A generative model for fBm with deep ReLU neural networks

Michaël Allouche, Stéphane Girard, Emmanuel Gobet

A generative model for fBm with deep ReLU neural networks

Michaël Allouche
michael.allouche@polytechnique.edu
CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris
Route de Saclay, 91128 Palaiseau Cedex, France

Stéphane Girard
stephane.girard@inria.fr
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK
38000 Grenoble, France

Emmanuel Gobet
emmanuel.gobet@polytechnique.edu
CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris
Route de Saclay, 91128 Palaiseau Cedex, France

April 21, 2022

Abstract

We provide a large probability bound on the uniform approximation of fractional Brownian motion with Hurst parameter H, by a deep-feedforward ReLU neural network fed with a N-dimensional Gaussian vector, with bounds on the network construction (number of hidden layers and total number of neurons). Essentially, up to log terms, achieving an uniform error of $\mathcal{O}\left(N^{-H}\right)$ is possible with $\log(N)$ hidden layers and $\mathcal{O}(N)$ parameters. Our analysis relies, in the standard Brownian motion case ($H = 1/2$), on the Levy construction and in the general fractional Brownian motion case ($H \neq 1/2$), on the Lemarié-Meyer wavelet representation. This work gives theoretical support on new generative models based on neural networks for simulating continuous-time processes.

Keywords: fractional Brownian motion, Gaussian process, neural networks, generative models
MSC: 62M45, 60G15, 60G22

1 Introduction

Over last few years a new paradigm of generative model has emerged in the new machine learning community with the goal of sampling high-dimensional complex objects (such as images, videos or natural language) from a data set of these objects. If X denotes the random variable taking values in a general metric space $(\mathcal{X}, d_{\mathcal{X}})$ from which we have observations $(X_i)_{i\geq 1}$, the problem of generative model construction amounts to finding a function $G_\theta : \mathbb{R}^N \mapsto \mathcal{X}$ and a latent probability distribution μ on $\mathbb{R}^N$ such that

$$X \overset{d}{=} G_\theta(Z) \text{ and } Z \sim \mu. \tag{1}$$

Usually, the choice of the dimension N (the so-called latent dimension) is part of the problem. The function G_θ belongs to a parametric family of functions $\mathcal{G} = \{G_\theta\}_{\theta\in\Theta}$, and it is common to consider neural networks: in this work, we follow this approach. Essentially, two main questions have to be addressed to obtain a generative model: a) how to choose $\mathcal{G}$ to have a chance to get the equality in distribution (1), or at least a good approximation of it for some $G_\theta \in \mathcal{G}$? b) how to learn the parameter θ from the data set? The second question b) has been tackled by [22] in their seminal work of Generative Adversarial Network (GAN). We will not focus on that problematic in this work, there is a tremendous number of works (about 30,000 citations of [22] on Google Scholar at the date of writing this article). Instead, we are to focus on a), *i.e.* quantifying how to choose $\mathcal{G}$ and the latent space (N, μ) when $\mathcal{X}$ is the space of continuous functions indexed by time, equipped with the sup norm $d_{\mathcal{X}}$, and when the distribution of X is that of a stochastic process (infinite dimensional object), possibly non-Markovian.

Among the huge and expanding literature on GANs, lot of works studied the ability to generate time-series (in a discrete time), either in finance [43], in medicine [15] or in meteorology [24], for citing only some of them. However, to the best of our knowledge, none of them is dealing with continuous-time processes. Moreover, designing the architecture of a neural network G_θ with respect to its depth (number of hidden layers), size (number of neurons), type (feed-forward, recurrent, convolutional, etc.) and activation functions (sigmoid, ReLU, etc.), is a very difficult question and therefore often left to empirical grid search. In this work, we aim at tackling these aspects and providing precise quantitative guidelines on $\mathcal{G}$ in the case where X is a fractional Brownian motion (fBm) with Hurst parameter $H \in (0, 1)$ including standard Brownian motion $(H = 1/2)$ as a particular case.

A fBm is a centered Gaussian process with a specific covariance function [34], detailed definition and properties are given in Section 2. Remind that almost all sample paths of fBm are α-Hölder-continuous for any $\alpha < H$, see [31, Corollary 4.2] for the thorough statement. The motivation in choosing such a model for our study is threefold. First, its stochastic simulation is known to be quite delicate (at least for $H \neq 1/2$), especially when the number of time points gets larger and larger – see [5, 13] for a review and [6, 27] for recent contributions – hence having at hand a generative model for the full path is really appealing for practical use. Second, it is widely used in various real-life modelings: uni and bipedal postural standing in biomechanics [4]; volatility of financial assets [8, 19]; vortex filament structures observed in 3D fluids [17]; prices of electricity in a liberated market [3]; solar cycle [38]; for other fractional-based modeling, see [7]. Third, understanding the right design of $\mathcal{G}$ for generating the fBm distribution may well open the way to handle more complicated stochastic models written as a Stochastic Differential Equation (SDE) driven by fBm for instance: indeed, as we will see, the design of the current $\mathcal{G}$ inherits much from the time-regularity of X and this property is lifted to SDE driven by X. This part is left to further investigation.

In this work we study the required depth (number of hidden layers) and complexity (number of neurons and parameters) of a deep-feedforward neural network (NN) for $\mathcal{G}$, with a Rectified Linear Unit (ReLU) for the activation function [21, Chapter 6]: it is referred to as ReLU NN in the sequel. For the latent distribution μ, we consider N independent components and without loss of generality for the simulation purpose, each of them is taken as a standard Gaussian random variable. Essentially, our results state (Theorems 2 and 3) that for a given latent dimension N, there is a $G_\theta \in \mathcal{G}$ such that equality (1) holds with an error $N^{-H}\left(1 + \log(N)\right)^{1/2}$ in sup norm with probability $1 - p$. Moreover, focusing on the rates with respect to $N \to +\infty$, the depth of G_θ is at most

$$\mathcal{O}\left(\log N\right)$$

and its global complexity is

$$\mathcal{O}\left(N^{1+\zeta}\log N\right),$$

where ζ is a positive parameter that can be taken as small as desired, and where the $\mathcal{O}\left(.\right)$ depend on p, ζ and H. In particular for the Brownian motion $(H = 1/2)$ we can take $\zeta = 0$. A more detailed dependence on p, ζ and H is given latter.

These results are original to the best of our knowledge, and should play a key role in tuning GAN-based methods in the choice of the parametric family of NN for generating fractional stochastic processes in continuous time. These results make a clear connection between the time-regularity of the path (that could be measured on the real observed data) and the architecture of the parameterization to set up.

This work is organized as follows. In Section 2, we recall few properties of fBm. Our approximations are based on wavelet decomposition and we will provide appropriate materials. Then we state our main quantitative results about depth and complexity of deep ReLU NN for generating fBm. Section 3 is devoted to the proofs. For pedagogical and technical reasons, we start with the case $H = 1/2$ (standard Brownian motion) in Subsection 3.1; then we handle the general case of fBm in Subsection 3.2. Simulation of fBm with a NN approach is illustrated in Section 4. A few technical proofs and numerical illustrations are postponed to Appendix.

Notations: The set of naturals without zero is defined by $\mathbb{N}_0 := \{1, 2, \dots, n, \dots\}$ and $\mathbb{N} := \mathbb{N}_0 \cup \{0\}$; define the set $\mathcal{M} := \left\{2^{n+1}, n \in \mathbb{N}\right\}$; the vector of N standard Gaussian random variables $G_1, \dots, G_N$ is denoted by $G_{1:N}$; the imaginary number $\mathbf{i}^2 = -1$. We write $x = \mathcal{O}_c(y)$ if $|x| \leq c\,|y|$ for some positive constant c which, in the context where it is used, does not depend neither on the latent dimension N nor on the accuracy ε; usually y will be a not-small quantity ($y \geq 1$) as a polynomial or logarithmic function of N or/and ε^{-1} according to the context. Finally, we write $a_N \asymp b_N$ if there exists a constant $c \geq 1$ such that $\forall N \in \mathbb{N}_0$, $c^{-1} \leq a_N/b_N \leq c$.

2 Preliminaries and main results

2.1 About Fractional Brownian motion

Fractional Brownian motion (fBm) $\left\{B^H(t)\right\}_{t\in\mathbb{R}}$ with a Hurst parameter $H \in (0,1)$ is a Gaussian process, centered ($\mathbb{E}\left[B^H(t)\right] = 0$), with covariance function

$$\mathbb{C}\mathrm{ov}\left(B^H(t), B^H(s)\right) = \frac{V_H}{2}\left(|t|^{2H} + |s|^{2H} - |t-s|^{2H}\right), \text{ for any } \quad s, t \geq 0, \tag{2}$$

with $V_H = \mathbb{V}\mathrm{ar}[B^H(1)]$. We call $B^H(\cdot)$ a standard fBm if $V_H = 1$. When $H = 1/2$, we will simply write B instead of $B^{1/2}$. Our aim is to approximate the distribution of B^H on a finite interval: owing to the self-similarity property of fBm [36, Proposition 2.1], we can consider, without loss of generality, the interval $[0,1]$, which is our setting from now on.

As B^H is a centered Gaussian process in a Banach space $(\mathscr{C}^0([0,1],\mathbb{R}), \|.\|_\infty)$ (see [29, Proposition 3.6]), B^H admits almost sure (a.s.) series representation of the form

$$B^H(t) = \sum_{k=0}^{\infty} u_k(t) G_k, \quad \forall t \in [0,1], \tag{3}$$

where $\{u_k\}_{k\in\mathbb{N}}$ is a sequence of continuous non-random functions, and $\{G_k\}_{k\in\mathbb{N}}$ is a sequence of independent standard Gaussian variables $\mathcal{N}(0,1)$. Equality (3) holds in the sense that the series converges a.s. uniformly. Such representations for fBm are studied in [35] using wavelets.

Let $H \in (0,1)$; [28] showed that there exists a sequence $\{u_k\}_k$ such that the L^2-truncation error is

$$\left(\mathbb{E}\left[\sup_{t\in[0,1]}\left|\sum_{k=N}^{\infty} u_k(t) G_k\right|^2\right]\right)^{1/2} \asymp N^{-H}(1 + \log(N))^{1/2}; \tag{4}$$

in addition, the above convergence rate is optimal among all sequences $\{u_k\}_k$ for which (3) converges a.s. in sup-norm.

In [35] the authors focused on the a.s. uniform convergence on $[0,1]$ for different wavelet representations series (3) using a specific mother wavelet function ψ, and the authors of [2, Theorem 5] showed their optimality in the sense of (4). Not only ψ has to generate an orthonormal basis $\left\{\psi_{j,k}(x) = 2^{j/2}\psi\left(2^j x - k\right)\right\}_{(j,k)\in\mathbb{Z}^2}$ of $\mathbf{L^2}(\mathbb{R}, \mathrm{d}x) = \left\{f : \int_{-\infty}^{\infty} |f(x)|^2 \,\mathrm{d}x < \infty\right\}$ [33, Theorem 7.3, p. 278], but also it must respect some other regularity properties discussed hereafter.

In the following, our convention is to write the Fourier transform and its inverse as

$$\widehat{f}(\xi) := \int_{-\infty}^{\infty} f(x)\mathrm{e}^{-\mathrm{i}x\xi}\,\mathrm{d}x, \qquad f(x) := \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{f}(\xi)\mathrm{e}^{\mathrm{i}x\xi}\,\mathrm{d}\xi. \tag{5}$$

2.2 Brownian motion: wavelet representation and main result for NN generative model

A first well-known series representation is the so-called Lévy construction of the standard Brownian motion ($V_H = 1$, $H = 1/2$) obtained by plugging in (3) the basis functions

$$\psi_{j,k}^{\mathrm{FS}}(t) = 2^{j/2}\psi^{\mathrm{FS}}\left(2^j t - k\right), \qquad j \in \mathbb{N},\ k = 0, \dots, 2^j - 1, \tag{6}$$

where $\psi^{\mathrm{FS}}(x) = 2\left(x\mathbb{1}_{0\le x<1/2} + (1-x)\mathbb{1}_{1/2\le x\le 1}\right)$ is twice the antiderivative of the Haar mother wavelet [23]. The set $\{\psi_{j,k}^{\mathrm{FS}}\}_{j\in\mathbb{N},k=0,\dots,2^j-1}$ defines the Faber-Schauder (F-S) system [16, 40] and forms an orthogonal basis of $\mathbf{L^2}(\mathbb{R}, \mathrm{d}x)$. Thus, given $\{G_1, G_{j,k}\}_{j\ge 0,\ 0\le k<2^j}$ a sequence of independent standard Gaussian random variables $\mathcal{N}(0,1)$, the Lévy construction of the standard Brownian motion states that a.s. the truncated series

$$B^{(n)}(t) := G_1 t + \sum_{j=0}^{n}\sum_{k=0}^{2^j-1} 2^{-(j+1)}\psi_{j,k}^{\mathrm{FS}}(t) G_{j,k} \tag{7}$$

converges uniformly on $[0,1]$ to a Brownian motion B as $n \to \infty$ (see [41, Section 3.4]). We write $B_N := B^{(n)}$ with $N = 2^{n+1}$, to emphasize that (7) contains N scalar Gaussian random variables, which is consistent with the latent dimension discussed above. The next result quantifies the a.s. convergence rate of B_N to B, the proof is postponed to Section A.1.

Lemma 1. *Let $N \in \mathcal{M}$. Then, there exists a finite random variable $C_{(8)} \ge 0$ such that almost surely*

$$\sup_{t\in[0,1]} |B(t) - B_N(t)| \le C_{(8)} N^{-1/2}\big(1 + \log(N)\big)^{1/2}. \tag{8}$$

The above result is somehow well-known and shows that it is enough to approximate with a high probability the function $t \mapsto B_N(t)$ by a ReLU NN with suitable architecture, which is the purpose of the following statement.

Theorem 2. *Let $N \ge 2$ and $\left(\Omega^N, \mathcal{F}^N, \mathbb{P}^N\right)$ be a probability space supporting N i.i.d. standard Gaussian random variables $G_{1:N}$. Therefore, there exists an extension $(\Omega, \mathcal{F}, \mathbb{P})$ supporting a Brownian motion B such that $\forall p \in (0,1]$, there exist a ReLU neural network*

$$\tilde{B}_{N,p} : \begin{cases} \mathbb{R}^N & \to \mathscr{C}^0([0,1], \mathbb{R}) \\ G_{1:N} := (G_1, \dots, G_N) & \mapsto \tilde{B}_{N,p}(\cdot, G_{1:N}) \end{cases}$$

and a finite random variable $C \ge 0$ (independent from N and p) such that

$$\mathbb{P}\left(\sup_{t\in[0,1]} \left|B(t) - \tilde{B}_{N,p}(t, G_{1:N})\right| \le CN^{-1/2}\left(1 + \log N\right)^{1/2}\right) \ge 1 - p. \tag{9}$$

Additionally, $\tilde{B}_{N,p}$ is composed at most by

1. *$\mathcal{O}_c\left(\log\left(\frac{N\rho_N}{(1+\log N)^{1/2}}\right)\right)$ hidden layers,*
2. *$\mathcal{O}_c\left(N\log\left(\frac{N\rho_N}{(1+\log N)^{1/2}}\right)\right)$ neurons and parameters,*

with $\rho_N = -\Phi^{-1}(\frac{p}{2N})$ and Φ^{-1} the quantile function of the standard Gaussian distribution.

The proof is postponed to Subsection 3.1. The finiteness of C means only $\mathbb{P}(C < +\infty) = 1$ and a careful inspection of the proof would show that C has finite polynomial moments at any order. This will be similar for the fBM-result of Theorem 3.

Remark 1. *It is known that $\Phi^{-1}(u) \sim -\sqrt{-2\log u}$ as $u \to 0^+$, see [14]. Therefore we shall get equivalents of the architecture size, either as $p \to 0$ or as $N \to \infty$ (which results in $\rho_N \to \infty$ anyhow):*

1. *For a fixed p and as $N \to \infty$, $\rho_N/(1+\log N)^{1/2}$ tends to a constant so the depth and the complexity are respectively of order $\mathcal{O}_c(\log(N))$ and $\mathcal{O}_c(N\log N)$;*
2. *For a fixed N and as $p \to 0$, the impact on the network size is moderate since both depth and complexity are of order $\mathcal{O}_c(\log\log(1/p))$.*

As a complement to the previous marginal asymptotics, the estimates of Theorem 2 allow to have p dependent on N: for instance, building a ReLU NN with an error tolerance of order $N^{-1/2}(1+\log N)^{1/2}$ with probability $1-N^{-k}$ (for any given $k>0$) can be achieved using a depth $\mathcal{O}_c(\log N)$ and a complexity $\mathcal{O}_c(N\log N)$.

The next two remarks apply both to the current Brownian motion case and to the fBM studied in next subsection.

Remark 2. *One may wonder how to improve the rate of convergence of Theorem 2 in terms of complexity. As far as Lemma 1 is concerned, the convergence rate is optimal in the sense of [28] using a linear approximation w.r.t to the Gaussian inputs. In other words, there cannot be another fBm series expansion with a faster rate of convergence. But once we consider non-linear approximation (with spline functions for instance [9]), there is no reason that using N Gaussians, we cannot approximate (using a suitable NN) the fBm with an accuracy higher than $N^{-1/2}(1+\log(N))^{1/2}$. In particular, and even if it is already quite cheap, there is no reason for the NN depth to be of order $\log(N)$ as we propose. On the other hand, once the latent dimension N is fixed, clearly one cannot use less that $\mathcal{O}_c(N)$ parameters (no matter which NN is used), which is exactly what we propose (up to the logarithmic term).*
Finally, taking $p=0$ is not possible within our method of proof, it is an open question if another construction would allow $p=0$.

Remark 3. *From the GAN point of view, it is enough to know (like in Theorem 2) that such a ReLU NN can generate a Brownian motion without the knowledge of the NN parameters explicitly; indeed, the GAN optimization algorithm will retrieve the parameters.*
Regarding the practical use of this generative model, notice that sampling a Brownian motion path boils down to sample $G_{1:N}$ (with N independent standard Gaussian variables), and then compute $\tilde{B}_{N,p}(t, G_{1:N})$ for all times t required by the situation at hand.

2.3 Fractional Brownian motion: wavelet representation and main result for NN generative model

Among the wavelet fBm series representations proposed in [35], we will focus on the following one

$$B^H(t) = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} 2^{-jH}\left(\Psi_H\left(2^j t - k\right) - \Psi_H(-k)\right) G_{j,k}, \tag{10}$$

with

$$\widehat{\Psi}_H(\xi) := \frac{\widehat{\psi}(\xi)}{(\mathbf{i}\xi)^{H+1/2}}, \tag{11}$$

and

$$V_H = \mathbb{V}\mathrm{ar}\left[B^H(1)\right] = \frac{1}{2H\sin(\pi H)\Gamma(2H)}, \quad \text{(see Definition 2)}, \tag{12}$$

and where $\Gamma(\cdot)$ is the Gamma function. The proof of this representation is recalled in Appendix-Section B for the convenience of the reader. One choice for the wavelet ψ is the Lemarié-Meyer wavelet [30] (see [33, Equations (7.52)-(7.53)-(7.85) and Example 7.10] for more details on its construction) defined by its Fourier transform

$$\widehat{\psi^{\mathrm{M}}}(\xi) := \mathrm{e}^{-\mathbf{i}\frac{\xi}{2}} \begin{cases} \sin\left(\frac{\pi}{2}\nu\left(\frac{3|\xi|}{2\pi}-1\right)\right), & \frac{2}{3}\pi \leq |\xi| \leq \frac{4}{3}\pi, \\ \cos\left(\frac{\pi}{2}\nu\left(\frac{3|\xi|}{4\pi}-1\right)\right), & \frac{4}{3}\pi \leq |\xi| \leq \frac{8}{3}\pi, \\ 0, & \text{otherwise}, \end{cases} \tag{13}$$

where $\nu : \mathbb{R} \to [0,1]$ is a smooth function satisfying

$$\nu(x) = \begin{cases} 0, & \text{if } x \leq 0, \\ 1, & \text{if } x \geq 1, \end{cases} \quad \text{and} \quad \nu(x) + \nu(1-x) = 1. \tag{14}$$

Such properties allow to satisfy the quadrature conditions of the conjugate mirror filter [33, Subsection 7.1.3 p. 270] which specifies the scaling function in the construction of wavelet bases (see [33, Chapter 7] for a complete overview of wavelet bases analysis). Considering the truncated series of (10) over a specific set $\mathcal{I}_N$ containing at most N indices (j,k), the authors of [2, Section 5 p. 469] have shown that there exists a finite r.v. $C_{(15)} \geq 0$ such that

$$\sup_{t\in[0,1]} \left|B^H(t) - B^H_N(t)\right| \leq C_{(15)} N^{-H} \left(1 + \log(N)\right)^{1/2}. \tag{15}$$

In other words, if Ψ_H is well chosen such that ψ satisfies conditions $(\mathcal{A}_1), (\mathcal{A}_2), (\mathcal{A}_3)$ listed below then the wavelet decomposition (10) is optimal [2, Theorem 5] in the sense of (4).
Back to the construction of (13), a classical example of ν due to Daubechies [11, p. 119] is

$$\nu(x) = x^4\left(35 - 84x + 70x^2 - 20x^3\right),$$

which entails that $\widehat{\psi^{\mathrm{M}}}$ has 3 vanishing derivatives at $|\xi| = 2\pi/3,\ 4\pi/3,\ 8\pi/3$. Below, we will propose another example of ν with higher order vanishing derivatives at the boundaries in order to get (see the proof in Subsection 3.2.1) a fast decay rate of all the derivatives $(\psi^{\mathrm{M}})^{(k)}$ at infinity, which results in reducing the complexity cost of the ReLU NN architecture that we will build in Theorem 3. Note that the construction below might not be numerically optimal among the large literature on wavelets and applications in signal processing, however it is a concrete example on which we can base our theoretical result.

Construction of ψ^{M} Let $q_{\mathrm{st}(\beta)}$ be the quantile function of a Student distribution with β degrees of freedom. Thus, considering the function

$$\nu(u) := \frac{1}{1 + \exp(-q_{\mathrm{st}(\beta)}(u))}, \qquad u \in [0,1] \tag{16}$$

conditions (14) are easily satisfied. To be self content, we now briefly recall and verify $(\mathcal{A}_1), (\mathcal{A}_2), (\mathcal{A}_3)$ from [2, p. 456] to validate the use of such ν:

$(\mathcal{A}_1)$. $\left\{x \mapsto 2^{j/2}\psi(2^j x - k), j \in \mathbb{Z}, k \in \mathbb{Z}\right\}$ is an orthonomal basis of $\mathbf{L^2}(\mathbb{R}, \mathrm{d}x)$.

$(\mathcal{A}_2)$. The Fourier transform $\widehat{\psi}$ is 4 times continuously differentiable. Moreover, for any $k = 0, 1, 2, 3$ there is a constant $C > 0$ such that

$$\left|\widehat{\psi}^{(k)}(\xi)\right| \leq C(1 + |\xi|)^{-3/2}, \text{ for any } \xi \in \mathbb{R}.$$

$(\mathcal{A}_3)$. $\widehat{\psi}(\xi)$ has a zero of order 4 at $\xi = 0$ (*i.e.* for all $k = 0, 1, 2, 3$, $\widehat{\psi}^{(k)}(0) = 0$).

Condition $(\mathcal{A}_1)$ clearly holds since the wavelets $\left\{\psi^{\mathrm{M}}_{j,k}\right\}_{(j,k)\in\mathbb{Z}^2}$ generate an orthonormal basis of $\mathbf{L^2}(\mathbb{R}, \mathrm{d}x)$. Condition $(\mathcal{A}_3)$ is straightforwardly satisfied because $\widehat{\psi^{\mathrm{M}}}$ vanishes at 0. Last, consider $(\mathcal{A}_2)$: the decay of $\widehat{\psi^{\mathrm{M}}}$ and its derivatives at infinity is straightforward since it has compact support. What really needs to be checked is the smoothness property of $\widehat{\psi^{\mathrm{M}}}$: let us justify that it is $\mathscr{C}^\infty$. Observe that this follows from the tentative property $\nu^{(q)}(0^+) = \nu^{(q)}(1^-) = 0$ for all $q \in \mathbb{N}_0$. To see this, remind that the Student distribution belongs to the Fréchet maximum domain of attraction [12, Theorem. 1.2.1(1.)], therefore $q_{\mathrm{st}(\beta)}$ increases as a power function with exponent $\gamma = 1/\beta > 0$ called the tail-index. Second, for all $q \in \mathbb{N}$, $q^{(q)}_{\mathrm{st}(\beta)}(u)$ increases at most as a power function around 0 and 1, see [20, Lemma 15]. Moreover, the sigmoid function $\Delta(x) := (1+\exp(-x))^{-1}$ clearly satisfies $\Delta^{(q)}(x) = \mathcal{O}_c(\exp(-|x|))$ for $q \geq 1$. Hence, applying the Faà di Bruno formula [25, p. 224-226] for expanding the derivative of the composition of $\Delta(\cdot)$ and $q_{\mathrm{st}(\beta)}(\cdot)$ gives

$$\nu^{(q)}(u) = \sum_{l=1}^{q} \frac{1}{l!}\Delta^{(l)}\left(q_{\mathrm{st}(\beta)}(u)\right) \sum_{i_s\in\mathbb{N}_0: i_1+\cdots+i_l=q} \frac{q!}{i_1!i_2!\ldots i_l!} \prod_{s=1}^{l} q^{(i_s)}_{\mathrm{st}(\beta)}(u), \quad q \in \mathbb{N}_0,$$

which readily leads to $\nu^{(q)}(0^+) = \nu^{(q)}(1^-) = 0$ since the exponential function decays faster than any polynomials. See Figure 9 for an illustration of Ψ_H built with (13) and (16) .

We are now in a position to state our second main result.

Theorem 3. *Let $N \geq 2$ and $\left(\Omega^N, \mathcal{F}^N, \mathbb{P}^N\right)$ be a probability space supporting N i.i.d. standard Gaussian random variables $G_{1:N}$. Therefore, there exists an extension $(\Omega, \mathcal{F}, \mathbb{P})$ supporting a fractional Brownian motion B^H such that $\forall p \in (0, 1]$, for all $r \in \mathbb{N}_0$ there exist a ReLU neural network*

$$\widetilde{B}^H_{N,p} : \begin{cases} \mathbb{R}^N & \to \mathscr{C}^0([0,1], \mathbb{R}) \\ G_{1:N} := (G_1, \ldots, G_N) & \mapsto \widetilde{B}^H_{N,p}(\cdot, G_{1:N}) \end{cases}$$

and a finite random variable $C \geq 0$ (independent from N and p) such that

$$\mathbb{P}\left(\sup_{t\in[0,1]} \left|B^H(t) - \widetilde{B}^H_{N,p}(t, G_{1:N})\right| \leq CN^{-H}(1 + \log(N))^{1/2}\right) \geq 1 - p.$$

Additionally, $\widetilde{B}^H_{N,p}$ is composed by

1. $\mathcal{O}_c\left(\log\left(\frac{N\rho_N}{(1+\log(N))^{1/2}}\right)\right)$ *hidden layers,*
2. $\mathcal{O}_c\left(N^{1+\frac{H+1}{2r}} \log\left(\frac{N\rho_N}{(1+\log(N))^{1/2}}\right)\left(\frac{\rho_N}{(1+\log(N))^{1/2}}\right)^{\frac{1}{2r}}\right)$ *neurons and parameters,*

where ρ_N is defined in Theorem 2. The constants in $\mathcal{O}_c(\cdot)$ may depend on r and H.

Observe that Remarks 1-2-3 apply similarly to the above Theorem.

2.4 Discussion

In Table 1 we compare the asymptotic architecture bounds between a BM and a fBm. Note that the BM benefits from a natural construction of the F-S wavelet through ReLU functions. In comparison, the fBm construction suffers from 1) an additional approximation of the wavelet Ψ_H and 2) a larger bound on the sum over $\mathcal{I}_N$, which has only a log impact in the asymptotic NN architecture (see details in the proof in Subsection 3.2.2). Therefore, both models have the same asymptotic depth (with a constant depending on r) and a very close complexity in terms of the latent dimension N.

	BM	fBm
error tolerance (TOL)	$N^{-1/2}$	N^{-H}
depth	$\log\left(\mathtt{TOL}^{-1}\right)$	$\log\left(\mathtt{TOL}^{-1}\right)$
complexity	$\mathtt{TOL}^{-2}\log\left(\mathtt{TOL}^{-1}\right)$	$\mathtt{TOL}^{-\left(\frac{1}{H}+\zeta\right)}\log\left(\mathtt{TOL}^{-1}\right)$

Table 1: For a given confidence probability p, asymptotic complexity rates with respect to tolerance error (TOL). The parameter ζ can be taken arbitrary small, constants depending on H, p and $\zeta > 0$ are omitted.

The takeaway message from these results is that a NN with N Gaussian r.v. as inputs for approximating a process with a time regularity H (and an approximation error N^{-H} up to log-term) may have at most a depth $\log N$ and a complexity $N \log N$. Although the set $\mathcal{I}_N$ is not explicit for finding the optimal fBm NN parameters, this part can be achieved through the optimization of the GAN model with the appropriate architecture detailed in Subsection 3.2.

3 Proofs

In this section we will discuss the constructive proofs of the ReLU NN that appear in the main results. We start with the case $H = 1/2$ (standard Brownian motion) in Subsection 3.1 and we then handle the general case of fBm in Subsection 3.2. Before that, recall the output expression of a 1-hidden layer NN given some input $x \in \mathbb{R}$ and parameterized by $\theta = \left\{w_k^{(1)}, w_k^{(2)}, b_k^{(1)}, b^{(2)}\right\}_{k=1}^{K}$ is

$$\sum_{k=1}^{K} w_k^{(2)} \sigma\left(w_k^{(1)} x + b_k^{(1)}\right) + b^{(2)}, \tag{17}$$

with $\sigma(x) := \max(0, x)$ the ReLU function. Similarly, a multi-layer NN is just multiple compositions of σ with (17) between different hidden layers. For readers interested in having references on approximation properties of NN, we may refer to [32, Theorem 1 p. 70] for L_2 error using single hidden layer NN, to [37, Corollary 6.4 p. 170] for uniform approximations, and to a more recent paper [44] which has shown some uniform convergence rate for multi-layers NN.

3.1 NN representation of BM

In the following proof of Theorem 2 we will restrict to $N \in \mathcal{M} = \left\{2^{n+1}, n \in \mathbb{N}\right\}$. However note that if one wants to choose a $N \notin \mathcal{M}$, it will neither impact the error nor the complexity bounds in Theorem 2. Indeed, it suffices to take $n = \left\lfloor \frac{\log(N)}{\log(2)} - 1 \right\rfloor$ and $N' = 2^{n+1}$ such that $N' \in (\frac{N}{2}, N]$, and then set $\tilde{B}_{p,N}\left(t, G_{1:N}\right) := \tilde{B}_{p,N'}\left(t, G_{1:N'}\right)$. Regarding the error bound and complexity w.r.t. N, use those for N' by easily adjusting constants: indeed, since $N' \leq N$, it follows that $\rho_{N'} \leq \rho_N$, $\frac{1}{(1+\log(N'))} \leq$

$\frac{1}{1-\log(2)} \frac{1}{(1+\log(N))}$ for the complexity bound and $N'^{-1/2}(1+\log(N'))^{1/2} \leq \sqrt{2}N^{-1/2}(1+\log(N))^{1/2}$ for the error bound.

From now on, $N = 2^{n+1}$. For ease of notation, let

$$s_{j,k}(t) := \frac{\psi_{j,k}(t)}{2^{j/2}} = \psi\left(2^j t - k\right) \in [0,1], \tag{18}$$

be the normalized F-S wavelet, where $\psi = \psi^{\mathrm{FS}}$ in this section. Then, in view of (7) and Lemma 1, the objective is to find a ReLU NN with N standard Gaussian variables and the time t as inputs, that can approximate with uniform error and high probability

$$B_N(t) = G_1 t + \sum_{j=0}^{n} \sum_{k=0}^{2^j-1} 2^{-(j/2+1)} s_{j,k}(t) G_{j,k}. \tag{19}$$

The key advantage with the F-S wavelet (6) is that the mother wavelet ψ can be built easily with 3 ReLUs and 9 parameters such as

$$\psi(x) = 2\left(\sigma(x) - 2\sigma\left(x - \frac{1}{2}\right) + \sigma\left(x-1\right)\right). \tag{20}$$

Clearly a product operation in (19) is required between the inputs $G_{j,k}$ (*i.e.* the latent space in a GAN setting) and the normalized wavelets $s_{j,k}$ just built (see Figure 1). Since such an operation is not natively done in a feedforward network, let us study how to approximate it.

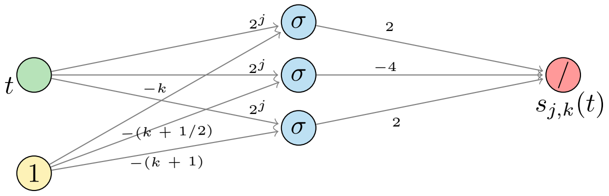


Figure 1: Neural network construction of a normalized Faber-Schauder basis function (18). The circles filled with σ represent a ReLU function, while the ones with a / represent the identity function.

3.1.1 How to make a product with a NN

Let $h(x) = x^2$. The key observation in [44, Proposition 2] based on [42] is that h can be approximated by piece-wise linear interpolation

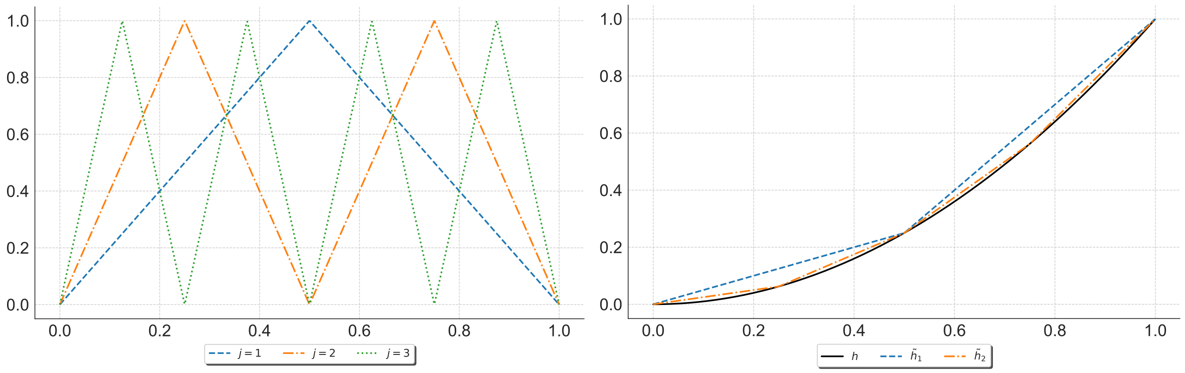
$$\widetilde{h}_\ell(x) = x - \sum_{j=1}^{\ell} \frac{\psi^{[\circ j]}(x)}{2^{2j}}, \tag{21}$$

with

$$\psi^{[\circ j]}(x) := \underbrace{\psi \circ \cdots \circ \psi}_{j \text{ times}}(x), \tag{22}$$

such that

$$\sup_{x\in[0,1]} \left| h(x) - \widetilde{h}_\ell(x) \right| = 2^{-2(\ell+1)}.$$



(a)

(b)

Figure 2: Plot of $\psi^{[\circ j]}$ for $j = \{1, 2, 3\}$ (a); approximation of $h(x) = x^2$ with $\widetilde{h}_\ell$ for $\ell = \{1, 2\}$

Expression (21) can be interpreted as a NN approximation with ℓ hidden layers, where each composition in (22) is just the sum of all translated positions of a F-S wavelet, *i.e.* $\forall j \geq 0, \psi^{[\circ j+1]}(x) = \sum_{k=0}^{2^j-1} \psi\left(2^j x - k\right)$. Therefore, instead of making a linear combination of such functions built through a long single hidden layer, the benefit of increasing the depth of the network allows to increase at a geometric rate the number of wavelets and to reduce the complexity cost from $3 \times 2^{\ell-1}$ to 3ℓ neurons.

Additionally, one shall be aware that $\widetilde{h}_\ell$ does approximate the square function only inside the interval $[0, 1]$ (see Figure 2b). Therefore we introduce a new function

$$\overset{\vee}{h}_\ell(x) = \widetilde{h}_\ell\left(|x|\right) = \widetilde{h}_\ell\left(\sigma(x) + \sigma(-x)\right), \tag{23}$$

which applies a ReLU absolute value on the input. Obviously $\overset{\vee}{h}_\ell$ extends the approximation of h on $[-1, 1]$ such that

$$\sup_{|x|\leq 1} \left| h(x) - \overset{\vee}{h}_\ell(x) \right| = 2^{-2(\ell+1)}. \tag{24}$$

The NN construction of (23) requires $\ell + 1$ hidden layers and $\mathcal{O}_c\left(\ell\right)$ neurons and parameters (see Figure 3).

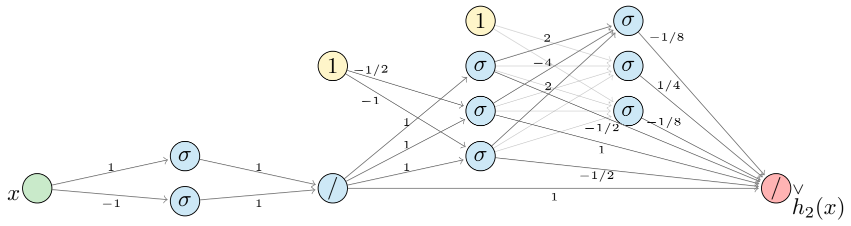


Figure 3: Neural network architecture of $\overset{\vee}{h}_\ell$ with $\ell = 2$. Lighter arrows refer to similar parameters which can easily be inferred from (20). For implementation purpose, one can obviously bypass the identity function in the middle of the network which is put here for the sake of clarity.

Once the square operation is approximately synthetized through a ReLU NN, we can leverage the polarization identity to get the product operation $(x, y) \mapsto xy$. Because the above approximation (24) is valid only on the interval $[-1, 1]$, it is useful to use a polarization identity with some flexible rescalings of x and y. It writes, for any $a, b > 0$,

$$xy = ab\left(-\left(\frac{x}{2a} - \frac{y}{2b}\right)^2 + \left(\frac{x}{2a} + \frac{y}{2b}\right)^2\right).$$

The following Proposition provides a uniform error bound on the approximation of the product with a ReLU NN.

Proposition 4. *Let $k(x, y) := xy$. Then, for any $\ell \in \mathbb{N}_0$, for given $a > 0$ and $b > 0$,*

1. *there exists a NN $\widetilde{k}_\ell^{a,b} : \mathbb{R}^2 \to \mathbb{R}$ with $\ell + 1$ hidden layers such that*

$$\sup_{x,y:|x|\le a,|y|\le b} \left|k(x, y) - \widetilde{k}_\ell^{a,b}(x, y)\right| \le ab2^{-(2\ell+1)}; \tag{25}$$

2. *if $x = 0$ or $y = 0$, then $\widetilde{k}_\ell^{a,b}(x, y) = 0$;*
3. *the ReLU NN $\widetilde{k}_\ell^{a,b}$ can be implemented with no more than $\mathcal{O}_c(\ell)$ complexity and a depth $\ell+1 := \left\lceil \frac{1}{2\log(2)} \log\left(\frac{ab}{\varepsilon}\right) - \frac{1}{2}\right\rceil + 1$, where ε is the error tolerance in sup norm.*

Proof. It is enough to set

$$\widetilde{k}_\ell^{a,b}(x, y) := ab\left(-\overset{\vee}{h}_\ell\left(\frac{x}{2a} - \frac{y}{2b}\right) + \overset{\vee}{h}_\ell\left(\frac{x}{2a} + \frac{y}{2b}\right)\right)$$

and to apply (24), while observing that when $|x| \le a$ and $|y| \le b$, $\frac{x}{2a} \pm \frac{y}{2b} \in [-1, 1]$. □

3.1.2 Final approximation of B_N

Based on Proposition 4, it seems that we can deduce a uniform bound on the product $s_{j,k}(t)G_{j,k}$ by a linear combination of composition functions of ReLUs (*i.e.* a multi-layer NN). Nevertheless, recall that (25) only holds for $|x| \le a$ and $|y| \le b$. Thus, although it is clear from (18) that for all $t \in [0, 1]$ we have $s_{j,k}(t) \in [0, 1]$, the random variables $G_{j,k}$ need however to be bounded in order to use Proposition 4: it can be made only with some probability.

Proposition 5. *Let $N \in \mathbb{N}_0$ and $p \in (0, 1]$, set*

$$\rho_N = -\Phi^{-1}\left(\frac{p}{2N}\right) \ge 0,$$

with Φ^{-1} the quantile function of the standard Gaussian distribution, and let $G_{1:N}$ be i.i.d. standard Gaussian random variables. Then

$$\mathbb{P}\left(\forall i = 1, \dots, N : |G_i| \le \rho_N\right) \ge 1 - p.$$

Proof. Clearly, the probability on the above left hand side equals

$$1 - \mathbb{P}\left(\bigcup_{i=1}^{N} \left\{ |G_i| \ge \rho_N \right\}\right) \ge 1 - 2N\Phi\left(-\rho_N\right) = 1 - p.$$

□

Therefore combining Propositions 4 and 5 with $a = 1$ and $b = \rho_N$, we can define

$$\widetilde{k}_\ell^{1,\rho_N}(s_{j,k}(t), G_{j,k}) := \rho_N\left(-\overset{\vee}{h}_\ell\left(\frac{s_{j,k}(t)}{2} - \frac{G_{j,k}}{2\rho_N}\right) + \overset{\vee}{h}_\ell\left(\frac{s_{j,k}(t)}{2} + \frac{G_{j,k}}{2\rho_N}\right)\right), \tag{26}$$

which can be implemented with $\ell + 2$ hidden layers (since we need an additional one to build the $s_{j,k}$) and $\mathcal{O}_c(\ell)$ neurons and parameters (see Figure 4).

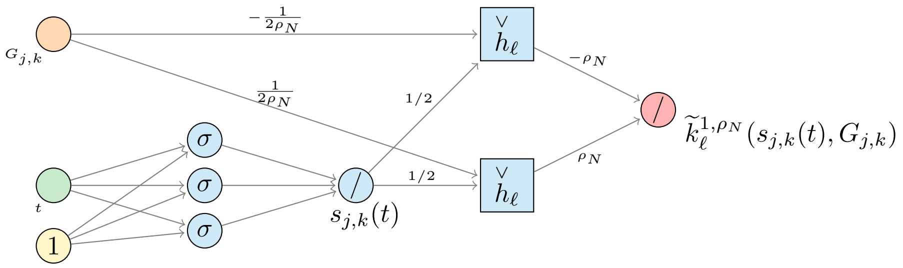


Figure 4: Neural network architecture of $\widetilde{k}_\ell^{1,\rho_N}(s_{j,k}(t),G_{j,k})$.

Remark 4. *A slight advantage of* (26) *over the polarization identity in [44, Equation (4) p. 106] is that it only requires two square approximations instead of three.*

Let $\widetilde{B}_N$ be the NN approximation of (19) such that

$$\widetilde{B}_N(t)=\widetilde{k}_\ell^{1,\rho_N}(t,G_1)+\sum_{j=0}^{n}\sum_{k=0}^{2^j-1}2^{-(j/2+1)}\widetilde{k}_\ell^{1,\rho_N}\left(s_{j,k}(t),G_{j,k}\right), \tag{27}$$

with $\widetilde{k}_\ell^{1,\rho_N}$ defined in (26). Therefore, on the event $\{|G_i|\le\rho_N : i=1,\dots,N\}$ which has a probability greater than $1-p$, one has

$$\begin{aligned}\sup_{t\in[0,1]}\left|B_N(t)-\widetilde{B}_N(t)\right| &\le \sup_{t\in[0,1]}\left|tG_1-\widetilde{k}_\ell^{1,\rho_N}\left(t,G_1\right)\right| \\ &\quad+\sup_{t\in[0,1]}\left|\sum_{j=0}^{n}\sum_{k=0}^{2^j-1}2^{-(j/2+1)}\left(s_{j,k}(t)G_{j,k}-\widetilde{k}_\ell^{1,\rho_N}\left(s_{j,k}(t),G_{j,k}\right)\right)\right|\end{aligned}$$

(from Proposition 4 and since $s_{j,k}(t)\in[0,1]$ and $|G_{j,k}|\le\rho_N$)

$$\begin{aligned}&\le \rho_N 2^{-(2\ell+1)}\left(1+\sum_{j=0}^{n}\sum_{k=0}^{2^j-1}2^{-(j/2+1)}\right)\\ &=\rho_N 2^{-(2\ell+1)}\left(1+\frac{N^{1/2}-1}{2\left(\sqrt{2}-1\right)}\right)\quad\text{(recall that } N=2^{n+1}\text{)}\\ &\le \rho_N 2^{-2\ell}N^{1/2}.\end{aligned}$$

Hence, with probability at least $1-p$, combining Lemma 1 with the above yields

$$\sup_{t\in[0,1]}\left|B(t)-\widetilde{B}_N(t)\right|\le C_{(8)}N^{-1/2}\big(1+\log\left(N\right)\big)^{1/2}+\rho_N 2^{-2\ell}N^{1/2}.$$

It follows that, if we choose

$$\ell=\left\lceil\frac{1}{2\log(2)}\log\left(\frac{N\rho_N}{\big(1+\log\left(N\right)\big)^{1/2}}\right)\right\rceil\vee 1, \tag{28}$$

then (9) is proved with $\widetilde{B}_{N,p}(t,G_{1:N}):=\widetilde{B}_N(t)$. All in all, based on Figure 4, the architecture required for the N products in (27), *i.e.* N sub-networks, yields a total of at most $\ell+2$ hidden layers and a complexity $\mathcal{O}_c\left(N\ell\right)$. Replacing with (28) gives the stated bounds. □

3.2 NN representation of fBm

Now that we are acquainted with the case of BM, we can move on to the more general case which requires additional arguments. In view of (15) with a fixed $\gamma > 0$, the goal here is to prove that there exists a ReLU NN approximating uniformly

$$B_N^H(t) = \sum_{(j,k)\in\mathcal{I}_N} 2^{-jH}\left(\Psi_H\left(2^j t - k\right) - \Psi_H(-k)\right) G_{j,k}, \tag{29}$$

with $\mathrm{Card}\,(\mathcal{I}_N) \leq N$ and $G_{j,k} \sim \mathcal{N}(0,1)$. The proof will be composed in two parts. First we will discuss how Ψ_H can be approximated by ReLU basis functions in $\mathbb{R}$. Second, we will see how to control the error on the product with Gaussians in (29). In this section we will write $\psi^{\mathrm{M}}(\cdot) = \psi(\cdot)$ for the Lemarié-Meyer wavelet (13) with $\nu(\cdot)$ as in (16).

3.2.1 Approximation of Ψ_H

We want to show that for all $\varepsilon \in (0,1)$ there exists a ReLU NN $\widetilde{g}$ such that

$$\sup_{u\in\mathbb{R}} |\Psi_H(u) - \widetilde{g}(u)| \leq \varepsilon. \tag{30}$$

Note that we cannot apply the universal approximation theorem [10, Theorem 1] which holds for continuous functions with compact support. To tackle the infinite support, the strategy will consist of first approximating Ψ_H in some interval $[-u_{\max}, u_{\max}]$, and then using the fast decay rate of $|\Psi_H(u)|$ for $|u| > u_{\max}$. Indeed, since by construction $\widehat{\psi}$ and its derivatives vanish in the neighborhood of $\xi = 0$, $\widehat{\Psi}_H$ defined in (11) is $\mathscr{C}^\infty$ with compact support for any parameter $H \in \mathbb{R}$. So for all $(m,q) \in \mathbb{N}^2$, we claim that

$$\left|\Psi_H^{(q)}(u)\right| \leq \frac{C_{H-q,m}}{1+|u|^{m+1}}, \tag{31}$$

where $C_{H-q,m}$ is a constant depending on $H-q$ and m. The property for $q=0$ is clear: use the inverse Fourier transform and $m+1$ integration by parts, taking advantage that the derivatives of $\widehat{\Psi}_H$ vanish at the boundary of its support (see discussion after (16)). For $q \neq 0$, observe that $\Psi_H^{(q)}(u) = \Psi_{H-q}(u)$ and the property follows. Now we proceed to (30), by following the ideas of [44, Theorem 1] with some variations. In (31), we have a degree of freedom with the choice of the parameter m, it will be fixed at the end of the proof.

Consider a uniform grid of M points $\{u_i = (i-1)\delta - u_{\max}\}_{i=1}^M$ with $M > 1$ and $\delta = \frac{2u_{\max}}{M-1}$ on the domain $[-u_{\max},\, u_{\max}]$, assuming $\delta \leq 1/2$. The parameter $u_{\max} > 0$ will be fixed later. Additionally, for $i = 1, \dots, M$, we define a triangular function

$$\phi_i(u) := \phi\left(\frac{u-u_i}{\delta}\right),$$

where

$$\phi(t) := \sigma(t+1) + \sigma(t-1) - 2\sigma(t),$$

and with the following (obvious) properties:

1. $\phi_i(\cdot)$ is symmetric around u_i,
2. $\sup_{u\in\mathbb{R}} |\phi_i(u)| = \phi_i(u_i) = 1$,
3. $\mathtt{supp}(\phi_i) \in [u_i - \delta,\, u_i + \delta]$,
4. $\sum_{i=1}^M \phi_i(u) \equiv 1, \quad$ for $u \in [-u_{\max},\, u_{\max}]$.

The function ϕ_i is nothing else than another FS wavelet $\psi_{j,k}^{\mathrm{FS}}$ with slightly different scaling and position parameters. Now let $r \in \mathbb{N}_0$ and consider a localized Taylor polynomial function

$$g_1(u) := \sum_{i=1}^{M} \phi_i(u) P_i(u), \tag{32}$$

where P_i is the Taylor polynomial of degree $(r-1)$ of $\Psi_H \in \mathscr{C}^\infty$ at the point u_i given by

$$P_i(u) := \sum_{q=0}^{r-1} \frac{\Psi_H^{(q)}(u_i)}{q!} (u-u_i)^q.$$

To approximate the q-power function, we will need the following result.

Proposition 6. *Let $\ell \in N_0$, $a > 0$ and $b > 0$. For any $q \in \mathbb{N}$, define recursively the ReLU NN with at most $(q-1)(\ell+1)$ hidden layers by*

$$y \mapsto \ \widetilde{y^q} := \widetilde{k}_\ell^{b,b_q}\left(y, \widetilde{y^{q-1}}\right), \quad q \geq 2,$$

with by convention $\widetilde{y^0} := 1$, $\widetilde{y^1} := y$, where $\widetilde{k}_\ell^{a,b}$ is defined in Proposition 4 and where

$$b_q := b^{q-1}\left(1 + 2^{-(2\ell+1)}\right)^{q-2}. \tag{33}$$

It is such that

$$\sup_{y:|y|\leq b} \left| y^q - \widetilde{y^q} \right| \leq b^q \left(\left(1 + 2^{-(2\ell+1)}\right)^{q-1} - 1 \right), \tag{34}$$

$$\sup_{x,y:|x|\leq a, |y|\leq b} \left| xy^q - \widetilde{k}_\ell^{a,b_{q+1}}\left(x, \widetilde{y^q}\right) \right| \leq ab^q \left(\left(1 + 2^{-(2\ell+1)}\right)^{q} - 1 \right). \tag{35}$$

Proof. We set $\eta := 2^{-(2\ell+1)}$ and we proceed by induction. Inequality (34) holds for $q = 2$ thanks to Proposition 4. Now take $q \geq 3$, assume (34) holds for $q-1$. Clearly, this implies

$$\sup_{|y|\leq b} \left| \widetilde{y^{q-1}} \right| \leq b^{q-1}(1+\eta)^{q-2} = b_q. \tag{36}$$

Therefore,

$$\begin{aligned}
\sup_{|y|\leq b} \left| y^q - \widetilde{y^q} \right| &\leq \sup_{|y|\leq b} \left| y^q - y\ \widetilde{y^{q-1}} \right| + \sup_{|y|\leq b} \left| y\ \widetilde{y^{q-1}} - \widetilde{y^q} \right| \\
&\leq b \sup_{|y|\leq b} \left| y^{q-1} - \widetilde{y^{q-1}} \right| + \sup_{|y|\leq b} \left| y\ \widetilde{y^{q-1}} - \widetilde{k}_\ell^{b,b_q}\left(y, \widetilde{y^{q-1}}\right) \right| \\
&\leq bb^{q-1}\left((1+\eta)^{q-2} - 1\right) + bb^{q-1}(1+\eta)^{q-2}\eta \\
&= b^q\left((1+\eta)^{q-1} - 1\right)
\end{aligned}$$

where, at the last inequality, we have used Proposition 4 combined with bound (36). We are done with (34). Similarly for (35), we get

$$\sup_{|x|\leq a, |y|\leq b} \left| xy^q - \widetilde{k}_\ell^{a,b_{q+1}}\left(x, \widetilde{y^q}\right) \right| \leq a \sup_{|y|\leq b} \left| y^q - \widetilde{y^q} \right| + \sup_{|x|\leq a, |y|\leq b} \left| x\ \widetilde{y^q} - \widetilde{k}_\ell^{a,b_{q+1}}\left(x, \widetilde{y^q}\right) \right|.$$

Combining (34) and Proposition 4 with (36), we get (35). □

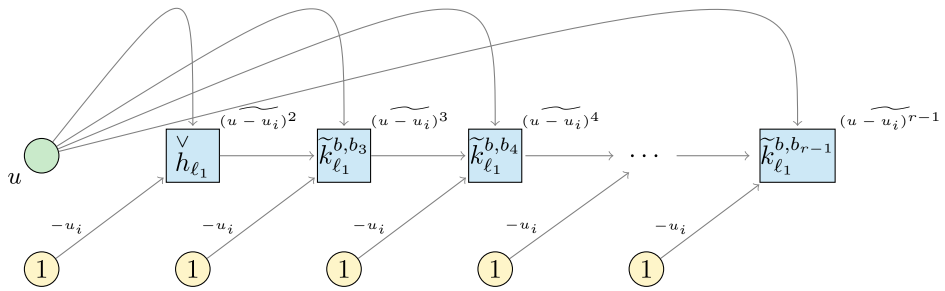

u
$\overset{\vee}{h}_{\ell_1}$
$\widetilde{(u-u_i)^2}$
$\widetilde{k}^{b,b_3}_{\ell_1}$
$\widetilde{(u-u_i)^3}$
$\widetilde{k}^{b,b_4}_{\ell_1}$
$\widetilde{(u-u_i)^4}$
$\cdots$
$\widetilde{k}^{b,b_{r-1}}_{\ell_1}$
$\widetilde{(u-u_i)^{r-1}}$
$-u_i$
$-u_i$
$-u_i$
$-u_i$
$-u_i$
1
1
1
1
1

Figure 5: Neural network architecture of all power functions of $(u-u_i)^q$ for $q \in \{2, \dots, r-1\}$ with $b = 2\delta$ and b_q defined in (33).

We are now in a position to prove (30). Given the support property of ϕ_i, the strategy consists of splitting the error approximation in three terms:

1. A classical Taylor bound on the main interval yields

$$\begin{aligned}\sup_{|u|\le u_{\max}} |\Psi_H(u) - g_1(u)| &= \sup_{|u|\le u_{\max}} \left|\sum_{i=1}^{M} \phi_i(u)\left(\Psi_H(u) - P_i(u)\right)\right| \\ &\le 2 \max_{i=1,\dots,M} \sup_{u\in\mathsf{supp}(\phi_i)} |\Psi_H(u) - P_i(u)|\end{aligned}$$

since u is in the support of at most two ϕ_i's and $|\phi_i(u)| \le 1$,

$$\begin{aligned}&\le 2 \max_{i=1,\dots,M} \sup_{u\in\mathsf{supp}(\phi_i)} \frac{\left|\Psi_H^{(r)}(u)\right|}{r!}(2\delta)^r \\ &\le \frac{2}{r!} C_{H-r,m}(2\delta)^r,\end{aligned}$$

using (31). Let $\widetilde{g_{i,q}}(\cdot)$ be the ReLU NN approximation of $u \mapsto \phi_i(u)(u-u_i)^q$ using (35) with $|\phi_i(u)| \le 1 = a$ and $|u-u_i| \le 2\delta = b$ (see Figure 5). In view of (32), set

$$\widetilde{g}(u) := \sum_{q=0}^{r-1} \frac{1}{q!} \sum_{i=1}^{M} \Psi_H^{(q)}(u_i)\widetilde{g_{i,q}}(u). \tag{37}$$

Observe, from the second statement of Proposition 4 that $\widetilde{g_{i,q}}(u) = 0$ for $u \notin \mathsf{supp}(\phi_i)$. So using (35) (setting $\eta := 2^{-(2\ell_1+1)}$ with $\ell_1 \in \mathbb{N}_0$) leads to

$$\begin{aligned}\sup_{|u|\le u_{\max}} |g_1(u) - \widetilde{g}(u)| &\le \sum_{q=0}^{r-1} \sup_{|u|\le u_{\max}} \frac{\left|\Psi_H^{(q)}(u)\right|}{q!} \sum_{i=1}^{M} \sup_{|u|\le u_{\max}} |\phi_i(u)(u-u_i)^q - \widetilde{g_{i,q}}(u)| \\ &\le 2r \max_{q=0,\dots,r-1} C_{H-q,m} \sum_{q=1}^{\infty} (2\delta)^q \frac{((1+\eta)^q - 1)}{q!}.\end{aligned} \tag{38}$$

Using that $2\delta \le 1$ and $\eta \le 1/2$, we easily get that the above right hand side is bounded by $r\ \delta\ \eta\ C_e \max_{q=0,\dots,r-1} C_{H-q,m}$ for some universal constant C_e. To sum up, we have proved

$$\sup_{|u|\le u_{\max}} |\Psi_H(u) - \widetilde{g}(u)| \le \mathcal{C}_{H,r,m}\left(\delta^r + \delta\eta\right)$$

where, here and in what follows, $\mathcal{C}_{H,r,m}$ stands for a finite positive constant depending on H, r, m, which value may change from line to line, without changing its name. By taking

$$\delta = \left(\frac{\varepsilon}{6\mathcal{C}_{H,r,m}}\right)^{\frac{1}{r}} \wedge \frac{1}{2} = \mathcal{O}_c\left(\varepsilon^{\frac{1}{r}}\right), \qquad \eta \leq \frac{\varepsilon}{6\mathcal{C}_{H,r,m}\delta} = \mathcal{O}_c\left(\varepsilon^{1-\frac{1}{r}}\right), \tag{39}$$

we have

$$\sup_{|u|\leq u_{\max}} |\Psi_H(u) - \widetilde{g}(u)| \leq \frac{\varepsilon}{3}.$$

The condition on η≤ $1/2$ is satisfied for

$$\ell_1 = \left\lceil \frac{1}{2\log(2)} \log\left(\frac{6C_{H-r,m}\delta}{\varepsilon}\right) - \frac{1}{2} \right\rceil \vee 1 = \mathcal{O}_c\left(\log\left(\varepsilon^{-1}\right)\right). \tag{40}$$

2. Focusing on the small interval $|u| \in [u_{\max}, u_{\max}+\delta]$ where u belongs to $\mathtt{supp}(\phi_M)$ only, write

$$\begin{aligned}
&\sup_{|u|\in[u_{\max},u_{\max}+\delta]} |\Psi_H(u) - g_1(u)| \\
&\quad = \sup_{|u|\in[u_{\max},u_{\max}+\delta]} |\Psi_H(u) - \phi_M(u)P_M(u)| \\
&\quad \leq \sup_{|u|\in[u_{\max},u_{\max}+\delta]} |\Psi_H(u)| + \sup_{|u|\in[u_{\max},u_{\max}+\delta]} |\Psi_H(u) - P_M(u)| \\
&\quad \underset{(31)}{\leq} \frac{C_{H,m}}{1+u_{\max}^{m+1}} + \frac{C_{H-r,m}}{1+u_{\max}^{m+1}} \frac{\delta^r}{r!} \\
&\quad \underset{(39)}{\leq} \frac{\mathcal{C}_{H,r,m}}{1+u_{\max}^{m+1}}.
\end{aligned}$$

Similarly to bound (38) but taking advantage of the fast decay of $\sup_{u\in[u_{\max},u_{\max}+\delta]}\left|\Psi_H^{(q)}(u)\right|$ yields

$$\sup_{|u|\in[u_{\max},u_{\max}+\delta]} |g_1(u) - \widetilde{g}(u)| \leq \mathcal{C}_{H,r,m} \frac{\delta\ \eta}{1+u_{\max}^{m+1}}.$$

All in all, and using $\delta\ \eta \leq 1/4$,

$$\sup_{|u|\in[u_{\max},u_{\max}+\delta]} |\Psi_H(u) - \widetilde{g}(u)| \leq \frac{\mathcal{C}_{H,r,m}}{1+u_{\max}^{m+1}} \leq \frac{\varepsilon}{3}$$

for a new constant $\mathcal{C}_{H,r,m}$ and with the choice

$$u_{\max} := \left(\frac{3\mathcal{C}_{H,r,m}}{\varepsilon}\right)^{\frac{1}{m+1}}.$$

3. Finally, on the last interval $|u| \in [u_{\max}+\delta, +\infty)$, both $\widetilde{g}(\cdot)$ and $g_1(\cdot)$ vanish, and from (31), we readily get

$$\sup_{|u|\in[u_{\max}+\delta,+\infty)} |\Psi_H(u) - \widetilde{g}(u)| \leq \frac{C_{H,m}}{1+u_{\max}^{m+1}} \leq \frac{\varepsilon}{3}.$$

All in all, (30) is proved with the ReLU NN (37). Collecting previous asymptotics, we get

$$M = \frac{2u_{\max}}{\delta} + 1 = \mathcal{O}_c\left(\varepsilon^{-\frac{1}{m+1}}\varepsilon^{-\frac{1}{r}}\right). \tag{41}$$

3.2.2 Error control including Gaussian random variables

We are back to the approximation of (29). For $(j,k) \in \mathcal{I}_N$ we set

$$Y_{j,k}(t) := \varPsi_H\left(2^j t - k\right) - \varPsi_H(-k) \quad \text{ and } \quad \widetilde{Y}_{j,k}(t) := \widetilde{g}\left(2^j t - k\right) - \widetilde{g}(-k)$$

for its ReLU NN approximation. In view of (29), let us derive an error bound of the product $Y_{j,k}(t)G_{j,k}$ for $t \in [0,1]$ and $G_{j,k}$ a standard Gaussian random variable. From (30) with $\varepsilon \le 1$ and (31), we get

$$\sup_{t\in[0,1]}\left|\widetilde{Y}_{j,k}(t)\right| \vee \sup_{t\in[0,1]}\left|Y_{j,k}(t)\right| \le 2\varepsilon + 2\sup_{u\in\mathbb{R}}\left|\varPsi_H(u)\right| \le 2\left(1 + C_{H,m}\right) =: \bar{C}_H. \tag{42}$$

Similarly to (26), we can rewrite for $t \in [0,1]$ and $(j,k) \in \mathcal{I}_N$ the NN product approximation of $\widetilde{Y}_{j,k}(t)G_{j,k}$ with $\ell_2 \in \mathbb{N}_0$ as

$$\widetilde{k}_{\ell_2}^{\bar{C}_H,\rho_N}\left(\widetilde{Y}_{j,k}(t), G_{j,k}\right) = \bar{C}_H\rho_N\left(-\overset{\vee}{h}_{\ell_2}\left(\frac{\widetilde{Y}_{j,k}(t)}{2\bar{C}_H} - \frac{G_{j,k}}{2\rho_N}\right) + \overset{\vee}{h}_{\ell_2}\left(\frac{\widetilde{Y}_{j,k}(t)}{2\bar{C}_H} + \frac{G_{j,k}}{2\rho_N}\right)\right). \tag{43}$$

Let us work on the event $\{|G_{j,k}| \le \rho_N : (j,k) \in \mathcal{I}_N\}$ which has a probability greater than $1-p$ and let us focus on the approximation error of the first term on the right-hand side of (43):

$$\begin{aligned}
& \sup_{t\in[0,1]}\left|\overset{\vee}{h}_{\ell_2}\left(\frac{\widetilde{Y}_{j,k}(t)}{2\bar{C}_H} - \frac{G_{j,k}}{2\rho_N}\right) - \left(\frac{Y_{j,k}(t)}{2\bar{C}_H} - \frac{G_{j,k}}{2\rho_N}\right)^2\right| \\
\le \quad & \sup_{t\in[0,1]}\left|\overset{\vee}{h}_{\ell_2}\left(\frac{\widetilde{Y}_{j,k}(t)}{2\bar{C}_H} - \frac{G_{j,k}}{2\rho_N}\right) - \left(\frac{\widetilde{Y}_{j,k}(t)}{2\bar{C}_H} - \frac{G_{j,k}}{2\rho_N}\right)^2\right| \\
+ \quad & \sup_{t\in[0,1]}\left|\left(\frac{\widetilde{Y}_{j,k}(t)}{2\bar{C}_H} - \frac{G_{j,k}}{2\rho_N}\right)^2 - \left(\frac{Y_{j,k}(t)}{2\bar{C}_H} - \frac{G_{j,k}}{2\rho_N}\right)^2\right| \\
\underset{(24)}{\le} \quad & 2^{-2(\ell_2+1)} + \frac{\sup_{t\in[0,1]}\left|\widetilde{Y}_{j,k}(t) - Y_{j,k}(t)\right|}{2\bar{C}_H} \\
& \qquad \times \left(\frac{\sup_{t\in[0,1]}\left|\widetilde{Y}_{j,k}(t)\right| + \sup_{t\in[0,1]}\left|Y_{j,k}(t)\right|}{2\bar{C}_H} + \frac{|G_{j,k}|}{\rho_N}\right) \\
\underset{(30),(42)}{\le} \quad & 2^{-2(\ell_2+1)} + \frac{2\varepsilon}{\bar{C}_H}.
\end{aligned}$$

So replacing in (43) and similarly for the second term, it entails

$$\sup_{(j,k)\in\mathcal{I}_N}\sup_{t\in[0,1]}\left|Y_{j,k}(t)G_{j,k} - \widetilde{k}_{\ell_2}^{\bar{C}_H,\rho_N}\left(\widetilde{Y}_{j,k}(t), G_{j,k}\right)\right| \le 2^{-(2\ell_2+1)}\bar{C}_H\rho_N + 4\varepsilon\rho_N.$$

For the final ReLU NN approximation of (29), define $\widetilde{B}_N^H$ as

$$\widetilde{B}_N^H(t) := \sum_{(j,k)\in\mathcal{I}_N} 2^{-jH}\widetilde{k}_{\ell_2}^{\bar{C}_H,\rho_N}\left(\widetilde{g}\left(2^j t - k\right) - \widetilde{g}(-k), G_{j,k}\right). \tag{44}$$

Combining (15),(29),(44) gives (still on the event $\{|G_{j,k}| \le \rho_N : (j,k) \in \mathcal{I}_N\}$)

$$\begin{aligned}
\sup_{t\in[0,1]}\left|B^H(t) - \widetilde{B}_N^H(t)\right| &\le \sup_{t\in[0,1]}\left|B^H(t) - B_N^H(t)\right| + \sup_{t\in[0,1]}\left|B_N^H(t) - \widetilde{B}_N^H(t)\right| \\
&\le C_{(15)}N^{-H}\left(1 + \log(N)\right)^{1/2} + N\left(2^{-(2\ell_2+1)}\bar{C}_H\rho_N + 4\varepsilon\rho_N\right)
\end{aligned}$$

recalling that $\mathrm{Card}\,(\mathcal{I}_N) \leq N$. It suffices to ensure that the second term at the right-hand side is bounded by $2N^{-H}\,(1+\log(N))^{1/2}$ thanks to the choices

$$\varepsilon = \frac{(1+\log(N))^{1/2}}{4\rho_N N^{H+1}} \wedge 1, \tag{45}$$

$$\ell_2 = \left\lceil \frac{1}{2\log(2)} \log\left(\frac{\bar{C}_H \rho_N N^{H+1}}{(1+\log(N))^{1/2}}\right)\right\rceil \vee 1 = \mathcal{O}_c\left(\log\left(\varepsilon^{-1}\right)\right). \tag{46}$$

3.2.3 Architecture

The total architecture of $\widetilde{g}$ is composed by M sub-networks, where each $\widetilde{g_{i,q}}$ is built as a cascade of q NN with (ℓ_1+1) hidden layers, *i.e.* $(q-1)$ NN from Proposition 6 and 1 more from the product with ϕ_i. Therefore, $\widetilde{g}$ requires at most a depth $\mathcal{O}_c\,(\ell_1)$ and a complexity $\mathcal{O}_c\,(M\ell_1)$, with constants clearly depending on r. Using M in (41) and ℓ_1 in (40), we get the architecture bounds as a function of the accuracy ε, for just one approximation of Ψ_H with $\widetilde{g}$.

As mentioned above, $\widetilde{Y}_{j,k}$ is composed of $2\widetilde{g}$ NN and so it has the same depth but twice the complexity (number of neurons and parameters) of $\widetilde{g}$. Additionally, (44) requires (ℓ_2+1) hidden layers to perform the multiplications with the Gaussian random variables. Finally, all these operations are computed for N different scaling/transition parameters (j,k). All in all, the total architecture of $\widetilde{B}_N^H$ (see Figure 6) is composed of at most

1. $\mathcal{O}_c\,(\ell_1+\ell_2)$ hidden layers,
2. $\mathcal{O}_c\left(N\Big(M\ell_1+\ell_2\Big)\right)$ neurons and parameters.

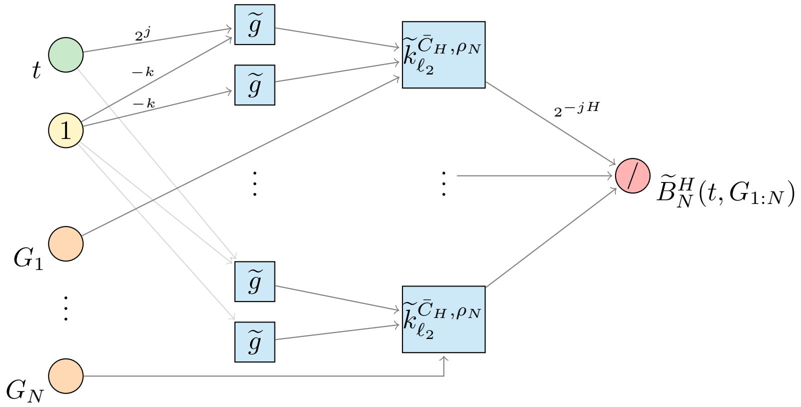

t
1
G_1
G_N
2^j
$-k$
$-k$
$\widetilde{g}$
$\widetilde{g}$
$\widetilde{g}$
$\widetilde{g}$
$\widetilde{k}_{\ell_2}^{\bar{C}_H,\rho_N}$
$\widetilde{k}_{\ell_2}^{\bar{C}_H,\rho_N}$
2^{-jH}
$\widetilde{B}_N^H(t, G_{1:N})$

Figure 6: Neural network architecture of $\widetilde{B}_N^H(t, G_{1:N})$.

Replacing with (39), (41) and (46) gives the architecture bounds with respect to ε, *i.e.*

1. hidden layers:

$$\mathcal{O}_c\,(\ell_1+\ell_2) = \mathcal{O}_c\left(\log\left(\varepsilon^{-(1-\frac{1}{r})}\right) + \log\left(\varepsilon^{-1}\right)\right) = \mathcal{O}_c\left(\log\left(\varepsilon^{-1}\right)\right),$$

where we have observed that the exponent inside the log term can be put in the $\mathcal{O}_c$ since the constants are allowed to depend on r in our notation;

2. neurons and parameters:

$$\mathcal{O}_c\left(N\Big(M\ell_1+\ell_2\Big)\right) = \mathcal{O}_c\left(N\left(\varepsilon^{-\frac{1}{2r}}\log\left(\varepsilon^{-(1-\frac{1}{r})}\right) + \log\left(\varepsilon^{-1}\right)\right)\right)$$

$$= \mathcal{O}_c\left(N\varepsilon^{-\frac{1}{2r}}\log\left(\varepsilon^{-1}\right)\right),$$

with equilibrium at $r = m+1$ in (41). Remembering the choice (45) of ε w.r.t. N gives the announced result.

Remark 5. *One can observe that $\widetilde{Y}_{j,k}(\cdot)$ may require only one $\widetilde{g}(\cdot)$ since the term $\Psi_H(-k)$ does not depend on t and so could be approximated by a single NN parameter. It slightly simplifies the construction of Figure 6 although it does not change the final asymptotic bounds.*

4 Numerical results

Computational aspects The numerical experiments have been conducted on the Cholesky computing cluster from Ecole Polytechnique http://meso-ipp.gitlab.labos.polytechnique.fr/user_doc/. All the code was implemented in Python 3.8.2 and using the library PyTorch 1.7.1 for the neural network training.

As mentioned in the introduction, the objective of this paper is neither to focus on the learning task nor to solve the adversarial optimization problem in a GAN setting. As a complement to the above theoretical part, we present here one approach to build a generative NN able to simulate fBm paths. The computing implementation of (29) on a time grid time $\{t_j = j/T,\, j = 0, \dots, T\}$ is realized in three steps (see more details below): (1) approximate Ψ_H on a specific support, (2) find the set of indices $\mathcal{I}_N$ and (3) draw some noise from the latent space.

1. In order to approximate the function

$$x \mapsto \Psi_H(x) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{ix\xi}\frac{\widehat{\psi^{\mathrm{M}}}(\xi)}{(i\xi)^{H+1/2}}\,\mathrm{d}\xi, \tag{47}$$

take M points $\{x_i\}_{i=1}^M$ from a grid with uniform step in a large enough finite interval and compute $\{\Psi_H(x_i)\}_{i=1}^M$ using (13) and (16) with a fixed tail-index $\gamma = 1/\beta$ in (16). As illustrated in Figure 9d, one can benefit from a non-uniform grid of points $\{x_i\}_{i=1}^M$ for better approximation. Note that given the high regularity of Ψ_H, a quadrature method with degree D will give an excellent estimation of the integral in (47). Then, it is straightforward to build a feedforward NN $\widetilde{\Psi}_{H,\theta}$ that can minimize the L^2 distance

$$\mathcal{L}(\theta) = \frac{1}{M}\sum_{i=1}^M \left|\Psi_H(x_i) - \widetilde{\Psi}_{H,\theta}(x_i)\right|^2.$$

2. Compute the index set $\mathcal{I}_N$ based on the one proposed in [2, Section 5]. For every integer $J \geq 0$, let $\mathcal{F}_J$ and $\mathcal{P}_J$ be the sets of indices defined as

$$\mathcal{F}_J = \left\{(j,k) \in \mathbb{Z}^2 : 0 \leq j \leq J \text{ and } |k| \leq (J-j+1)^{-2}2^{J+4}\right\},$$
$$\mathcal{P}_J = \left\{(j,k) \in \mathbb{Z}^2 : -J \leq j \leq -1 \text{ and } |k| \leq 2^{[J/2]}\right\}.$$

Next, for $(j,k) \in \mathcal{I}_J = \mathcal{F}_J \cup \mathcal{P}_J$, consider the coefficients

$$c_{j,k} = 2^{-2jH}\left(\Psi_H(2^j - k) - \Psi_H(-k)\right)^2,$$

and with a slight variation from [2], define the truncated set $\mathcal{I}_N$ containing the N largest coefficients $c_{j,k}$ as

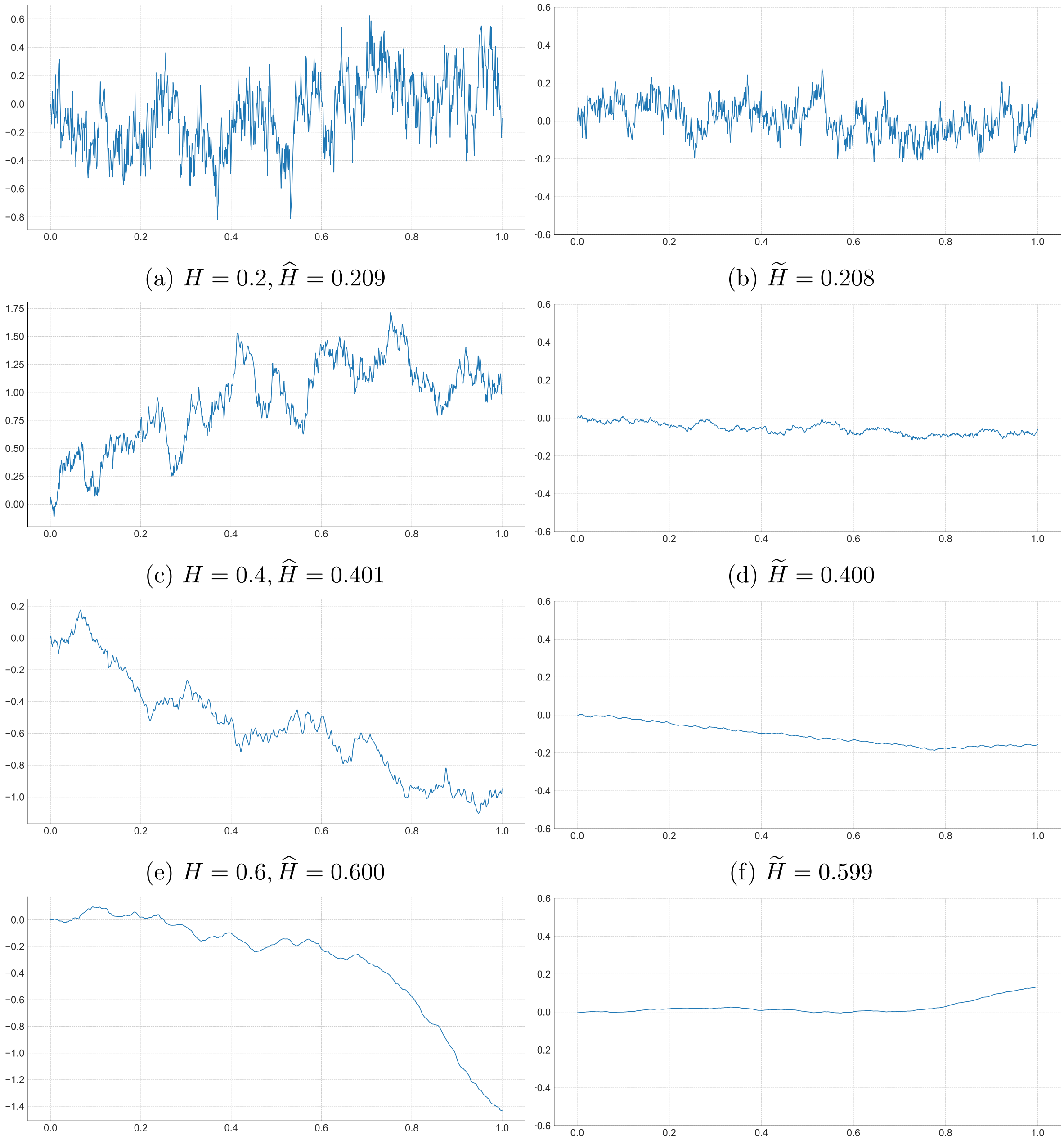
$$\mathcal{I}_N := \bigcup_{i=1}^N \mathcal{B}_i,$$

with $\mathcal{B}_{i+1} = \left\{ \underset{(j,k)\in\mathcal{I}_J}{\arg\max}\, c_{j,k} \backslash \mathcal{B}_i \right\}$, $\mathcal{B}_1 = \underset{(j,k)\in\mathcal{I}_J}{\arg\max}\, c_{j,k}$ and $N \leq \mathrm{card}(\mathcal{I}_J)$. Note that (29) requires the evaluation of Ψ_H at some points $\{x_i\}_{i=1}^{M}$ with an increasing support with respect to J. Therefore, one must choose a degree D large enough such that the integral values are well estimated at the boundaries of the support.

3. Simulate some independent standard Gaussian random variables $G_{j,k}$. For the sake of simplicity and since the product approximation error (Proposition 4) is geometrically small with respect to the depth, one can use the real product instead.

In the simulation study, we used $M = 10000$ in $[-1000, 1000]$, including 8000 points in $[-50, 50]$, $J = 10, N = 40000, \gamma = 10000, T = 1000$ and $D = 20000$ with a Gauss-Chebyshev quadrature [1, p. 889]. The neural network $\widetilde{\Psi}_{H,\theta}$ is composed by 10 hidden layers of 200 neurons in order to be consistent with the theoretical result ($\mathcal{O}_c(\log N)$ hidden layers and $\mathcal{O}_c(N\log N)$ parameters). Obviously, one could use a much lighter parametrization. The model was trained with the Adam optimizer [26] with default parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$, a learning rate of 1e-3 and a batch-size of 1024 during 1000 iterations. The best θ is selected by minimizing the loss evaluated every 10 iterations on the training data set.

The estimated Hurst parameter $\widehat{H}$ on B_N^H (resp. $\widetilde{H}$ on $\widetilde{B}_N^H$) is computed using a basic absolute-moment estimation by identifying the slope of $\log t \mapsto \log B_N^H(t)$ (resp. $\log \widetilde{B}_N^H(t)$) [18]. In order to assess the quality of the simulated fBm for different H, Figure 7 proposes graphical fBm wavelet representations of one trajectory and its error with a NN fBm simulation. The persistence (resp. non-persistence) phenomena is graphically confirmed on B_N^H and the small error shows that it is also the case for $\widetilde{B}_N^H$ as $H < 1/2$ (resp. $H > 1/2$). Additionally, the excellent approximation of the estimated Hurst parameters $\widetilde{H}$ confirms that the regularity of the NN simulated paths is well preserved. On the other hand in Figure 8, we compare the real covariance function (2) with the one on $\widetilde{B}_N^H$. The small error between the two surfaces illustrates that the covariance function of $\widetilde{B}_N^H$ is also well preserved. To conclude, although this method of simulation may not be optimal in the sense of speed or accuracy, the numerical results highlight a good performance of our proposed generative NN model to simulate realistic fBm paths.



(a) $H = 0.2, \widehat{H} = 0.209$

(b) $\widetilde{H} = 0.208$

(c) $H = 0.4, \widehat{H} = 0.401$

(d) $\widetilde{H} = 0.400$

(e) $H = 0.6, \widehat{H} = 0.600$

(f) $\widetilde{H} = 0.599$

(g) $H = 0.8, \widehat{H} = 0.792$

(h) $\widetilde{H} = 0.790$

Figure 7: Simulation of fBm for $H = \{0.2, 0.4, 0.6, 0.8\}$ presented as rows. Left: wavelet representation $t \mapsto V_H^{-1/2} B_N^H(t)$ with the estimated Hurst index $\widehat{H}$. Right: error $t \mapsto V_H^{-1/2}(B_N^H(t) - \widetilde{B}_N^H(t))$ with the estimated Hurst index $\widetilde{H}$ on the NN fBm.

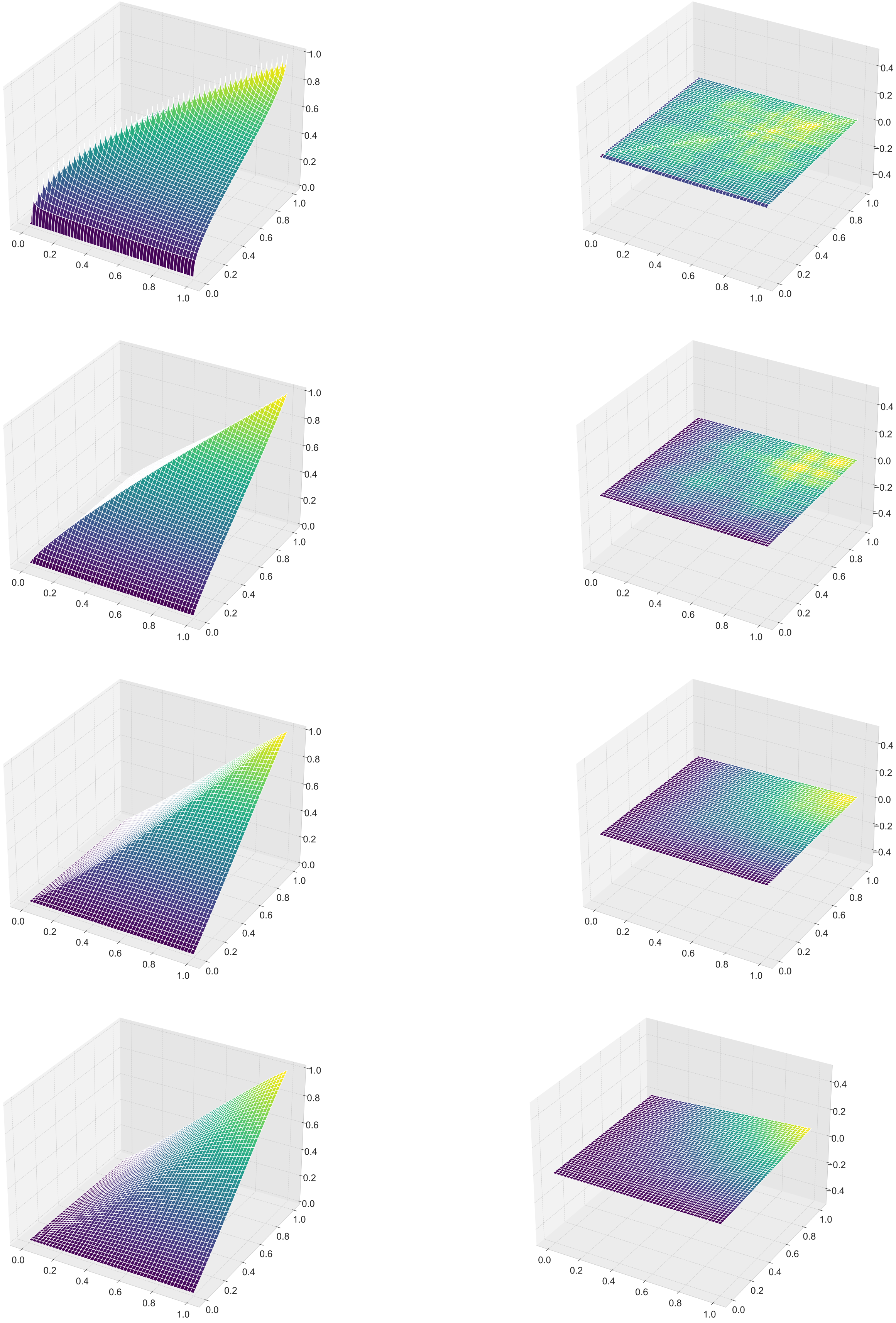


Figure 8: fBm covariance surface for $H \in \{0.2, 0.4, 0.6, 0.8\}$ presented as rows. Left: real normalized function $(t,s) \mapsto V_H^{-1}\mathbb{C}\mathrm{ov}(B^H(t), B^H(s))$. Right: error $(t,s) \mapsto V_H^{-1}(\mathbb{C}\mathrm{ov}(B^H(t), B^H(s)) - \mathbb{C}\mathrm{ov}(\widetilde{B}_N^H(t), \widetilde{B}_N^H(s)))$ for $(t,s) \in [0,1]^2$.

A Complements

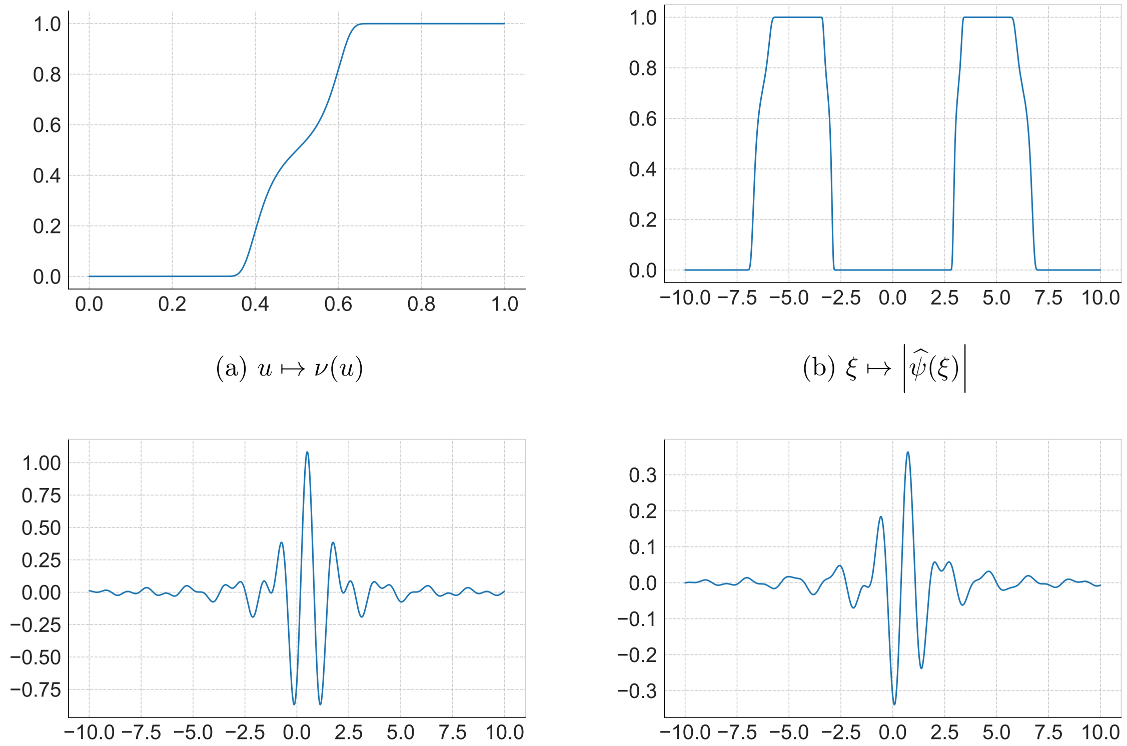
A.1 Proof of Lemma 1

Let us bound the truncated approximation of (7) for all $n \in \mathbb{N}$,

$$\begin{aligned}\sup_{t\in[0,1]}\left|B(t)-\left(G_1 t+\sum_{j=0}^{n}\sum_{k=0}^{2^j-1}2^{-(j+1)}\psi^{\mathrm{FS}}_{j,k}(t)G_{j,k}\right)\right|\\
&=\sup_{t\in[0,1]}\left|\sum_{j=n+1}^{\infty}\sum_{k=0}^{2^j-1}2^{-(j+1)}\psi^{\mathrm{FS}}_{j,k}(t)G_{j,k}\right|\\
&\leq\sum_{j=n+1}^{\infty}2^{-(j/2+1)}\sup_{0\leq k\leq 2^j-1}|G_{j,k}|\\
&\leq C\sum_{j=n+1}^{\infty}2^{-(j/2+1)}\big(\log(j+2^j+1)\big)^{1/2}\quad\text{a.s.}\\
&\leq C\sum_{j=n+1}^{\infty}2^{-\frac{1}{2}(j+1)}\left(1+j\right)^{1/2}\\
&\leq C\;2^{-\frac{1}{2}(n+1)}\left(n+1\right)^{1/2}\\
&\leq C\;N^{-1/2}\big(1+\log(N)\big)^{1/2},\end{aligned}$$

where C is a non-negative random variable which may change from line to line. In the third line, use the fact that the wavelets have disjoint support in k and so for fixed j, any t belongs to the support of at most one $\psi^{\mathrm{FS}}_{j,k}$, with $\left\|\psi^{\mathrm{FS}}_{j,k}\right\|_\infty \leq 2^{j/2}$. In the fourth, invoke [2, Lemma 2]; in the fifth the inequality holds for j large enough; in the sixth use a classical integral test, and lastly replace with N. □

A.2 Wavelet representation



(a) $u \mapsto \nu(u)$

(b) $\xi \mapsto \left|\widehat{\psi}(\xi)\right|$

(c) $x \mapsto \psi(x)$

(d) $x \mapsto \varPsi_H(x)$ with $H = 0.2$

Figure 9: Lemarié-Meyer wavelet constructed with (16), (13), (11) for $\gamma = 10$.

B Harmonizable representation of fBm

B.1 Real valued Gaussian measure

Let $f \in \mathbf{L^2}(\mathbb{R}, \mathrm{d}x) = \left\{f : \mathbb{R} \mapsto \mathbb{R} : \int_{-\infty}^{\infty} |f(x)|^2 \,\mathrm{d}x < \infty\right\}$ and denote $\langle f,\, g\rangle = \int_{-\infty}^{\infty} f(x)\overline{g(x)}\,\mathrm{d}x$ the inner product in $\mathbf{L^2}(\mathbb{R}, \mathrm{d}x)$ (still valid for complex-valued functions). Define respectively the Fourier transform $\mathcal{F} : f \in \mathbf{L^1}(\mathbb{R}, \mathrm{d}x) \mapsto \widehat{f} = \mathcal{F}(f) : \xi \in \mathbb{R} \mapsto \int_{-\infty}^{\infty} f(x)\mathrm{e}^{-\mathbf{i}x\xi}\,\mathrm{d}x$ and the inverse Fourier transform $\mathcal{F}^{-1} : \varphi \in \mathbf{L^1}(\mathbb{R}, \mathrm{d}\xi) \mapsto \mathcal{F}^{-1}(\varphi) : x \in \mathbb{R} \mapsto \frac{1}{2\pi}\int_{-\infty}^{\infty} \varphi(\xi)\mathrm{e}^{\mathbf{i}x\xi}\,\mathrm{d}\xi$ by the definitions (5). We recall the Parseval-Plancherel formula for any $f, h \in \mathbf{L^1}(\mathbb{R}, \mathrm{d}x) \cap \mathbf{L^2}(\mathbb{R}, \mathrm{d}x)$:

$$\int_{-\infty}^{\infty} f(x)\overline{h(x)}\,\mathrm{d}x = \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{f}(\xi)\overline{\widehat{h}(\xi)}\,\mathrm{d}\xi. \tag{48}$$

This isometry (up to the constant 2π) allows to extend the Fourier transform $\mathcal{F}$ and its inverse $\mathcal{F}^{-1}$ to square integrable functions, see [33, Section 2.2]. Observe that $\mathcal{F}^{-1}(\varphi)$ is a real-valued function when

$$\varphi \in \mathbf{L^2_r}(\mathbb{R}, \mathrm{d}\xi) = \left\{\varphi : \mathbb{R} \mapsto \mathbb{C} \text{ s.t. } \int_{-\infty}^{\infty} |\varphi(x)|^2 \,\mathrm{d}x < \infty \text{ and } \varphi(-\xi) = \overline{\varphi(\xi)}\right\},$$

therefore $\mathcal{F}$ is an isometry between $\mathbf{L^2}(\mathbb{R}, \mathrm{d}x)$ and $\mathbf{L^2_r}(\mathbb{R}, \mathrm{d}\xi)$.

Then, see [7, Section 2.1.6], one can define the centered-real valued Gaussian measure by

$$X(\varphi) := \int_{-\infty}^{\infty} \mathcal{F}^{-1}(\varphi)(x)\,\mathrm{d}W(x) \overset{notation}{=} \int_{-\infty}^{\infty} \varphi(\xi)\,\mathrm{d}\widehat{W}(\xi)$$

for $\varphi \in \mathbf{L}^{\mathbf{2}}_{\mathbf{r}}(\mathbb{R}, \mathrm{d}\xi)$, such that

$$\mathbb{C}\mathrm{ov}\left(X(\varphi_1), X(\varphi_2)\right) = \int_{-\infty}^{\infty} \mathcal{F}^{-1}(\varphi_1)(x)\mathcal{F}^{-1}(\varphi_2)(x)\,\mathrm{d}x. \tag{49}$$

If $(e_n)_n$ is an orthonormal basis of $\mathbf{L}^{\mathbf{2}}(\mathbb{R}, \mathrm{d}x)$, then

$$X(\varphi) = \sum_n \left\langle e_n,\, \mathcal{F}^{-1}(\varphi)\right\rangle G_n \tag{50}$$

where $G_n = \int_{-\infty}^{\infty} e_n(x)\,\mathrm{d}W(x)$ are i.i.d. standard Gaussian random variables.

B.2 Series representation of fBm

For $t \in [0,1]$, take $\varphi_t(\xi) := \frac{\mathrm{e}^{\mathrm{i}t\xi}-1}{(\mathbf{i}\xi)^{H+1/2}} \in \mathbf{L}^{\mathbf{2}}_{\mathbf{r}}(\mathbb{R}, \mathrm{d}\xi)$ and set

$$B^H(t) := \int_{-\infty}^{\infty} \varphi_t(\xi)\,\mathrm{d}\widehat{W}(\xi). \tag{51}$$

We now verify that this harmonizable representation leads to the fBm-representation (10), (11), (12) consistently with the covariance formula (2).

For this, as an orthonormal basis of $\mathbf{L}^{\mathbf{2}}(\mathbb{R}, \mathrm{d}x)$, consider the basis functions generated by mother wavelet function ψ: $\left\{\psi_{j,k}(x) = 2^{j/2}\psi(-2^j x - k)\right\}_{(j,k)\in\mathbb{Z}^2}$. Note that usually, to a given mother wavelet ψ, one invokes the function basis $\left\{\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)\right\}_{(j,k)\in\mathbb{Z}^2}$: our choice of changing the sign in front of x is made for getting exactly the representation (10), (11) at the end, of course the sign change does not affect the orthonormality property of the basis as it can be easily checked. In view of (50) and (48), we have to compute the coefficients

$$c_{j,k}(t) := \left\langle \psi_{jk},\, \mathcal{F}^{-1}(\varphi_t)\right\rangle = \frac{1}{2\pi}\int_{-\infty}^{\infty} \widehat{\psi_{jk}}(\xi)\overline{\varphi_t(\xi)}\,\mathrm{d}\xi.$$

A direct computation gives

$$\begin{aligned}
\widehat{\psi_{j,k}}(\xi) &= 2^{j/2}\int_{-\infty}^{\infty} \mathrm{e}^{-\mathbf{i}x\xi}\psi(-2^j x - k)\,\mathrm{d}x = 2^{-j/2}\mathrm{e}^{\mathbf{i}\xi k 2^{-j}}\widehat{\psi}(-2^{-j}\xi),\\
c_{j,k}(t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{\mathrm{e}^{-\mathbf{i}\xi t}-1}{(-\mathbf{i}\xi)^{H+1/2}} 2^{-j/2}\mathrm{e}^{\mathbf{i}\xi k 2^{-j}}\widehat{\psi}(-2^{-j}\xi)\,\mathrm{d}\xi\\
&= \frac{2^{-jH}}{2\pi}\int_{-\infty}^{\infty} \frac{\mathrm{e}^{\mathbf{i}\omega 2^j t}-1}{(\mathbf{i}\omega)^{H+1/2}}\mathrm{e}^{-\mathbf{i}\omega k}\widehat{\psi}(\omega)\,\mathrm{d}\omega \quad (\text{setting } \omega = -2^{-j}\xi)\\
&= 2^{-jH}\left(\varPsi_H\left(2^j t - k\right) - \varPsi_H\left(-k\right)\right)
\end{aligned}$$

recalling the definition (11) of $\varPsi_H$. All in all, from (50) we have obtained

$$B^H(t) = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} 2^{-jH}\left(\varPsi_H\left(2^j t - k\right) - \varPsi_H\left(-k\right)\right) G_{j,k}.$$

We get the announced representation (10).

B.3 Covariance of B^H

We now establish (2), with V_H given by (12). We first state a useful formula.

Lemma 7. *Let $\beta \in (0,2)$. Then for all $t \in [0,1]$,*

$$\int_0^\infty \frac{1-\cos(ut)}{u^{\beta+1}}\,\mathrm{d}u = 2^{-\beta}\sqrt{\pi}\frac{t^\beta}{\beta}\frac{\Gamma\left(1-\frac{\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right)}.$$

Proof. Starting from

$$\int_0^\infty \frac{1-\cos(ut)}{u^{\beta+1}}\,\mathrm{d}u = t^\beta \int_0^\infty \frac{1-\cos(y)}{y^{\beta+1}}\,\mathrm{d}y = \frac{t^\beta}{\beta}\int_0^\infty \frac{\sin(y)}{y^\beta}\,\mathrm{d}y,$$

by change of variable $y = ut$, integrating by part and removing the first term since $\beta \in (0,2)$ and $1-\cos(y) \sim y^2/2$ as $y \to 0$. Recognizing a known integral function [39, Equation (13) p. 387] finishes the proof. □

Now, let $0 \le s \le t \le 1$. Starting from (51), and using (49)-(48), we get

$$\begin{aligned}\mathbb{C}\mathrm{ov}\left(B^H(t), B^H(s)\right) &= \frac{1}{2\pi}\int_{-\infty}^\infty \frac{\left(\mathrm{e}^{\mathbf{i}\xi t}-1\right)\left(\mathrm{e}^{-\mathbf{i}\xi s}-1\right)}{|\xi|^{2H+1}}\,\mathrm{d}\xi \\ &= \frac{1}{\pi}\int_0^\infty \frac{(1-\cos(\xi t)) + (1-\cos(\xi s)) - (1-\cos(\xi(t-s)))}{|\xi|^{2H+1}}\,\mathrm{d}\xi.\end{aligned}$$

Using Lemma 7 with $\beta = 2H \in (0,2)$ entails

$$\mathbb{C}\mathrm{ov}\left(B^H(t), B^H(s)\right) = \frac{2^{-2H}}{2H\sqrt{\pi}}\frac{\Gamma(1-H)}{\Gamma(\frac{1}{2}+H)}\left(|t|^{2H} + |s|^{2H} - |t-s|^{2H}\right).$$

The Euler reflection and the Legendre duplication formulas write for all $a \notin \mathbb{Z}$:

$$\Gamma(1-a)\Gamma(a) = \frac{\pi}{\sin(\pi a)}, \quad \Gamma\left(\frac{1}{2}+a\right)\Gamma(a) = 2^{1-2a}\sqrt{\pi}\Gamma(2a).$$

Therefore, we retrieve (2) with

$$\mathbb{V}\mathrm{ar}\left(B^H(1)\right) = V_H = \frac{2^{-2H}}{H\sqrt{\pi}}\frac{\Gamma(1-H)}{\Gamma(\frac{1}{2}+H)} = \frac{1}{2H\sin(\pi H)\Gamma(2H)}$$

as announced. □

Acknowledgements

The authors would like to thank the referee and the Associate Editor for their valuable suggestions, which have significantly improved the paper. This action benefited from the support of the Chair Stress Test, Risk Management and Financial Steering, led by the French Ecole polytechnique and its foundation and sponsored by BNP Paribas. This work has been partially supported by MIAI @ Grenoble Alpes, (ANR-19-P3IA-0003).

References

[1] M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions with formulas, graphs, and mathematical tables.* National Bureau of Standards Applied Mathematics Series, No. 55. U. S. Government Printing Office, Washington, D. C., 1964.

[2] A. Ayache and M. S. Taqqu. Rate optimality of wavelet series approximations of fractional Brownian motion. *J. Fourier Anal. Appl.*, 9(5):451–471, 2003.

[3] M. Bennedsen. A rough multi-factor model of electricity spot prices. *Energy Econ.*, 63:301–313, 2017.

[4] C. Burdet and P. Rougier. Analysis of center-of-pressure data during unipedal and bipedal standing using fractional Brownian motion modeling. *J. Appl. Biomech.*, 23(1):63–69, 2007.

[5] J.-F. Coeurjolly. Simulation and identification of the fractional Brownian motion: A bibliographical and comparative study. *J.. Stat. Softw.*, 5:1–53, 2000.

[6] J.-F. Coeurjolly and E. Porcu. Fast and exact simulation of complex-valued stationary Gaussian processes through embedding circulant matrix. *J. Comput. Graph. Statist.*, 27(2):278–290, 2018.

[7] S. Cohen and J. Istas. *Fractional fields and applications*, volume 73 of *Mathématiques & Applications.* Springer, Heidelberg, 2013.

[8] F. Comte and E. Renault. Long memory in continuous-time stochastic volatility models. *Math. Finance*, 8(4):291–323, 1998.

[9] J. Creutzig, T. Müller-Gronbach, and K. Ritter. Free-knot spline approximation of stochastic processes. *J. Complexity*, 23(4-6):867–889, 2007.

[10] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Systems*, 2(4):303–314, 1989.

[11] I. Daubechies. *Ten lectures on wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992.

[12] L. de Haan and A. Ferreira. *Extreme value theory.* Springer Series in Operations Research and Financial Engineering. Springer, New York, 2006.

[13] T. Dieker. Simulation of fractional Brownian motion. Master's thesis, Department of Mathematical Sciences, University of Twente, 2004.

[14] D. E. Dominici. The inverse of the cumulative standard normal probability function. *Integral Transforms Spec. Funct.*, 14(4):281–292, 2003.

[15] C. Esteban, S. L. Hyland, and G. Rätsch. Real-valued (medical) time series generation with recurrent conditional gans. 2017.

[16] G. Faber. Über stetige Funktionen. *Math. Ann.*, 66(1):81–94, 1908.

[17] F. Flandoli and M. Gubinelli. Random currents and probabilistic models of vortex filaments. In *Seminar on Stochastic Analysis, Random Fields and Applications IV*, volume 58 of *Progr. Probab.*, pages 129–139. Birkhäuser, Basel, 2004.

[18] M. Garcin. Estimation of time-dependent Hurst exponents with variational smoothing and application to forecasting foreign exchange rates. *Phys. A*, 483:462–479, 2017.

[19] J. Gatheral, T. Jaisson, and M. Rosenbaum. Volatility is rough. *Quant. Finance*, 18(6):933–949, 2018.

[20] E. Gobet, J. G. López-Salas, and C. Vázquez. Quasi-regression Monte-Carlo scheme for semilinear PDEs and BSDEs with large scale parallelization on GPUs. *Arch. Comput. Methods Eng.*, 27(3):889–921, 2020.

[21] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 2016.

[22] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Adv. Neural Inf. Process. Syst.*, volume 27, 2014.

[23] A. Haar. Zur Theorie der orthogonalen Funktionensysteme. *Math. Ann.*, 69(3):331–371, 1910.

[24] M. Haas and S. Richter. Statistical analysis of Wasserstein GANs with applications to time series forecasting. 2020.

[25] W. P. Johnson. The curious history of Faà di Bruno's formula. *Amer. Math. Monthly*, 109(3):217–234, 2002.

[26] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *3rd Int. Conf. on Learn Represent., ICLR*, 2015.

[27] Y. Kozachenko, A. Pashko, and O. Vasylyk. Simulation of generalized fractional Brownian motion in $C([0,T])$. *Monte Carlo Methods Appl.*, 24(3):179–192, 2018.

[28] T. Kühn and W. Linde. Optimal series representation of fractional Brownian sheets. *Bernoulli*, 8(5):669–696, 2002.

[29] M. Ledoux and M. Talagrand. *Probability in Banach spaces*, volume 23 of *Ergebnisse der Mathematik und ihrer Grenzgebiete (3) Results in Mathematics and Related Areas (3)*. Springer-Verlag, Berlin, 1991.

[30] P. G. Lemarié and Y. Meyer. Ondelettes et bases Hilbertiennes. *Rev. Mat. Iberoamericana*, 2(1-2):1–18, 1986.

[31] J. Li and Z. Qian. Fine properties of fractional Brownian motions on Wiener space. *J. Math. Anal. Appl.*, 473(1):141–173, 2019.

[32] V. E. Maiorov. On best approximation by ridge functions. *J. Approx. Theory*, 99(1):68–94, 1999.

[33] S. Mallat. *A wavelet tour of signal processing*. Elsevier/Academic Press, Amsterdam, third edition, 2009.

[34] B. B. Mandelbrot and J. W. Van Ness. Fractional Brownian motions, fractional noises and applications. *SIAM Rev.*, 10:422–437, 1968.

[35] Y. Meyer, F. Sellan, and M. S. Taqqu. Wavelets, generalized white noise and fractional integration: the synthesis of fractional Brownian motion. *J. Fourier Anal. Appl.*, 5(5):465–494, 1999.

[36] I. Nourdin. *Selected aspects of fractional Brownian motion*, volume 4 of *Bocconi & Springer Series*. Springer ; Bocconi University Press, Milan, 2012.

[37] A. Pinkus. Approximation theory of the MLP model in neural networks. In *Acta numerica, 1999*, volume 8 of *Acta Numer.*, pages 143–195. Cambridge Univ. Press, Cambridge, 1999.

[38] A. Pontieri, F. Lepreti, L. Sorriso-Valvo, A. Vecchio, and V. Carbone. A simple model for the solar cycle. *Sol. Phys.*, 213(1):195–201, 2003.

[39] A. P. Prudnikov, Y. A. Brychkov, and O. I. Marichev. *Integrals and series. Vol. 1.* Gordon & Breach Science Publishers, New York, 1986.

[40] J. Schauder. Eine Eigenschaft des Haarschen Orthogonalsystems. *Math. Z.*, 28(1):317–320, 1928.

[41] J. M. Steele. *Stochastic calculus and financial applications*, volume 45 of *Applications of Mathematics (New York)*. Springer-Verlag, New York, 2001.

[42] M. Telgarsky. Representation benefits of deep feedforward networks. 2015.

[43] M. Wiese, R. Knobloch, R. Korn, and P. Kretschmer. Quant GANs: deep generation of financial time series. *Quant. Finance*, 20(9):1419–1440, 2020.

[44] D. Yarotsky. Error bounds for approximations with deep ReLU networks. *Neural Networks*, 94:103–114, 2017.