

$$\left\| f - \sum_{k=1}^{K} TK_{M(k)}[A_k - B_{k-1}] \right\|_\infty \leq 2^{-K}.$$

15

with the advantage that the summands are small

$$\left\|TK_{M(k)}\left[A_k - B_{k-1}\right]\right\|_\infty \leq \left\|TK_{M(k)}\left[A_k\right] - TK_{M(k-1)}\left[A_{k-1}\right]\right\|_\infty$$
$$\leq \left\|TK_{M(k)}\left[A_k\right] - f\right\|_\infty + \left\|TK_{M(k-1)}\left[A_{k-1}\right] - f\right\|_\infty$$
$$\leq 2^{-k+1} + 2^{-k+2} = 6 \cdot 2^{-k}.$$

This implies that in the signed difference $A_k - B_{k-1}$, i.e. all the bits before location $k - \lceil \log_2 6 \rceil = k - 3$ are zero. There is a circuit C_k with signed outputs such that $2^{-k+3} TK_{M(k)}[C_k] = TK_{M(k)}[A_k - B_{k-1}]$ and there is a constant c_1 such that $|C_k|_{p=\frac{2}{3}} \leq c_1 \|f\|_{H^\gamma(\infty,p)}^\gamma 2^{\gamma k}$ because C_k is obtained by a finite number of sums of the algorithms A_k and A_{k-1}. $\qquad \square$

These two propositions combine into a corollary that illustrate why these Tetrakis functions can be thought of as approximations of the vertices of the HTMC ball intersected with the Hölder ball. More precisely, for all γ, α, we define a certain rescaling of our set of Tetrakis functions

$$\mathcal{TK}_{\gamma,\alpha} = \left\{ \frac{TK_M[C]}{\sqrt[\gamma]{cd_{in}\,|C|} \vee 2^{\alpha M}} : \forall \text{circuits } C, M \in \mathbb{N} \right\}$$

Theorem 9. *For all $\gamma > 2$, $\delta > 0$ and $\alpha \in (0, 1]$, there are constants $c_{\gamma,\alpha,\delta}, C_{\gamma,\alpha,\delta}$ such that*

$$c_{\gamma,\alpha,\delta} d_{in}^{-\left(\frac{1}{\gamma} \vee \frac{\alpha}{2}\right)} \left(B_{H^{\gamma-\delta}(L_\infty)} \cap B_{C^\alpha} \right) \subset \text{Conv } \mathcal{TK}_{\gamma,\alpha} \subset C_{\gamma,\alpha,\delta} \left(B_{H^\gamma(L_2(\pi))} \cap B_{C^\alpha} \right)$$

Proof. (RHS) Since the RHS is convex, we only need to prove that the scaled Tetrakis function are included in the RHS. This follows from the fact that

$$\left\| \frac{TK_M[C]}{\sqrt[\gamma]{cd_{in}\,|C|} \vee 2^{\alpha M}} \right\|_{H^\gamma} \leq \frac{\sqrt[\gamma]{cd_{in}\,(|C| + M)}}{\sqrt[\gamma]{cd_{in}\,|C|} \vee M} \leq \sqrt{2}$$

and

$$\left\| \frac{TK_M[C]}{\sqrt[\gamma]{cd_{in}\,|C|} \vee 2^{\alpha M}} \right\|_{C^\alpha} \leq \frac{\|TK_M[C]\|_{C^\alpha}}{2^{\alpha M}} \leq \frac{2^{\alpha M}}{2^{\alpha M}} = 1$$

because at worst two neighboring grid points have a difference in values of at most 1 and a distance of 2^{-M}.

(LHS) Proposition 8 tells us that for any function f with $\|f\|_{H^{\gamma(1-\delta)}}, \|f\|_{C^{\frac{\alpha}{1-\delta}}} \leq 1$, then there are circuits $(C_k)_{k \geq k_{min}}$ which define a series of approximations

$$\tilde{f}_K - \sum_{k=k_{min}}^{K} 2^{-k+3} TK_{M(k)}[C_k]$$

with $M(k) = \left\lceil \frac{1}{2} \log_2 d_{in} + \frac{1-\delta}{\alpha} k \right\rceil$, $|C_k|_p \leq c_1 2^{\gamma(1-\delta)k}$, and

$$\left\| f - \tilde{f}_K \right\|_\infty \leq 2^{-K}.$$

16

Our goal is now to show that this sum of Tetrakis is a convex combination:

$$\tilde{f}_K = \sum_{k=k_{min}}^{K} 2^{-k+3} \left(\sqrt[\gamma]{cd_{in}\,|C|} \vee 2^{\alpha M(k)} \right) \frac{TK_{M(k)}[C_k]}{\sqrt[\gamma]{cd_{in}\,|C|} \vee 2^{\alpha M(k)}}$$

$$= \sum_{k=k_{min}}^{K} 2^{-k+3} \left(\sqrt[\gamma]{cd_{in}c_1}\, 2^{(1-\delta)k} \vee d_{in}^{\frac{\alpha}{2}} 2^{(1-\delta)k} \right) \frac{TK_{M(k)}[C_k]}{\sqrt[\gamma]{cd_{in}\,|C|} \vee 2^{\alpha M(k)}}$$

$$= \sum_{k=k_{min}}^{K} 2^{3-\delta k} \left(\sqrt[\gamma]{cd_{in}c_1} \vee d_{in}^{\frac{\alpha}{2}} \right) \frac{TK_{M(k)}[C_k]}{\sqrt[\gamma]{cd_{in}\,|C|} \vee 2^{\alpha M(k)}}$$

Therefore, there is a constant $c_{\gamma,\alpha,\delta}$ such that $c_{\gamma,\alpha,\delta} d_{in}^{-\left(\frac{1}{\gamma} \vee \frac{\alpha}{2}\right)} \tilde{f}_K$ is contained in the convex hull for all K. $\qquad\square$

This result can be interpreted as saying that the Tetrakis function are an approximation of the vertices of the HTMC ball intersected with the Hölder ball, up to the constants $c_{\gamma,\alpha,\delta}, C_{\gamma,\alpha,\delta}$, the log terms which lead to the δs and the discrepancy between the L_∞ vs $L_2(\pi)$ norms. Also we did not prove that the Tetrakis functions are the smallest such set, only that it is countable and therefore 'much smaller' than the full ball.

This result also allows us to concretize the idea of using convex optimization to minimize circuit size: one could simply implement a Frank-Wolfe algorithm over the Tetrakis functions, where at each steps, one searches over the list of Tetrakis functions to find the function $TK_M[C]$ most aligned with the current error gradient $\max \langle TK_M[C], f^* - f \rangle$ and does a steps towards $TK_M[C]$.

The countability of the Tetrakis functions possible, though it would still require a close to exhaustive search thus leading to a super-polynomial runtime in the worst case. But the convexity should still yield some advantage, in particular if the true function is a sum of many simple circuits: by decomposing a global search into a sequence of local searches the complexity should be exponential in the size of the largest sub-circuit which could be much smaller than the global circuit size.

This is a very promising direction, but we leave a careful analysis of such an algorithm to follow up work, and instead focus on connecting the HTMC norm to ResNets. The major role that these Tetrakis functions will play in our analysis is a testament to the power of a convex geometric approach to computation in the HTMC regime.

4 ResNets as a Computation Paradigm

In this section, we focus on ResNets and their related (pseudo-)norm $\|f\|_{R^\omega}$, and prove our main sandwich bound, Theorem 2, using the theoretical tools developed in the previous sections.

In Section 4.1, we prove a few basic properties of the ResNet complexity measure and related norm. Section 4.2 then proves the LHS of Theorem 2 in the form of a pruning bound. Finally, the RHS is proven in Section 4.3 with a construction bound based on the Tetrakis functions of Section 3.3.

4.1 Basic Properties

For ResNet norm, we have

Proposition 10. *For two functions f, g, and a norm $\|\cdot\|$, we have*

1. *If $\|f\| \le \|f\|_\infty$ for all f, then $\|f\| \le \omega \left(\frac{2}{\omega-1}\right)^{\frac{\omega-1}{\omega}} \|f\|_{R^\omega(\|\cdot\|)}$,*

2. *$\|f \circ S\|_{C^{\alpha = \omega-1}} \le \omega \left(\frac{2}{\omega-1}\right)^{\frac{\omega-1}{\omega}} \|f\|_{R^\omega(L_\infty)}$,*

3. *$\|f + g\|_{R^\omega} \le \left(\|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} + \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}}\right)^{\frac{1+2\omega}{2\omega}}$,*

4. *$\|f \circ g\|_{R^\omega} \le 2 \left(\|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} + Lip(f)^{\frac{2\omega}{1+2\omega}} \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}}\right)^{\frac{1+2\omega}{2\omega}}.$*

Also note that $\omega \left(\frac{2}{\omega-1}\right)^{\frac{\omega-1}{\omega}} \le 3$ and it converges to 1 as $\omega \searrow 1$.

Proof. (1) $\|f_\theta\|_\infty \le \sum_{i,j=1}^d |W_{out,ij}| \|\alpha_{L,j}\|_\infty = \|W_{out}C_L\|_1 \le R(\theta)$.

More generally, $\|f\| \le \min_\epsilon \|f_{\theta_\epsilon}\|_\infty + \epsilon \le \min_\epsilon \|f\|_{R^\omega}^\omega \epsilon^{1-\omega} + \epsilon = \omega(\omega-1)^{\frac{1-1}{\omega}} \|f\|_{R^\omega}$.

(2) We first observe that $Lip(f_\theta \circ S) \le Lip((\alpha_0 \to f_\theta)D_0^{-1}) \|D_0 W_{in} S\|_{op} \le \|D_0 W_{in} S\|_1 \le R(\theta)$.

Now for all $\epsilon > 0$, we have parameter θ_ϵ such that $\|f - f_{\theta_\epsilon}\|_\infty \le \epsilon$ and $R(\theta_\epsilon) \le \|f\|_{R^\omega}^\omega \epsilon^{1-\omega}$, which implies that $Lip(f_{\theta_\epsilon} \circ S) \le \|f\|_{R^\omega}^\omega \epsilon^{1-\omega}$. Now for any two points x, y, we have

$$\|f(Sx) - f(Sy)\| \le \min_\epsilon \|f(Sx) - f_{\theta_\epsilon}(Sx)\| + \|f_{\theta_\epsilon}(Sx) - f_{\theta_\epsilon}(Sy)\| + \|f_{\theta_\epsilon}(Sy) - f(Sy)\|$$

$$\le \min_\epsilon 2\epsilon + \|f\|_{R^\omega}^\omega \epsilon^{1-\omega} \|x - y\|$$

$$= \omega \left(\frac{2}{\omega-1}\right)^{\frac{\omega-1}{\omega}} \|f\|_{R^\omega} \|x - y\|^{\frac{1}{\omega}}.$$

This implies $\|f \circ S\|_{C^{\alpha = \omega-1}} \le \omega \left(\frac{2}{\omega-1}\right)^{\frac{\omega-1}{\omega}} \|f\|_{R^\omega}$.

(3) We put two networks with parameters θ and ϕ in parallel $\theta|\phi$ so that $f_{\theta|\phi} = f_\theta + f_\phi$. Given E_ℓ and F_ℓ the diagonal matrices that yield the optimal representation costs for θ and ϕ respectively, we choose $D_\ell = \begin{pmatrix} a^{-\frac{1}{2}} E_\ell & 0 \\ 0 & (1-a)^{-\frac{1}{2}} F_\ell \end{pmatrix}$ and obtain

$$Lip((\alpha_\ell \to f_{\theta|\phi}) \circ D_\ell^{-1}) \le \sqrt{a Lip((\alpha_\ell \to f_\theta)E_\ell^{-1})^2 + (1-a)Lip((\alpha_\ell \to f_\phi)F_\ell^{-1})^2} \le 1.$$

We add $\tilde{\ }$ to all the objects corresponding to the second set of parameters ϕ, and since $\min_a a^{-\frac{1}{2}} x + (1-a)^{-\frac{1}{2}} y = \left(x^{\frac{2}{3}} + y^{\frac{2}{3}}\right)^{\frac{3}{2}}$ we obtain

$$R(\theta|\phi)^{\frac{2}{3}} \le \sum_{\ell=1}^{L-1} \left(\min_{a_\ell} \sqrt{a_\ell} \sum_i \|E_\ell W_{\ell,\cdot i}\|_1 \|V_{\ell,i\cdot} C_\ell\|_1 + \sqrt{1-a_\ell} \sum_i \left\|F_\ell \tilde{W}_{\ell,\cdot i}\right\|_1 \left\|\tilde{V}_{\ell,i\cdot} \tilde{C}_\ell\right\|_1\right)^{\frac{2}{3}}$$

$$\le \sum_{\ell=1}^{L-1} \left(\sum_i \|E_\ell W_{\ell,\cdot i}\|_1 \|V_{\ell,i\cdot} C_\ell\|_1\right)^{\frac{2}{3}} + \left(\sum_i \left\|F_\ell \tilde{W}_{\ell,\cdot i}\right\|_1 \left\|\tilde{V}_{\ell,i\cdot} \tilde{C}_\ell\right\|_1\right)^{\frac{2}{3}}$$

$$= R(\theta)^{\frac{2}{3}} + R(\phi)^{\frac{2}{3}}.$$

18

Note that for simplicity, we have dropped the terms corresponding to W_{in} and W_{out}, but they can be handled similarly.

To extend the bound to $\omega > 1$, we choose $\epsilon_f = \left(\|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} + \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} \right)^{-1} \|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} \epsilon$ and $\epsilon_g = \left(\|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} + \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} \right)^{-1} \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} \epsilon$ and choose parameters θ_{ϵ_f} and ϕ_{ϵ_g} such that $\left\| f + g - f_{\theta_{\epsilon_f} | \phi_{\epsilon_g}} \right\| \leq \epsilon_f + \epsilon_g = \epsilon$ and

$$R(\theta_{\epsilon_f} | \phi_{\epsilon_g})^{\frac{2}{3}} \leq R(\theta_{\epsilon_f})^{\frac{2}{3}} + R(\phi_{\epsilon_g})^{\frac{2}{3}} \leq \|f\|_{R^\omega}^{\frac{2\omega}{3}} \epsilon_f^{\frac{2}{3}(1-\omega)} + \|g\|_{R^\omega}^{\frac{2\omega}{3}} \epsilon_g^{\frac{2}{3}(1-\omega)}$$

$$= (\|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} + \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}})^{\frac{2}{3}\omega + \frac{1}{3}} \epsilon^{\frac{2}{3}(1-\omega)}$$

which implies $\|f + g\|_{R^\omega} \leq \left(\|f\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} + \|g\|_{R^\omega}^{\frac{2\omega}{1+2\omega}} \right)^{\frac{1+2\omega}{2\omega}}$.

(4) We obtain $\|f \circ g\|_R \leq 2 \left(\|f\|_R^{\frac{2}{3}} + Lip(f)^{\frac{2}{3}} \|g\|_R^{\frac{2}{3}} \right)^{\frac{2}{3}}$ by simply composing the two networks, with the specificity that the middle W_{in} and W_{out} need to be doubled with opposite signs to leverage the property of the ReLU that $x = \sigma(x) - \sigma(-x)$, thus potentially leading to a doubling of the R-norm. This constant factor could be avoided with some small modifications to the architecture.

A similar optimization over the choice of ϵ_f and ϵ_g yields

$$\|f \circ g\|_R \leq 2 \left(\|f\|_R^{\frac{2\omega}{1+2\omega}} + Lip(f)^{\frac{2\omega}{1+2\omega}} \|g\|_R^{\frac{2\omega}{1+2\omega}} \right)^{\frac{1+2\omega}{2\omega}}.$$

$\square$

We see the similarity with the HTMC norm, with the notable additional property that the R-norm controls the Lipschitzness of the function, which is not at all the case for the HTMC norm, and this will result in the additional Hölder norm term in in the LHS of the main sandwich bound.

4.2 Pruning Bound

We can now prove our pruning bound

Theorem 11. *For any ResNet f_θ, one has for all $\gamma > 2$, $\|f_\theta\|_{H^\gamma} \leq c_\gamma R(\theta)$.*
Furthermore, for all $\omega > 1$ and $\delta > 0$ we have

$$\|f\|_{H^{\gamma = 2\omega + \delta}} \leq c_{\omega, \delta} \|f\|_{R^\omega (L_2(\pi))}.$$

Proof. (1) We write $f_\theta = f_{L:1}$ for $f_\ell(x) = D_\ell \left(x + W_\ell \sigma(V_\ell^T(x, 1)) \right) D_{\ell-1}^{-1}$, where we write $(x, 1)$ for the vector x with a 1 concatenated at the end, and the D_ℓs are diagonal matrices such that $Lip((\alpha_\ell \to f_\theta) \circ D_\ell^{-1}) = Lip(f_{L:\ell+1}) \leq 1$.

Since the outputs of f_ℓ will be contained inside the hyper-rectangle with sides $D_\ell C_\ell$, we may use the compositionality of the HTMC norm (Proposition 5):

$$\|f_\theta\|_{H^\gamma(\pi)} \leq \sum_{\ell=1}^{L} \|f_\ell\|_{H^\gamma(\pi_{\ell-1})},$$

where π_ℓ is any distribution supported on the hyper-rectangle with sides $D_\ell C_\ell$.

We then use the convexity of the HTMC norm (Theorem 4) to bound the HTMC norm of f_ℓ

$$\|f_\ell\|_{H^\gamma(\pi_{\ell-1})} \leq c_\gamma \sum_{i=1}^{w} \left\| D_\ell W_{\ell,\cdot i}\sigma(V_{\ell,i\cdot}^T(D_{\ell-1}^{-1}x,1)) \right\|_{H^\gamma(\pi_{\ell-1})}.$$

For a single neuron function $D_\ell W_{\ell,\cdot i}\sigma(V_{\ell,i\cdot}^T(D_{\ell-1}^{-1}x,1))$, the intermediate representation is bounded $\left| V_{\ell,i\cdot}^T(D_{\ell-1}^{-1}x,1) \right| \leq \sum_{j=1}^{d+1} |V_{\ell,ij}| C_{\ell-1,jj}$, we can therefore split it into functions $g_i(x) = V_{\ell,i\cdot}^T(D_{\ell-1}^{-1}x,1)$ and $h_i(z) = D_\ell W_{\ell,i}\sigma(z)$. For all $\gamma > 0$, there is a constant b_γ such that the HTMC norm of scalar multiplication is bounded $\|x \mapsto ax\|_{H^\gamma(\pi)} \leq b_\gamma |a| |c|$ for any distribution supported in $[-c,c]$. Similarly for the ReLU followed by a multiplication, we have $\|x \mapsto a\sigma(x)\|_{H^\gamma(\pi)} \leq b_\gamma |a| |c|$. We can now bound the HTMC norms of g and h using Theorem 4:

$$\|g_i\|_{H^\gamma(\pi_{\ell-1})} = \left\| x \mapsto \sum_{j=1}^{d} V_{\ell,ij}D_{\ell-1,jj}^{-1}x_j + V_{\ell,d+1} \right\|_{H^\gamma(\pi_{\ell-1})}$$

$$\leq c_\gamma b_\gamma \sum_j |V_{\ell,ij}| C_{\ell-1,jj}$$

$$\|h_i\|_{H^\gamma(\pi_{\ell-1})} \leq \left\| z \mapsto \sum_{k=1}^{d} D_{\ell,kk}W_{\ell,ki}\sigma(z) \right\|_{H^\gamma(g\#\pi_{\ell-1})}$$

$$\leq c_\gamma b_\gamma \left(\sum_k D_{\ell,kk} |W_{\ell,ki}| \right) \left(\sum_j |V_{\ell,ij}| C_{\ell-1,jj} \right)$$

Since we also have $Lip(h) \leq \|D_\ell W_{\ell,\cdot i}\|_1$, we obtain

$$\left\| x \mapsto W_{\ell,\cdot i}\sigma(V_{\ell,i\cdot}^T(x,1)) \right\|_{H^\gamma(\pi_{\ell-1})} \leq c^{\frac{1}{\gamma}}2^{\frac{\gamma+1}{\gamma}}c_\gamma b_\gamma D_\ell \|W_{\ell,\cdot i}\|_1 \|V_{\ell,i\cdot}C_{\ell-1}\|_1$$

and therefore

$$\|f_\ell\|_{H^\gamma(\pi_{\ell-1})} \leq c^{\frac{1}{\gamma}}2^{\frac{\gamma+1}{\gamma}}c_\gamma^2 b_\gamma \||W_\ell||V_\ell|C_{\ell-1}\|_1.$$

Applying Proposition 5 to the layer-wise composition to obtain

$$\|f_\theta\|_{H^\gamma(\pi)} \leq c^{\frac{2}{\gamma}}2^{\frac{\gamma+1}{\gamma}}c_\gamma^2 b_\gamma \sum_\ell \left(\||W_\ell||V_\ell|C_{\ell-1}\|_1^{\frac{\gamma}{\gamma+1}} \right)^{\frac{\gamma+1}{\gamma}}$$

(2) By the definition of the $R^\omega(L_2(\pi))$ norm, we have parameters θ_ϵ for all ϵ such that $\|f - f_{\theta_\epsilon}\|_\pi \leq \epsilon$ and $R(\theta_\epsilon) \leq \|f\|_{R^\omega(L_2(\pi))}^\omega \epsilon^{1-\omega}$. On the other hand, the first part of the proof implies that for all $\gamma > 2$ there is a circuit A_ϵ with error $\|A_\epsilon - f_{\theta_\epsilon}\|_\pi \leq \epsilon$ and size $|A_\epsilon| \leq C_\gamma^\gamma R(\theta_\epsilon)^\gamma \epsilon^{-\gamma}$ for $C_\gamma = c^{\frac{2}{\gamma}}2^{\frac{\gamma+1}{\gamma}}c_\gamma^2 b_\gamma$. Putting the two together, we have

$$C(f, 2\epsilon) \leq C_\gamma^\gamma \|f\|_{R^\omega(L_2(\pi))}^{\omega\gamma} \epsilon^{-\gamma\omega}$$

which implies that $\|f\|_{H^{\gamma\omega}} \leq 2C_\gamma^{\frac{1}{\omega}} \|f\|_{R^\omega(L_2(\pi))}$ and if one chooses $\gamma = 2 + \omega^{-1}\delta$, we obtain

$$\|f\|_{H^{\gamma=2\omega+\delta}} \leq c_{\omega,\delta} \|f\|_{R^\omega(L_2(\pi))}.$$

$\square$

This proof was written with a heavy use of Theorem 4 and Proposition 5, to showcase the versatility of these results, but it does obfuscate what is happening concretely. The same result could also be proven as a pruning bound, where the weights $W_{\ell,ki}$ and $V_{\ell,ij}$ are removed with probabilities proportional to $D_{\ell,kk} \|V_{\ell,i}.C_{\ell-1}\|_1$ and $\|D_\ell W_{\ell,\cdot i}\|_1 C_{\ell-1,jj}$ respectively, combined with some MLMC techniques.

4.3   Construction Bound

We have shown in Section 3.3 how any function f that is both HTMC computable and Hölder continuous can be represented as a sum of $O(\log_2 \epsilon)$ Tetrakis functions. This implies that if ResNets can represent these Tetrakis function then they can represent general HTMC functions.

We will construct a ResNet that roughly follows the structure of the proof of Proposition 7: first the input x is mapped to the weighted binary representations of its surrounding vertices $(p_i Bin(v_i))_{i=1,\ldots,d_{in}+1}$ and second the circuit is evaluated on each of the $d_{in} + 1$ vertices in parallel before being summed up. The first part can itself be decomposed into two parts, the first of which computes the binary representations of each coordinate using an iterative method, and the second part implements a sorting algorithm to recover the simplex that contains x.

For the second part, we rely on the following result that describes how to represent a binary circuit as min/max circuit:

Proposition 12. *Given a circuit C, there is a network with parameters θ that represents the circuit in the sense that for any input $x \in \{-a, a\}^{d_{in}}$, we have $f_\theta(x) = aC(\frac{x}{a})$ (where the binary values are represented as ± 1 instead of the usual $0, 1$). Furthermore, if $\|x\|_\infty \leq b$ over the input domain, we have*

$$R(\theta) \leq c\sqrt{d_{out}}b \left|C\right|_{p=\frac{2}{3}} .$$

Proof. We simply replace AND and OR with min and max respectively and NOT with negation. Each of these operations can be represented with a finite number of neurons, and we put them together into a neural network that is sparse, i.e. there is a $c > 1$ such that there are at most $c|C|$ nonzero entries and all entries have absolute value at most c. Note that since we are considering a ResNet, the 'unused' variables do not simply vanish, however, with a wide enough network there is always enough additional room that we can wait for the last layer to drop those unused variables.

Let us now compute the Lipschitz constants $Lip(f_{L:\ell})$, which takes the intermediate representations z and computes the outputs as a mix of max, min, and negations. The function $f_{L:\ell}$ is piecewise linear, with each 'piece' corresponding to inputs z where all max and min are attained at only one of these entries. Therefore the gradient $\nabla(f_{L:\ell})_k$ of the k-th output has only one non-zero entry that is either 1 or -1. We therefore have that $Lip(f_{L:\ell}) \leq \sqrt{d_{out}}$.

Since the activations $\alpha_j^{(\ell)}(x)$ are bounded by $\|x\|_\infty \leq b$, we obtain that

$$\left(\sum_{\ell=1}^{L-1} \left(Lip(f_{L:\ell})b \sum_i \|W_{\ell,\cdot i}\|_1 \|V_{\ell,i\cdot}\|_1 \right)^{\frac{2}{3}} \right)^{\frac{3}{2}} \leq c\sqrt{d_{out}\mathbb{E}_{x\sim\pi} \|x\|_\infty^2} \left|C\right|_{p=\frac{2}{3}} .$$

$\square$

We can describe the ResNet representation of Tetrakis functions:

Proposition 13. *Given a circuit* $C : \{0,1\}^{d_{in} \times M} \mapsto \{0,1\}^{d_{out} \times M_{out}}$, *then*

$$\|TK_M[C]\|_R = O\left(\sqrt{d_{out}}\left(d_{in}\sqrt{M}2^M + d_{in}(\log d_{in})^2 + \sqrt{(d_{in}+1)}\,|C|_{p=\frac{2}{3}}\right)\right).$$

Proof. The construction of the ResNet will be in three parts $f = f_3 \circ f_2 \circ f_1$ going through dimensions d_{in}, $d_{in}(2k+1)$, $(d_{in}+1)d_{in}k$ and finally d_{out}. First f_1 takes an input x and computes for each coordinate x_i the weighted binary representations of the closest even and odd grid coordinate $(1-s_i)Bin_k(x_i^e), s_i Bin_k(x_i^o)$ together with the weight $s_i \in [0,1]$. Second, f_2 takes these triples for each coordinate and computes the $d_{in}+1$ weighted binary representations $(p_i Bin(v_i))_{i=1,\ldots,d_{in}+1}$ of the vertices v_i of the simplex that contains x. Finally, f_3 applies the circuit in parallel to each of these representations before summing them up.

1. The first function is applied in parallel to each coordinate. And for each coordinate $x_i \in [a_i, a_i + w]$ we use recursion on the scale of the grid, we first map x_i to the weight $s_{1,i} = \frac{x_i - a_i}{w}$ and the weighted even and odd representations $r_{i,1,1}^e = s_{1,i} - 1$ and $r_{i,1,1}^o = s_{1,i}$ (there is a single even grid point at a_i and one odd one at $a_i + w$ corresponding to the integer 0 and 1 respectively, with binary representations -1 and 1).

For the recursion, we assume that we are given $s_{i,m} \in [0,1]$, $r_{i,m}^e = (1 - s_{i,m})Bin_m(x_{i,m}^e)$ and $r_{i,m}^o = (1 - s_{i,m})Bin_m(x_{i,m}^o)$ for $1 \le m < k$. First note that $s_{i,m+1} = 2\min\{s_{i,m}, 1 - s_{i,m}\}$ $x_{i,m+1}^e = \begin{cases} 2x_{i,m}^e & \text{if } s_{i,m} \le 0.5 \\ 2x_{i,m}^o & \text{if } s_{i,m} \ge 0.5 \end{cases}$ (the jump at $s_{i,m} = 0.5$ does not matter, because all things that depend on $x_{i,m+1}^e$ are multiplied $(1 - s_{i,m+1})$ which equals 0 at the jump) and $x_{i,m+1}^o = x_{i,m}^e + x_{i,m}^o$. Let us first define a helper function $g : \{(a,b) : |a| + |b| = 1\} \to \mathbb{R}^4$ whose values on the ℓ_1 unit ball interpolate linearly between 8 points located at the corner of the ball and the midpoints between corners, taking values:

$$g(+1,0) = (+1,-1;0,0) \qquad g(+\tfrac{1}{2},+\tfrac{1}{2}) = (0,0;+1,-1)$$

$$g(0,+1) = (+1,-1;0,0) \qquad g(-\tfrac{1}{2},+\tfrac{1}{2}) = (0,0;-1,+1)$$

$$g(-1,0) = (-1,-1;0,0) \qquad g(-\tfrac{1}{2},-\tfrac{1}{2}) = (0,0;-1,-1)$$

$$g(0,-1) = (-1,-1;0,0) \qquad g(+\tfrac{1}{2},-\tfrac{1}{2}) = (0,0;-1,+1).$$

You should think of g as taking a mix of two binary numbers $(1-s)a$ and sb for $a, b \in \{-1,1\}$, there are then two cases:

- If $s = \frac{1}{2}$, g returns two pairs, with the first pair being a binary representation with zero weight, and the second pair a binary representation of $a + b$ with weight 1.

- If $s = 0, 1$, g returns two pairs, with the first being a binary representation of $2a$ or $2b$ (depending on whether $s = 0$ or $s = 1$) with weight 1, and a second binary pair with weight 0.

The homogeneous extension of g from the unit disk to the plane can be represented by a finite number of neurons organized in at most two layers.

We may now compute $r_{i,m+1}^e, r_{i,m+1}^o$ according to the formula

$$r_{i,m+1,j}^e = g_1(r_{i,m,j}^e, r_{i,m,j}^o) \vee g_2(r_{i,m,j-1}^e, r_{i,m,j-1}^o)$$
$$r_{i,m+1,j}^o = g_3(r_{i,m,j}^e, r_{i,m,j}^o) \vee g_4(r_{i,m,j-1}^e, r_{i,m,j-1}^o),$$

and

$$r_{i,m+1,m+1}^e = g_2(r_{i,m,m}^e, r_{i,m,m}^o)$$
$$r_{i,m+1,m+1}^o = g_4(r_{i,m,m}^e, r_{i,m,m}^o),$$

Note that since $g_2(a,b) \leq g_1(a,b)$, one could remove it from the formula. The maximum $\vee$ here plays the role of a sum of two binary numbers, and thankfully we never have to sum two 1s so we are fine: indeed in the first case g_2 always outputs -1, and in the second case g_3 is 1 if both $Bin(x_{i,m}^e)_j$ and $Bin(x_{i,m}^o)_j$ are 1 and g_4 is 1 if $Bin(x_{i,m}^e)_{j-1} \neq Bin(x_{i,m}^o)_{j-1}$, but since $x_{i,m}^e$ and $x_{i,m}^o$ are adjacent integer, if they differ at one digit, they must differ at all subsequent digits, i.e if $Bin(x_{i,m}^e)_{j-1} \neq Bin(x_{i,m}^o)_{j-1}$ then $Bin(x_{i,m}^e)_j \neq Bin(x_{i,m}^o)_j$ too.

We can therefore implement this recursion as a ResNet with parameters θ_1 to compute $(s_{i,k}, r_{i,k,\cdot}^e, r_{i,k,\cdot}^o)$. In general the map from the level m grid $(s_{i,m}, r_{i,m,\cdot}^e, r_{i,m,\cdot}^o)$ to the level k grid $(s_{i,k}, r_{i,k,\cdot}^e, r_{i,k,\cdot}^o)$ is $O(\sqrt{k}2^{k-m})$ -Lipschitz, because each $r_{i,k}^e$ depends only one of the $r_{i,m,j}^e$s or $r_{i,m,j}^o$s (the one that attains all the maxima), and this dependence is of order 2^{k-m}. Furthermore, since all activations are of order 1, we have

$$R(\theta_1) = O\left(d_{in} \sum_{m=1}^k \sqrt{k}2^{k-m} \right) = O\left(\sqrt{d_{in}k}2^k \right).$$

2. For the second part, we are given the coordinate weights s_i and weighted representations r_i^e, r_i^o of the closest even and odd grid coordinate (we drop the k index $s_{i,k}, r_{i,k}^e, r_{i,k}^o$ in the previous part of the proof, since we are only working on the finest grid), and our goal is to compute the weighted binary representations of the vertices of the simplex containing x. That is for all $m = 0, \ldots, d_{in}$ and $i = 1, \ldots, d_{in}$ we need to compute the weights $p_m = (s_{\pi^{-1}(m+1)} - s_{\pi^{-1}(m)})$ for each vertices and the weighted binary representations:

$$A_{m,i} = (s_{\pi^{-1}(m+1)} - s_{\pi^{-1}(m)}) \begin{cases} Bin(x_i^e) & m \geq \pi(i) \\ Bin(x_i^o) & m \leq \pi(i) - 1 \end{cases}$$

where π is the permutation that sorts the s_is, and we we define $s_{\pi^{-1}(0)} = 0$ and $s_{\pi^{-1}(d_{in}+1)} = 1$ so that all differences $(s_{\pi^{-1}(m+1)} - s_{\pi^{-1}(m)})$ are positive and sum up to 1.

We first sort the $s_1, \ldots, s_{d_{in}}$ into another list $t_1, \ldots, t_d$ (i.e. $t_m = s_{\pi^{-1}(m)}$), which can be done with a 'sorting network' where any two pair values x, y are sorted using the formula $(x \wedge y, x \vee y)$. This requires a $O((\log d_{in})^2)$ depth and $O(d_{in}(\log d_{in})^2)$ comparisons. We then also define $t_0 = 0$ and $t_{d_{in}+1} = 1$.

Now note that because π sorts the s_i, if $m \geq \pi(i)$ then $t_m \geq s_i$ and if $m \leq \pi(i) - 1$ then $t_{m+1} \leq s_i$, we can therefore rewrite

$$A_{m,i} = (t_{m+1} - t_m) \begin{cases} Bin(x_i^e) & t_m \geq s_i \\ Bin(x_i^o) & t_{m+1} \leq s_i. \end{cases}$$

23

This can be rewritten

$$A_{m,i} = \left(t_{m+1} - [s_i]_{t_m}^{t_{m+1}}\right) Bin(x_i^e) + \left([s_j]_{t_i}^{t_{i+1}} - t_i\right) Bin(x_j^o)$$

in terms of the clamp function $[x]_a^b = (x \vee a) \wedge b$, using the fact that

$$\left(t_{m+1} - [s_i]_{t_m}^{t_{m+1}}\right) = \begin{cases} t_{m+1} - t_m & t_m \geq s_i \\ 0 & t_{m+1} \leq s_i \end{cases}$$

$$\left([s_i]_{t_m}^{t_{m+1}} - t_m\right) = \begin{cases} 0 & t_m \geq s_i \\ t_{m+1} - t_m & t_{m+1} \leq s_i \end{cases}.$$

Writing $d_{m,i}^e = \left(t_{m+1} - [s_i]_{t_m}^{t_{m+1}}\right)$ and $d_{m,i}^e = \left([s_i]_{t_m}^{t_{m+1}} - t_m\right)$, the A_{ij} can now be expressed in terms of the weighted binary representations $r_i^e = (1 - s_i)Bin(x_i^e)$ and $r_i^o = s_iBin(x_i^e)$ instead

$$A_{m,i} = [r_i^e]_{-d_{m,i}^e}^{+d_{m,i}^e} + [r_i^o]_{-d_{m,i}^o}^{+d_{m,i}^o}$$

where we used the fact that $d_{m,i}^e \leq (1 - s_i)$ and $d_{m,i}^o \leq s_i$.

We can build a ResNet layers with parameters θ_2 that represent this second part of the function, with $R(\theta_2) = O(kd_{in}(\log d_{in})^2)$.

3. Finally we have to evaluate the circuit $C : \{0,1\}^{d_{in} \times k} \mapsto \{0,1\}^{d_{out} \times m}$ in parallel to $A_0, \ldots, A_{d_{in}}$ and sum up the outputs. We use Proposition 12 to convert C into a neural network that is repeated in parallel $d_{in} + 1$ times, yielding $B_0, \ldots, B_{d_{in}} \in \mathbb{R}^{d_{out} \times m}$. Since $A_i = w_i Bin(v_i)$ for weights $w_0, \ldots, w_{d_{in}} \geq 0$ that sum up to one, and for vectors $v_0, \ldots, v_{d_{in}}$, we know that $B_i = w_i C(Bin(v_i))$. The final outputs of the network are then obtained as

$$b \sum_{j=1}^{m} 2^{-j} \left(1 + \sum_{i=0}^{d_{in}} B_{i,\cdot,j}\right) = \sum_{j=1}^{m} 2^{-j+1} \sum_{i=0}^{d_{in}} w_i C\left(\frac{Bin(v_i) + 1}{2}\right) \in [0,b]^{d_{out}}.$$

The parameters θ_3 of this last part will satisfy

$$R(\theta_3) \leq C\sqrt{(d_{in}+1)d_{out}} \, |C|_{p=\frac{2}{3}}.$$

Overall, the parameters θ will satisfy

$$\begin{aligned} R(\theta) &\leq R(\theta_1)Lip(f_{\theta_3} \circ f_{\theta_2}) + R(\theta_2)Lip(f_{\theta_3}) + R(\theta_3) \\ &= O\left(d_{in}\sqrt{k}2^k\sqrt{d_{out}} + kd_{in}(\log d_{in})^2\sqrt{d_{out}} + \sqrt{(d_{in}+1)d_{out}} \, |C|_{p=\frac{2}{3}}\right) \\ &= O\left(\sqrt{d_{out}}\left(d_{in}\sqrt{k}2^k + kd_{in}(\log d_{in})^2 + \sqrt{(d_{in}+1)} \, |C|_{p=\frac{2}{3}}\right)\right) \end{aligned}$$

$\square$

The RHS of Theorem 2 then follows directly from the representation of HTMC functions as sums of Tetrakis function of Proposition 8 from Section 3.3 along with some of the basic properties of the ResNet norm from Section 4.1:

Theorem 14. *For any $\omega > 1$, and any $\delta > 0$, we define $\gamma = \omega - \delta$ and $\alpha = \frac{1}{\omega - \delta}$, then there is a constant $c_{\omega,\delta}$ such that*

$$\|f\|_{R^\omega}^\omega \lesssim \sqrt{\frac{1}{\alpha} d_{out} d_{in}} \left(d_{in} \|f\|_{C^\alpha}^{\frac{1}{\alpha}} + \|f\|_{H_{p=\frac{2}{3}}^\gamma (L_\infty)}^\gamma \right),$$

where D is the diagonal matrix with diagonal entries equal to the sidelengths of the hyper-rectangle that contains the input distribution.

Proof. Proposition 8 guarantees that $\tilde{f}_K = \sum_{k=k_{min}}^K 2^{-k+3} T K_{M(k)}[C_k]$ approximates f within an 2^{-K} error. Proposition 13 then gives us

$$\left\|\tilde{f}_K\right\|_R \leq \sqrt{K - k_{min}} \sum_{k=k_{min}}^K \left\|2^{-k+3} T K_{M(k)}[C_k]\right\|_R$$

$$\lesssim \sqrt{d_{out} \log_2 \frac{\|f\|_\infty}{\epsilon}} \sum_{k=k_{min}}^K 2^{-k+3} \left(d_{in} \sqrt{M(k)} 2^{M(k)} + d_{in} (\log d_{in})^2 + \sqrt{d_{in} + 1} \, |C|_{p=\frac{2}{3}} \right)$$

$$\lesssim \sqrt{d_{out} d_{in} \log_2 \frac{\|f\|_\infty}{\epsilon}} \sum_{k=k_{min}}^K d_{in} \|f\|_{C^\alpha}^{\frac{1}{\alpha}} 2^{(\frac{1}{\alpha}-1)k} + \|f\|_{H_p^\gamma(L_\infty)}^\gamma 2^{(\gamma-1)k}$$

$$\lesssim \sqrt{\frac{1}{\alpha} d_{out} d_{in}} \left(d_{in} \|f\|_{C^\alpha}^{\frac{1}{\alpha}} + \|f\|_{H_p^\gamma(L_\infty)}^\gamma \right) \epsilon^{-(\omega-\delta-1)} \log_2 \frac{\|f\|_\infty}{\epsilon}.$$

This then implies that

$$\|f\|_{R^\omega}^\omega \leq \sqrt{\frac{1}{\alpha} d_{out} d_{in}} \left(d_{in} \|f\|_{C^\alpha}^{\frac{1}{\alpha}} + \|f\|_{H_p^\gamma(L_\infty)}^\gamma \right) \|f\|_\infty^\delta \max_{\epsilon \leq \|f\|_\infty} \left(\frac{\|f\|_\infty}{\epsilon} \right)^{-\delta} \log_2 \frac{\|f\|_\infty}{\epsilon}$$

$$\lesssim \sqrt{\frac{1}{\alpha} d_{out} d_{in}} \left(d_{in} \|f\|_{C^\alpha}^{\frac{1}{\alpha}} + \|f\|_{H_p^\gamma(L_\infty)}^\gamma \right) \|f\|_\infty^\delta .$$

And since $\|f\|_\infty^\delta \leq \|f\|_{R^\omega}^\delta$, we obtain

$$\|f\|_{R^\omega} \lesssim \left(\frac{1}{\alpha} d_{out} d_{in} \right)^{\frac{1}{\omega-\delta}} \left(d_{in}^{\frac{1}{\omega-\delta}} \|f\|_{C^\alpha} + \|f\|_{H_p^\gamma(L_\infty)} \right).$$

$\square$

5 Conclusion

This paper establishes a connection between two widely distinct models of real-valued computations: the discrete paradigm of binary computation, and the continuous paradigm of DNN computation. This opens the door to a whole range of similar results for different neural network architectures, or different notions of parameter complexity.

Looking forward, this is just the first step towards proving how DNNs implicitly minimize circuit size. To achieve this, we will need to study the training dynamics of DNNs under gradient descent, which are notoriously hard to capture. I believe that the convexity of the HTMC regime in function space will be key towards this goal, because most previous

DNN convergence proofs follow from the emergence of a 'hidden' limiting convexity [11, 26]. Some of my preliminary inquiries are promising: it appears that the function space convexity translates into an asymptotic absence of barriers in parameter space, which suggests that the main difficulty will lie in proving that gradient descent manages to escape saddles.

In comparison to a brute force search, we can expect ResNet to have a similar advantage as applying Frank-Wolfe on the Tetrakis functions: if the optimal circuit can be written as a sum of multiple circuits then one can learn each individual circuit separately, leading to an exponential complexity in the size of the largest subcircuit, which can be much smaller. Early inquiries suggest that each saddle escape roughly corresponds to the learning of a new sub-circuit, or a step of the Frank-Wolfe algorithm.

Even more exciting, the subadditivity under composition could make it possible to learn a composition of simple circuits by learning each sub-circuits separately. Previous results indicate that this type of compositional learning is possible, but it seems to require the presence of correlations between the intermediate steps and the outputs to guide the learning process [33, 8]. This seems to match the way humans learn and teach: explaining intermediate steps can make extremely complex ideas learnable.

if successful, this would describe DNNs as highly general statistical models, capable of learning any circuit/algorithm as long as it can be decomposed into the sum and composition of simple circuits, thus describing a very large family of tasks all learnable with the same model. This could explain why DNNs are the only models that have been able to approach human intelligence in its generality.

References

[1] El Mehdi Achour, François Malgouyres, and Sébastien Gerchinovitz. Global minimizers, strict and non-strict saddle points, and implicit regularization for deep linear neural networks, 2021.

[2] Francis Bach. Breaking the curse of dimensionality with convex neural networks. *The Journal of Machine Learning Research*, 18(1):629–681, 2017.

[3] Andrew R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Inf. Theory*, 39:930–945, 1993.

[4] Andrew R Barron and Jason M Klusowski. Complexity, statistical risk, and metric entropy of deep nets using total path variation. *stat*, 1050:6, 2019.

[5] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.

[6] Benedikt Bauer and Michael Kohler. On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *The Annals of Statistics*, 2019.

[7] M. S. Birman and M. Z. Solomjak. Piecewise-polynomial approximations of functions of the classes W_p^α. *Mathematics of The USSR-Sbornik*, 2:295–317, 1967.

[8] Francesco Cagnetta, Leonardo Petrini, Umberto M. Tomasini, Alessandro Favero, and Matthieu Wyart. How deep neural networks learn compositional data: The random hierarchy model. *Phys. Rev. X*, 14:031001, Jul 2024.

[9] Minshuo Chen, Kaixuan Huang, Tuo Zhao, and Mengdi Wang. Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 4672–4712. PMLR, 2023.

[10] Zhengdao Chen. Neural hilbert ladders: Multi-layer neural networks in function space. *Journal of Machine Learning Research*, 25(109):1–65, 2024.

[11] Lénaïc Chizat and Francis Bach. On the Global Convergence of Gradient Descent for Over-parameterized Models using Optimal Transport. In *Advances in Neural Information Processing Systems 31*, pages 3040–3050. Curran Associates, Inc., 2018.

[12] Zhen Dai, Mina Karzand, and Nathan Srebro. Representation costs of linear neural networks: Analysis and design. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[13] David A Danhofer, Davide D'Ascenzo, Rafael Dubach, and Tomaso Poggio. Position: A theory of deep learning must include compositional sparsity. *arXiv preprint arXiv:2507.02550*, 2025.

[14] David Donoho. The kolmogorov sampler, 01 2002.

[15] Weinan E, Chao Ma, and Lei Wu. Barron spaces and the compositional function spaces for neural network models. *arXiv preprint arXiv:1906.08039*, 2019.

[16] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022.

[17] Michael B Giles. Multilevel monte carlo methods. *Acta numerica*, 24:259–328, 2015.

[18] John M. Hitchcock and A. Pavan. On the NP-Completeness of the Minimum Circuit Size Problem. In Prahladh Harsha and G. Ramalingam, editors, *35th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2015)*, volume 45 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 236–245, Dagstuhl, Germany, 2015. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[19] Daniel Hsu, Ziwei Ji, Matus Telgarsky, and Lan Wang. Generalization bounds via distillation. In *International Conference on Learning Representations*, 2021.

[20] Marcus Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2005.

[21] Rahul Ilango. Sat reduces to the minimum circuit size problem with a random oracle. pages 733–742, 11 2023.

[22] Rahul Ilango, Bruno Loff, and Igor C. Oliveira. Np-hardness of circuit minimization for multi-output functions. In *CCC '20*, Dagstuhl, DEU, 2020. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[23] Arthur Jacot. Bottleneck structure in learned features: Low-dimension vs regularity tradeoff. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 23607–23629. Curran Associates, Inc., 2023.

[24] Arthur Jacot. Implicit bias of large depth networks: a notion of rank for nonlinear functions. In *The Eleventh International Conference on Learning Representations*, 2023.

[25] Arthur Jacot, Seok Hoan Choi, and Yuxiao Wen. How DNNs break the curse of dimensionality: Compositionality and symmetry learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[26] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. In *Advances in Neural Information Processing Systems 31*, pages 8580–8589. Curran Associates, Inc., 2018.

[27] Arthur Jacot, François Ged, Berfin Şimşek, Clément Hongler, and Franck Gabriel. Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity, 2022.

[28] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. *arXiv preprint arXiv:1912.02178*, 2019.

[29] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[30] Andrei N Kolmogorov. Three approaches to the quantitative definition of information. *Problems of information transmission*, 1(1):1–7, 1965.

[31] Leonid Levin. Hardness of search problems. https://www.cs.bu.edu/fac/lnd/research/hard.htm. Accessed: 2025-09-26.

[32] Leonid Levin. Universal search problems. *Problemy peredachi informatsii*, 9.3:115–116, 1973.

[33] Eran Malach. Auto-regressive next-token predictors are universal learners. In *Proceedings of the 41st International Conference on Machine Learning*, pages 34417–34431, 2024.

[34] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. Norm-based capacity control in neural networks. In *Conference on learning theory*, pages 1376–1401. PMLR, 2015.

[35] Kazusato Oko, Shunta Akiyama, and Taiji Suzuki. Diffusion models are minimax optimal distribution estimators. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 26517–26582. PMLR, 23–29 Jul 2023.

[36] Greg Ongie, Rebecca Willett, Daniel Soudry, and Nathan Srebro. A function space view of bounded norm infinite width relu nets: The multivariate case. In *International Conference on Learning Representations*, 2020.

[37] Tomaso Poggio and Maia Fraser. Compositional sparsity of learnable functions. *Bulletin of the American Mathematical Society*, 61(3):438–456, 2024.

[38] Tomaso Poggio, Hrushikesh Mhaskar, Lorenzo Rosasco, Brando Miranda, and Qianli Liao. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. *International Journal of Automation and Computing*, 14(5):503–519, 2017.

[39] Pedro Savarese, Itay Evron, Daniel Soudry, and Nathan Srebro. How do infinite width bounded norm networks look in function space? In Alina Beygelzimer and Daniel Hsu, editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 2667–2690. PMLR, 25–28 Jun 2019.

[40] Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *The Annals of Statistics*, 48(4):1875 – 1897, 2020.

[41] Ray J Solomonoff. A preliminary report on a general theory of inductive inference, 1960.

[42] Ray J. Solomonoff. A formal theory of inductive inference. part i. *Inf. Control.*, 7:1–22, 1964.

[43] Ray J. Solomonoff. A formal theory of inductive inference. part ii. *Inf. Control.*, 7:224–254, 1964.

[44] Boris Trakhtenbrot. A survey of russian approaches to perebor (brute-force searches) algorithms. *Annals of the History of Computing*, 6(4):384–400, 1984.

[45] Joel Veness, Kee Siong Ng, Marcus Hutter, William Uther, and David Silver. A monte-carlo aixi approximation. *Journal of Artificial Intelligence Research*, 40:95–142, 2011.

[46] Colin Wei and Tengyu Ma. Data-dependent sample complexity of deep neural networks via lipschitz augmentation. *Advances in Neural Information Processing Systems*, 32, 2019.

[47] Blake Woodworth, Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Kernel and deep regimes in overparametrized models. *arXiv preprint arXiv:1906.05827*, 2019.