Efficient Option Pricing Under Rough Volatility Models

by

Pietro Pezzoli Frigerio

Faculty: Applied Mathematics, TU Delft
First Supervisor: Kristin Kirchner
Second Supervisor: Antonis Papapantoleon

TU Delft

Abstract

Rough volatility models have become a prominent tool in quantitative finance due to their ability to capture the rough nature of financial time series. However, these models typically have a non-Markovian structure, and this poses significant computational challenges. Existing methods for approximating these models often involve either complex quadrature techniques or require optimization algorithms operating in high-dimensional spaces, making them difficult and computationally intensive to implement in practice. This work introduces a novel and highly efficient quadrature rule for a Markovian approximation of rough volatility models, along with its theoretical error bounds. We verify the error bounds through a detailed error analysis and conduct several experiments on option pricing, demonstrating that the proposed method outperforms other state-of-the-art methodologies in terms of efficiency and accuracy across a range of roughness parameters. A further advantage of the approach is its ease to implement it in practice.

Contents

Nomenclature

Frequently Used Notation

Symbol	Definition
Γ	Gamma function
h	Step-size
H	Hurst parameter
$\mathcal{L}^p([0,T])$	p-integrable functions over [0,T]
M	Truncation level on the half negative real line
N	Truncation level on the half positive real line
K	Total number of nodes in a given quadrature rule
$\mathbb{N}$	Set of natural numbers, zero included
$\mathbb{P}$	Physical Measure
$\mathbb{Q}$	Martingale Measure
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^+$	Half positive real line, zero excluded
$\mathbb{R}^-$	Half negative real line, zero excluded
T	Final time
W	Brownian Motion
$\lvert\cdot\rvert$	Absolute value
$\lVert\cdot\rVert$	Norm
$\lceil\cdot\rceil$	Ceil function

1

Introduction

In Financial Mathematics, volatility refers to the degree of variation of the price of a certain financial asset or instrument. It is a key measure for risk, that gives insight on the stability and uncertainty of the price. Hence the modelling of volatility is of utmost importance in financial mathematics and risk management.

The class of rough volatility models, has been introduced by Gatheral et al in [28] where for the Rough Fractional stochastic volatility (RFSV) model was born. Modifications and of this model lead to the Rough Bergomi model in [9]. Furthermore, generalization of the classical Heston model brought to the Rough Heston model in [22]. All those models have something in common, that is in their formulations appear a stochastic integral of the form:

$$\int_0^T g(t)dW_t, \quad g(\tau) = \frac{t^{H-1/2}}{\Gamma(H+1/2)}$$

where g is a power-law function, with a singularity in $t = 0$, being $H \in (-1/2, 1/2)$. The result of this stochastic integral, due to the nature of the integrand, is a process that it's neither a martingale nor a Markov process. This of course implies that the simulation and pricing under Rough volatility can be extremely expensive. When it comes to pricing, the literature distinguishes two cases: one is the class of affine forward variance models, which include also the rough Heston model.In this class, one can make use of the characteristic function of the model, which is available modulo the resolution of a fractional Riccati equation. Once the characteristic function of the model is available, one can use Fourier pricing methods like [25],[16]. In the other cases, where the characteristic function is not available and one needs to price in a different manner: Monte Carlo simulations [41],[33], Quasi Monte carlo [5], Weak Type schemes [29],[29], PDE approaches [4]. Also, a very popular method is the Hybrid Scheme [10] and its turbocharged version for the Rough Bergomi model [37].

In this work, we will focus on approximating these models using a Markovian approximation introduced first by Abi Jaber and El Euch in [1]. Further development of this approach have been [7],[8]. This method employs a quadrature rule to approximate the rough kernel. This type of approximations have the advantage of being extremely flexible, being useful both for the approximation of models like the Rough Heston and the Rough Bergomi model.

In this paper, we are going first to introduce the concept of volatility in mathematical finance, showing what type of characteristics we want such model to satisfy. After that we give a review for two different models: the Rough Heston and the Rough Bergomi. Then we present how one can construct the Markovian Approximation of such models. Finally, in chapter 6 and 7 we begin our contribution. Specifically, in our work we use the SINC quadrature. By transforming the rough kernel into an integral over the entire real line, we can then apply the simplest quadrature rule for integrals: the trapezoidal rule. The main advantage of this method is that it is extremely easy to implement without relying on complex polynomial bases, as required in Gaussian quadrature and at the same time is able to outperform the most recent advancements in the literature. Additionally, this approach maintains the exponential decay of the error as the number of quadrature points increases and in fact gives a better approximation than the state of the art [8] when the Hurst parameter $H \in (-0.5, 0.2)$.

The structure of this work is as follows:

In Chapter 2, our focus will be on introducing fundamental concepts of volatility modeling and its critical importance in financial applications, especially in pricing contexts. The chapter aims to emphasize the essential properties that a volatility model must satisfy to effectively capture market dynamics and facilitate accurate pricing.

In Chapter 3, we will introduce two different rough volatility models: the Rough Bergomi model and the Rough Heston model. Additionally, we will introduce Brownian Semi-Stationary processes and Truncated Brownian Semi-Stationary processes. This chapter will include a theoretical discussion on how these models have been introduced in the literature, along with their advantages and disadvantages.

In Chapter 4, we will present various methods for simulating rough volatility models. Specifically, we will introduce two different types of exact schemes and the Hybrid scheme, which remains the state-of-the-art method for simulating Truncated Brownian Semistationary Processes. This chapter will also examine the limitations of each of these methods.

In Chapter 5, we will introduce the construction of a Markovian approximation for rough volatility models. We will explain the underlying concept and specifically show how to bound the error, both weak and strong, using the $\mathcal{L}^p$ error between the fractional kernel and its approximation through a quadrature rule. Additionally, we will review various quadrature rules from the literature, with special emphasis on those that exhibit exponential convergence with respect to the total number of nodes used

In Chapter 6, that is the start of our contribution, we will delve into the theoretical foundation of our work, focusing on providing error bounds for the trapezoidal rule based on the total number of nodes. Specifically, we will derive bounds for the $\mathcal{L}^p([0,T])$ errors, where $p \in 1, 2$.

Moving to Chapter 7, we will conduct numerical experiments to validate the findings from Chapter 6. First, we are going to present two different algorithms that we have developed for the SINC quadrature. One will be an algorithm that optimally chooses the step size for final times in the range $T > 0.1$, while the second will choose the best step size for $T < 0.1$. Inspired by the approach outlined in [8], our emphasis will be on comparing theoretical and empirical convergence rates of the errors. Furthermore, we will compare different quadrature rules against our method, considering factors such as computational time, node distribution, and convergence rates. To complete our study, we will present various numerical experiments within financial applications. These experiments will involve computing option prices and constructing implied volatility smiles and surfaces under the Rough Heston model, considering both European-style and Asian geometric options.

2

Volatility

In this chapter, we present the core concept Volatility in financial mathematics and the so-called *stylised facts*. Our goal is to present the phenomena that characterise Volatility and in the next chapter, to see how rough volatility models propose a novel approach to capture these phenomena with high accuracy. Finance is a matter of balance between returns and risk. One of the benchmark for the latter is *volatility*. Volatility in itself, however, can not be observed in the same way returns can be observed. So one has to find a way to derive a proxy for this quantity. Having in mind that the price of an asset is always positive, the standard practice is to see its returns as an exponential of some random variable, say for example:

$$S_t/S_{t-1} = e^{\xi}$$

with $\xi \sim N(0, \sigma^2)$ So one can define an estimator of the volatility as

$$\sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(log\frac{S_t}{S_{t-1}}\right)^2}$$

Of course, if one were to compute this estimate for the volatility of any asset present on the market on a given time period and then recompute it on a different period one would most likely get different results. In fact, it seems logical and also realistic to assume that volatility is a process that is time varying. Despite these initial observations, it was in 1973 when Black and Scholes [38] revolutionised finance with a closed form option pricing formula that is independent of any utility function or subjective preference.

One of Black and Scholes formula assumptions is that volatility was assumed to be constant, implying then that the distribution of returns is log-normal. However, in the 1987 market crash known as Black Monday, where the SP500 futures price fell 29% made it clear that this assumption is not realistic and there was a need for a change in volatility modelling. In fact, for a log-normal distribution, a 29% change in price in one day would be -27 standard deviation event, that happens with probability 10^{-160}. Such an event, that was maybe caused by something exogenous made it clear that there was a need for a better modelling of asset's returns and as a consequence, volatility.

Another phenomenon that appeared in the aftermath of the Black Monday market crash was the so called Implied Volatility smile: Let us first introduce the concept of implied volatility.

2.1. Implied Volatility

In this section we are going to introduce the concept of implied volatility. First, recall [39] that a call, respectively put option, on an underlying S, with time to expiration T and strike price K, is a contract that allows its owner to buy or sell at time T an asset S at price K. Now, the option's payoff can be

seen as a function of S_T, K, T and be written as:

$$(S_T - K)^+ = max(S_T - K,\ 0), \text{ in case of a call option}$$
$$(K - S_T)^+ = max(K - S_T,\ 0), \text{ in case of a put option}$$

Now, while K, T are known and fixed for every contract, the final value of the price S_t is of course a random value. Usually then S_t is modelled as a certain $\mathcal{F}_t-$adapted stochastic process living in a filtered probability space $\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mathbb{P}\right)$. Let us now introduce a nomenclature for options that allows to understand the contract's characteristics with respect to the price of the underlying asset. In options trading, the terms 'out of the money', 'in the money', and 'at the money' are used to describe the relationship between the option's strike price and the current price of the underlying asset. Here are the definitions:

- Out of The Money (OTM): A call (put) option is out of the money if the current price of the underlying asset is below (above) the option's strike price. This means the option holder would not benefit from exercising the option at the current asset price
- In The Money (ITM): A call (put) option is in the money if the current price of the underlying asset is above (below) the option's strike price. Exercising the option allows the holder to buy the asset at a price lower (higher) than the current market price, which is advantageous.
- At The Money (ATM): (both call and put) is at the money if the current price of the underlying asset is equal to the option's strike price. In this scenario, exercising the option would neither result in a profit nor a loss, excluding the premium paid for the option.

It is a known fact in financial mathematics [39] that the price at time t of a derivative such as an option, with expiration T, in a market with risk-free interest rate $r > 0$ can be written as the conditional expectation of the discounted payoff function given $\mathcal{F}_t$, i.e.:

$$P_c(t,T) := e^{-r(T-t)}\mathbb{E}_{\mathbb{Q}}\left[(S_T - K)^+ | \mathcal{F}_t\right] \quad \text{for a Call Option}$$
$$P_p(t,T) := e^{-r(T-t)}\mathbb{E}_{\mathbb{Q}}\left[(K - S_T)^+ | \mathcal{F}_t\right] \quad \text{for a Put Option}$$

where the expectation is taken using a measure $\mathbb{Q}$, called equivalent martingale measure. Now, let's give the definition for Equivalent Martingale Measure:

Definition 2.1.1. Consider a Market $\left(\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\in[0,T]}, \mathbb{P}\right), (S_t)_{t\in[0,T]}, r\right)$, where $\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\in[0,T]}, \mathbb{P}\right)$ a filtered measure space, $(S_t)_{t\in[0,T]}$ a stochastic process adapted to the filtration $(\mathcal{F}_t)_{t\in[0,T]}$ that represent the value of the asset in the market, and let $r > 0$[1] the risk free interest rate in the market. Then we define $\mathbb{Q}$ to be an Equivalent Martingale Measure (EMM) if:

- $\mathbb{Q} \sim \mathbb{P}$, i.e.: $\mathbb{Q}$ is equivalent to $\mathbb{P}$
- $e^{-rt}S_t$, i.e. the discounted asset price is a Martingale with respect to $\mathbb{Q}$, so that the martingale property reads:

  $$\mathbb{E}_{\mathbb{Q}}\left[e^{-rt}S_t | \mathcal{F}_s\right] = e^{-rs}S_s, \quad \forall s,t : 0 < s < t$$

  where we remark that the expectation is taken under $\mathbb{Q}$.

Remark. Usually, in financial mathematics we refer to $\mathbb{P}$ as in definition 2.1.1 as the Empirical or Physical measure. So $\mathbb{P}$ represents the probability of events based on historical data, so that for example historical returns or volatility data is measured using the physical measure $\mathbb{P}$. Also, if we would want to predict future market data, then one would use the physical measure, instead when it comes to pricing then one uses the Equivalent Martingale Measure.

[1] Usually, the interest rate is considered positive. However in the last years also this assumption has had some troubles, which are outside the scope of our thesis.

For example, if we consider the Black Scholes model, one can find explicitly the equivalent martingale measure. Remember that ,[39] the price process S_t under $\mathbb{P}$ follows a Geometric Brownian Motion, with dynamics:

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t, \quad \mu \in \mathbb{R}, \sigma \in \mathbb{R}^+$$

So now, apply Ito's product rule to the discounted asset price:

$$\begin{aligned} d\left(e^{-rt}S_t\right) &= -re^{-rt}S_t dt + e^{-rt}d\left(S_t\right) \\ &= \left(\mu - r\right)e^{-rt}S_t dt + \sigma e^{-rt}S_t dW_t \end{aligned}$$

So that if we want the discounted asset price to be a martingale with respect to $\mathbb{Q}$, we need to "kill" the drift by choosing $\mu = r$ i.e.:

$$\frac{dS_t}{S_t} = rdt + \sigma dW_t$$

Under such process, applying Ito's formula then one obtains that

$$S_t = S_0 \exp\left(\left(r - \frac{\sigma^2}{2}\right)t + \sigma W_t\right)$$

Thus, S_t is log-normal, and hence one can easily compute the value of a call or a put option. We will denote as $C_{BS} := C_{BS}\left(S_0, T, K, r, \sigma\right)$ the price under Black-Scholes model of a call option expiring at time T, at strike K with the current underlying price being S_0 and with risk-free interest rate r. Notice that C_{BS} depends also on the parameter σ which is the only parameter that is a-priori unknown. Thus, the following definition naturally arise:

Definition 2.1.2 (Implied Volatility)**.** Let C_{MKT} (P_{MKT}) be the observed market price of a call (put) option with maturity T, strike price K, in a market with risk-free interest rate $r > 0$ while the underlying is worth S_0. Then we define the Implied Volatility (IV) as the value $\hat{\sigma} = \hat{\sigma}\left(T, k := log\frac{K}{S_0}\right) \in (0, \infty)$ for which the Black-Scholes formula yields exactly the market price of the option, i.e.:

$$\begin{aligned} \hat{\sigma}: \quad & C_{MKT} = C_{BS}\left(S_0, T, K, r, \hat{\sigma}\right), \quad \text{for a Call} \\ \hat{\sigma}: \quad & P_{MKT} = P_{BS}\left(S_0, T, K, r, \hat{\sigma}\right), \quad \text{for a Put} \end{aligned}$$

note that we will consider the implied volatility as a function of the log strike $k := log\frac{K}{S_0}$.

Note that the Implied Volatity measures the expected magnitude in the price change of the underlying. Also it is considered Implied, since its value depends not on historical data, but real time market data. In general, one could define the IV in different ways, using different models rather than the more classical Black Scholes, however here we chose to give this definition since it's the one that we will use throughout this work. If one takes the Black-Scholes formula and computes its derivative with respect to the volatility, then with simple mathematics would see that it's always positive, meaning that the BS formula is strictly increasing. Hence the solution of the equation defining the IV has always a unique solution. In order to compute the IV, one needs to use root-finding algorithms such as bisection or Newton's method since an analytic expression does not exist.
The Option Chain of an asset is the set of all the options that are quoted on the market with a specific underlying. Those can be ordered in this way:

- Expiry
- For every expiry, by Strike price

then for a given option market one would have $\{C_{MKT}\left(T_i, K_j\right)\}_{i\in I, j\in J}$ i.e. the set of all call prices quoted on the market. So that if one solves the Implied Volatility equation for every maturity and strike, one would get in the end a surface, which is called Implied Volatility surface.
Now, for a given surface, we can imagine to do cross sections: if we fix the time to maturity, then we will have a function of the strike price and vice versa.

2.1.1. At The Money Skew

The ATM volatility skew has a precise mathematical definition i.e. the derivative with respect to the log-strike price $k := \log \frac{K}{S_0}$ of the Implied volatility function $\hat{\sigma}$ evaluated at zero, in formulas:

$$\psi(\tau) = \left| \frac{\partial \hat{\sigma}(\tau, k)}{\partial k} \right|_{k=0}$$

Where $\hat{\sigma}$ is the implied volatility of an option with maturity τ and log-strike k. Note that this quantity can be extracted from call and put options prices. One can search for the best-fitting curve of the empirical data. However, solving this problem is not as straightforward as it may seem, as discussed in the paper from Gatheral [28]. Gatheral suggests that the best fit can be represented by a power-law equation of the form $\psi \sim A\tau^{-\alpha}$, where $\alpha \in [-0.44, -0.3]$ and $A > 0$. In fact [31] conclude that the power law behaviour fits the data well for maturities from 1 month to a few years, while stays finite for short maturities instead of blowing up. To the same conclusion arrives also Delemotte et Al, in the paper [19], where they propose to represent the at the money skew as two different power laws: one for short maturities, smaller than one or two months with exponent equal to ≈ -0.3, and for longer, with exponent ≈ 0.49. They note that two different models, precisely a two factor Bergomi model and a two-power-law model can be used to extrapolate the ATM skew for short maturities, however, the two factor Bergomi yields a finite skew for $T = 0$ while the other model is infinite. They say that at least for the SP500 index, this question of the explosiveness or not of the ATM skew could be hard to disambiguate. [19] Also proposed both regime-switching models or multi factor models which have the flexibility of decoupling long and short behaviour of ATM skew.

2.1.2. Volatility Smile, Leverage Effect and Zumbach Effect

In this section we are going to present the nomenclature for some stylised facts of volatility. Starting from the the first, when we talk about the volatility smile, we are referring to the typical shape of the volatility surface once we fix the expiration date. One example of such phenomenon can be seen in figure 2.1.

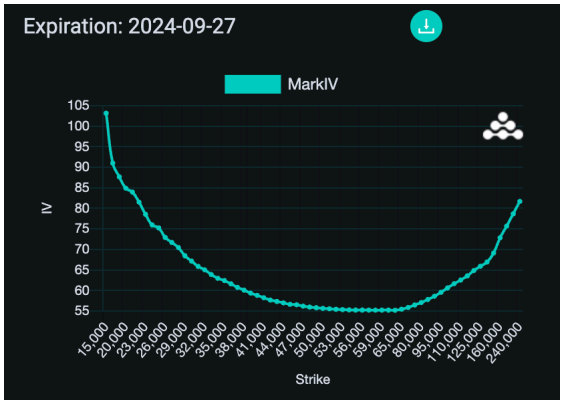

Figure 2.1: Implied Volatility smile of options with underlying Bitcoin, here the current underlying price is $69.376\$$, snapshot taken from [20].

The smile, comes from the fact that since the 1987 market crash, the implied volatility of out of the money put and call options is consistently higher than the at the money options. This could be explained economically by heuristic arguments such as that the high demand of OTM options drive up the price, or that the fat tails of the return distribution imply a higher chance of those options getting in the money, with a consequent higher price.

As for the Leverage Effect, we refer to the fact that returns and volatility are inversely correlated: the name leverage was introduced by Black, from the fact that if a stock price drops in value, then the company's equity will diminish, while its debt remain the same, making it more "leveraged". This explanation however, would be later discarded, and Zumbach says that downward movements in the stock price are seen as unfavourable events, triggering sell orders, while since most of market's partecipants hold a long position, if the price increase makes less of an impact.

Lastly, when we talk about the Zumbach's effect, we refer to the fact that pronounced price trends, regardless of the direction, increase the volatility, in other words, this could be reformulated with saying that past squared returns forecast future volatility better than past volatilities forecast future returns. This could be explained via the fact that if a trend persists, then most investors would have to re-balance their portfolio, unlike cases where the price remains in a narrow range. In a proper mathematical definition, it has been observed that financial time series, are not statistically symmetrical when future and past are interchanged [12]. In particular, consider σ_t^2 the true integrated variance of day t and let r_t be the corresponding daily return of some asset. Then, as in [24], define the statistic:

$$\mathcal{C}(\tau) := \langle \left(\sigma_t^2 - \langle \sigma_t^2 \rangle\right) r_{t-\tau}^2 \rangle, \quad \tau = k\delta, \quad k \in \mathbb{N} \tag{2.1}$$

where we denote here with $\langle \cdot \rangle$ the empirical mean and with δ the length of a trading day. In words, the statistic $\mathcal{C}$ describes the covariance between the current integrated variance, and the past squared returns. With this statistic, then one can define a measure for the Time Reversal Asymmetry, defined first in [17] as $\mathcal{Z}$:

$$\mathcal{Z}(\tau) := \mathcal{C}(\tau) - \mathcal{C}(-\tau), \quad \tau > 0 \tag{2.2}$$

It has been empirically observed, that the this measure is non-zero, see e.g. [17],[24].

So, concluding this section, we outlined all the properties that we would like our volatility model to satisfy.

3

Rough Volatility Models

In this chapter, we aim to present various Rough Volatility models, explain their derivation, and discuss the motivations for using them in volatility modeling.

Previously, we introduced the Black-Scholes model, which has a significant limitation: it assumes that volatility remains constant over the life of the contract. This assumption is unrealistic, prompting the development of alternative approaches. In particular, stochastic volatility models have been introduced, where volatility is modeled as a random variable. These models address the shortcomings of the constant volatility assumption in the Black-Scholes framework. Volatility then takes the form:

$$\sigma : [0, T] \times \Omega \to \mathbb{R}^+$$

and can be diffusion driven by a separate Brownian Motion, possibly correlated with the Brownian motion driving the price process. We refer to [21] for a review of different stochastic volatility models. To give an idea of the form of this model, one could consider the following equation:

$$\begin{aligned} dS_t &= \sqrt{\sigma_t} S_t dW_t \\ d\sigma_t &= b\left(t, \sigma_t\right) dt + d\left(t, \sigma_t\right) dB_t \end{aligned}$$

with B, W possibly correlated Brownian Motion, and b, d satisfying at least a linear growth and Lipschitsz conditions so that existence and uniqueness of a solution is guaranteed. Also, one needs to ask the process σ_t to be always positive, for a complete review of stochastic volatility models we refer to [21]. Those models lead to processes that have regularity close to the one of Brownian Motion. It is a known fact that Brownian motion has sample paths that are almost surely α-Hölder continuous $\alpha \in (0, 1/2)$, i.e. being W_t a Brownian motion we have that $\forall \alpha \in (0, 1/2), \exists C > 0$ such that $\forall t, h \in \mathbb{R}_0^+$:

$$|W_{t+h} - W_t| \leq C h^\alpha \quad \forall \alpha \in (0, 1/2) \quad \textit{a.s.}$$

However, in the statistical analysis of Gatheral et Al [28], they show that the realised variance using high frequency data has a rougher behaviour than usual Brownian Motion. In fact they show that paths of realised volatility are α-Hölder continuous with $\alpha < 1/2$. To do this, assuming that we can have N discrete observation of the volatility on a time-grid with mesh Δ on $[0, T]$ they compute for different values of q the quantity:

$$m\left(q, \Delta\right) := \frac{1}{N} \sum_{k=1}^{N} |\log\left(\sigma_{k\Delta}\right) - \log\left(\sigma_{(k-1)\Delta}\right)|^q$$

Assuming stationarity of the volatility process and that the law of large numbers can be applied, then this would be the empirical counterpart of the quantity:

$$\mathbb{E}\left[|\log\left(\sigma_{k\Delta}\right) - \log\left(\sigma_0\right)|^q\right]$$

What they observe, is that if you plot $\log\left(m\left(q, \Delta\right)\right)$ as a function of $\log \Delta$ the points lie on a straight

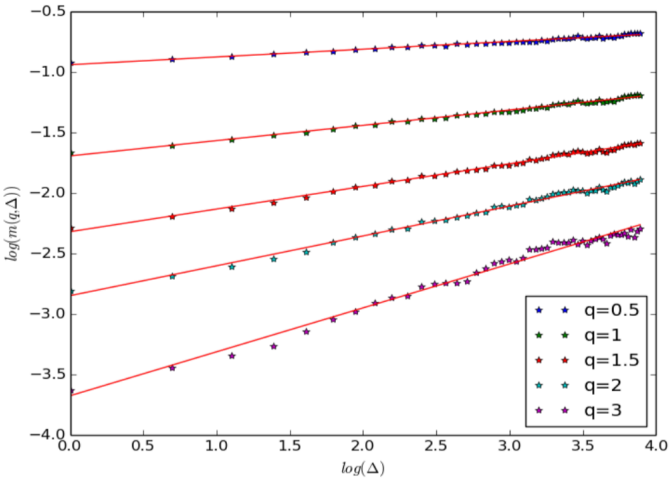

Figure 3.1: log $m(q, \Delta)$ as a function of log Δ of DAX index, Figure taken by [28, Sect. 2 Pg. 9].

line, as shown in Figure 3.1, so yielding an equation of the form:

$$\mathbb{E}\left[|\log\left(\sigma_{k\Delta}\right) - \log\left(\sigma_0\right)|^q\right] \approx K_q \Delta^{\zeta_q}$$

most noticeably, they note that $\zeta_q \approx Hq$, with $H \in [0.02, 0.08]$. This fact, has implications on the regularity of the log-volatility process, in particular it implies that the volatility process is $\alpha-$Holder continuous with $\alpha = H$.
In order to arrive to a path with such regularity, what Gatheral et Al propose in their pioneering paper [28], is to model volatility as a process which is driven not by a classical Brownian Motion, but use a Fractional Brownian motion with Hurst parameter equal to $H \in (0, 1/2)$. Let us now define Fractional Brownian Motion as in [9]:

Definition 3.0.1 (Mandelbrot-Van Ness Fractional Brownian Motion)**.** [9, Pg. 890] We define as Mandelbrot-Van Ness Fractional Brownian Motion with Hurst parameter $H \in (0, 1)$ the process

$$\hat{W}_t^H := C_H \left\{ \int_{-\infty}^{t} \frac{dW_s}{(t-s)^{1/2-H}} - \int_{-\infty}^{0} \frac{dW_s}{(-s)^{1/2-H}} \right\}$$

$$C_H := \sqrt{\frac{2H\Gamma\left(3/2 - H\right)}{\Gamma\left(H + 1/2\right)\Gamma\left(2 - 2H\right)}}$$

Where W_t is Brownian Motion.

For this choice of C_H we have that:

$$\mathbb{E}\left[\hat{W}_t^H \hat{W}_s^H\right] = \frac{1}{2}\left[t^{2H} + s^{2H} - |t-s|^{2H}\right]$$

As a first and simple approach, Gatheral in [28] proposes the Rough Fractional Stochastic Volatility (RFSV) model that has path-wise regularity similar to the one found empirically, id est of path-wise α-Hölder continuity with $\alpha \in [0.02, 0.08]$. The model takes the following form [28, Pg. 15]:

$$\sigma_t = \sigma_0 \exp\left(X_t\right)$$
$$dX_t = \nu d\hat{W}_t^H - a\left(X_t - m\right)dt$$

Where $\sigma_0, \nu, a \in \mathbb{R}^+$, and $m \in \mathbb{R}$ and most importantly, $H \in (0, 1/2)$. Note here that X_t is in fact a Ornstein–Uhlenbeck process that is driven by a fractional Brownian Motion instead of a classical Brownian Motion. X_t also allows an integral representation of the following form:

$$X_t = \nu \int_{-\infty}^{t} e^{-a(t-s)} d\hat{W}_s^H + m \tag{3.1}$$

The main idea for defining and constructing this model is the following: to try to fit the high frequency market data that shows how the volatility behaves at low time scales like fractional Brownian Motion. From a theoretical point of view, Gatheral proves the following proposition:

Proposition 1. [28, Prop. 3.1 Pg. 15] Let $\hat{W}_t^H$ be fractional Brownian Motion, and X^a as in (3.1) for a given $a > 0$, then one has:

$$\lim_{a\to 0} \mathbb{E}\left[\sup_{t\in[0,T]} \left|X_t^a - X_0^a - \nu \hat{W}_t^H\right|\right] = 0$$

this proposition shows that for $a \ll 1/T$ then the RFSV has a volatility that behaves like fractional Brownian Motion. This model, has been the starting point of Rough Volatility modelling, we will see in the next section how a slight modification of this model extends to a non-Markovian version of the Bergomi model.

3.1. rBergomi

In this section we are going to present how Bayer in [9] starting from the ideas given in [28] arrives to a non-Markovian version of the well known Bergomi model. First, we are going to present the classical Bergomi model, and then, we are going to present the work of [9] where the Rough Bergomi has been first introduced. First, the classical Bergomi model has been introduced first in [11]. The goal was to find a model with which one could consistently price derivatives with underlying the variance. In particular one wishes to model the forward variance curve defined as:

$$\xi_T(t) = \mathbb{E}\left[V_T|\mathcal{F}_t\right] \tag{3.2}$$

Where V_T is the spot volatily at time T. It is clear that the volatility can be expressed in terms of the forward variance as $\xi_\tau(\tau) = V_\tau, \forall \tau > 0$. In particular, Bergomi chooses the following structure for the forward variance [11, Pg.2]:

$$d\xi_T(t) = \omega e^{-k_1(T-t)} dW_t$$

where $k_1 \in \mathbb{R}^+, \omega \in \mathbb{R}$ and W is Brownian Motion. This SDE has a closed form solution of type:

$$\xi_T(t) = \xi_T(0) \exp\left(\omega e^{-k_1(T-t)} X_t - \frac{\omega^2}{2} e^{-2k_1(T-t)} \mathbb{E}\left[X_t^2\right]\right) \tag{3.3}$$

And the process X_t is an Orhnstein-Uhlenbeck process of the form:

$$X_t = \int_0^t e^{-k_1(t-u)} dW_u$$

so one can also specify the joint dynamics of the underlying and the volatility (S, V) by putting:

$$dS_t = rS_t dt + \sqrt{\xi_t(t)} S_t dZ_t$$

and $\xi_t(t)$ as in (3.3), Z_t a Brownian Motion possibly correlated with the Brownian Motion driving the forward variance curve $\xi_T(t)$ and $r > 0$.

Let us briefly introduce the concept of stochastic exponential, that will simplify the notation.

Definition 3.1.1. For a continuous (semi) Martingale Z, the stochastic exponential $\mathcal{E}(\cdot)$ is defined as the process given by:

$$\mathcal{E}(Z)_t = \exp\left(Z_t - Z_0 - [Z, Z]_{0,t}\right)$$

If Z is a local martingale, then so is $\mathcal{E}(Z)$. Moreover, if Z is a zero mean Gaussian random variable, that allows a representation of the form:

$$Z_t = \int_0^T f(s) dW_s$$

where f is a deterministic function and W is Brownian Motion, then the stochastic exponential of Z can be rewritten as:

$$\mathcal{E}(Z)_t = \exp\left(Z_t - \frac{1}{2}\mathbb{E}\left[Z_t^2\right]\right)$$

The bottom line here, that the definition of the stochastic integral allows for a simpler representation of the Bergomi model, in fact one can rewrite (3.3) as:

$$d\xi_T(t) = \xi_T(0)\mathcal{E}\left(\omega \int_0^t e^{-k_1(T-u)} dW_u\right) \tag{3.4}$$

The Bergomi model that we have just presented, is called one-factor Bergomi, since there is only one driver of the forward variance curve. This naturally leads to a generalisation of the model to n factors. Given $\omega_i \in \mathbb{R}$ and $k_i \in \mathbb{R}^+$, for $, i = 1, \ldots, n$ then one can define the n-factor Bergomi as in [11, Pg.4], where the forward variance curve takes the form:

$$d\xi_T(t) = \xi_T(0)\mathcal{E}\left(\sum_{i=1}^{n} \omega_i \int_0^t e^{-k_i(T-u)} dW_u^i\right) \tag{3.5}$$

Where also $\rho(W^i, W^j) \in [-1, 1]$, the correlation parameter between the different Brownian Motion driving the forward variance curve are chosen at will. Before moving on and presenting the Rough Bergomi, we remark that the n-factor Bergomi model produces an At The Money volatility skew that has approximately the form, [9, Pg.889]:

$$\psi(\tau) \approx \sum_{i=1}^{n} \frac{\omega_i}{k_i \tau}\left\{1 - \frac{1 - e^{-k_i\tau}}{k_i\tau}\right\} \tag{3.6}$$

this, however, as we have seen, is inconsistent with high frequency market data, that shows a power-law dependence in τ.
Now that we have introduced the classical Bergomi model, from the one factor to the multi-factor Bergomi, we can move on and present how the Rough Bergomi arises in a natural way by switching from the physical measure $\mathbb{P}$ to an equivalent martingale measure $\mathbb{Q}$ in the RFSV model.
As stated before, Gatheral in [28] proposes to model log-volatility under the physical measure $\mathbb{P}$ as a Mandelbrot-Van Ness Fractional Brownian Motion, this means that for $u > t$, and for some positive parameter $\nu > 0$, denoting the square root of the volatility with σ then one has:

$$log\sigma_u - log\sigma_t = \nu\left(\hat{W}_u^H - \hat{W}_t^H\right)$$

Then the increments of the square of the volatility under the physical measure $\mathbb{P}$ can be written as [9, Pg.890]:

$$\begin{aligned} log\sigma_u^2 - log\sigma_t^2 &= 2\nu C_H\left\{\int_{-\infty}^{u} \frac{dW_s^{\mathbb{P}}}{(u-s)^{1/2-H}} - \int_{-\infty}^{t} \frac{dW_s^{\mathbb{P}}}{(t-s)^{1/2-H}}\right\} \\ &= 2\nu C_H\left\{\int_t^u (u-s)^{H-1/2} dW_s^{\mathbb{P}} + \int_{-\infty}^{t}\left[\frac{1}{(u-s)^{1/2-H}} - \frac{1}{(t-s)^{1/2-H}}\right] dW_s^{\mathbb{P}}\right\} \\ &:= 2\nu C_H\left[M_t(u) + Z_t(u)\right] \end{aligned} \tag{3.7}$$

With C_H as in the definition of the Mandelbrot-Van Ness fractional Brownian Motion 3.0.1. Now, $Z_t(u)$ is $\mathcal{F}_t-$measurable, while $M_t(u)$ is independent of $\mathcal{F}_t$, Gaussian, with mean zero and variance $(u-t)^{2H}/2H$. Note that, when we compute option prices, we are interested in taking conditional expectations say w.r.t. $\mathcal{F}_t$, Hence the quantity $Z_t(u)$ would be a known quantity. So now, it is natural to introduce the Rienmann-Liouville Fractional Brownian Motion.

Definition 3.1.2 (Fractional Brownian Motion)**.** We define as Riemann-Liouville Fractional Brownian Motion (fBm) with Hurst parameter H the following process:

$$W_t^H := \sqrt{2H}\int_0^t (t-s)^{H-1/2} dW_s \quad t \in [0, T]$$

Remark. Note here that the constant in front of the stochastic integral is put so that the variance of W_t^H is equal to t^{2H}, exactly as in the Mandelbrot-Van Ness Fractional Brownian Motion.

Now, we want to give an expression for the forward variance curve under the physical measure $\mathbb{P}$, incorporating also the definition of fractional Brownian Motion given above, $\forall u > t$ starting from (3.7), that is equivalent of saying:

$$\sigma_u^2 = \sigma_t^2 \exp\left(2\nu C_H \left[M_t(u) + Z_t(u)\right]\right) \tag{3.8}$$

then if one conditions on $\mathcal{F}_t$, reminding the $\mathcal{F}_t$ measurability of $Z_t(u)$, and from the fact that $M_t(u)$ is a mean zero, Gaussian random variable independent of $\mathcal{F}_t$ then one has that the forward variance curve has the form:

$$\begin{aligned}
\mathbb{E}_{\mathbb{P}}\left[\sigma_u^2 | \mathcal{F}_t\right] &= \sigma_t^2 \mathbb{E}_{\mathbb{P}}\left[\exp\left(2\nu C_H \left[M_t(u) + Z_t(u)\right]\right) | \mathcal{F}_t\right] \\
&= \sigma_t^2 \exp\left(2\nu C_H Z_t(u)\right) \mathbb{E}_{\mathbb{P}}\left[\exp\left(2\nu C_H M_t(u)\right)\right] \\
&= \sigma_t^2 \exp\left(2\nu C_H Z_t(u)\right) \mathbb{E}_{\mathbb{P}}\left[\exp\left(\frac{2\nu C_H}{\sqrt{2H}}\left(W_u^H - W_t^H\right)\right)\right]
\end{aligned}$$

now, putting $\eta := \frac{2\nu C_H}{\sqrt{2H}}$ we have, as in [9, Pg. 890]:

$$\mathbb{E}_{\mathbb{P}}\left[\sigma_u^2 | \mathcal{F}_t\right] = \sigma_t^2 \exp\left[2\nu C_H Z_t(u) + \frac{1}{2}\eta^2 \mathbb{E}\left[\left|W_u^H - W_t^H\right|^2\right]\right]$$

so that, combining the last equation and (3.8) then one has:

$$\sigma_u^2 = \sigma_t^2 \exp\left(\eta\left(W_u^H - W_t^H\right) + 2\nu C_H Z_t(u)\right)$$

which can also be rewritten using the stochastic exponential:

$$\sigma_u^2 = \mathbb{E}_{\mathbb{P}}\left[\sigma_u^2 | \mathcal{F}_t\right] \mathcal{E}\left(\eta\left(W_u^H - W_t^H\right)\right) \tag{3.9}$$

Now, always in [9], they change measure, passing from the physical measure $\mathbb{P}$ to the equivalent martingale measure $\mathbb{Q}$. Using Girsanov's theorem, they set:

$$dW_s^{\mathbb{P}} = dW_s^{\mathbb{Q}} + \mu_s ds$$

for some deterministic function of time μ_s. Then, one can rewrite (3.9) as:

$$\begin{aligned}
\sigma_u^2 &= \mathbb{E}_{\mathbb{P}}\left[\sigma_u^2 | \mathcal{F}_t\right] \exp\left(\eta\sqrt{2H}\int_t^u \frac{dW_s^{\mathbb{P}}}{(u-s)^{1/2-H}} - \frac{\eta^2}{2}(u-t)^{2H}\right) \\
&= \mathbb{E}_{\mathbb{P}}\left[\sigma_u^2 | \mathcal{F}_t\right] \exp\left(\eta\sqrt{2H}\int_t^u \frac{dW_s^{\mathbb{Q}}}{(u-s)^{1/2-H}} + \eta\sqrt{2H}\int_t^u \frac{\mu_s ds}{(u-s)^{1/2-H}} - \frac{\eta^2}{2}(u-t)^{2H}\right) \qquad (3.10) \\
&= \mathbb{E}_{\mathbb{P}}\left[\sigma_u^2 | \mathcal{F}_t\right] \mathcal{E}\left(\eta\sqrt{2H}\int_t^u \frac{dW_s^{\mathbb{Q}}}{(u-s)^{1/2-H}}\right) \exp\left(\eta\sqrt{2H}\int_t^u \frac{\mu_s ds}{(u-s)^{1/2-H}}\right) \qquad (3.11)
\end{aligned}$$

Finally, one can arrive to the forward variance curve, which by definition is

$$\xi_t(u) = \mathbb{E}_{\mathbb{Q}}\left[\sigma_u^2 | \mathcal{F}_t\right]$$

then assuming that the filtration generated by $W^{\mathbb{Q}}$ coincides with the one generated by $W^{\mathbb{P}}$, then one has that the forward variance curve has the following form:

$$\xi_t(u) = \mathbb{E}_{\mathbb{P}}\left[\sigma_u^2 | \mathcal{F}_t\right] \exp\left(\eta\sqrt{2H}\int_t^u \frac{\mu_s}{(u-s)^{1/2-H}} ds\right) \tag{3.12}$$

Note here the striking resemblance of the Rough Bergomi with the classical one specified in (3.3), also, now we have that Fukasawa, in [26] shows that such model implies an ATM skew of the desired form $\psi(\tau) = \tau^{-\gamma}$ for small τ.

What Bayer obtained, is a model which is capable with only three parameters to describe and capture some characteristics of the market:

- H, the Hurst parameter controls the roughness of the path, and also controls the explosiveness of the ATM Skew i.e. $\psi(\tau) = \tau^{-1/2+H}$
- Once H is fixed, the product $\rho\nu$ can be fixed by seeing the level of the ATM skew ψ for larger expirations.
- ρ can be guessed by keeping in mind that it moves the minimum of the volatility smile down and to the right as it becomes more negative. And is the parameter which can capture the negative correlation between asset returns and volatility.

However, the rBergomi model is although far from being a flawless model. In fact, as noted by Abi Jaber, in a recent paper [2], he calibrates the model to different market periods, and he observes that the correlation parameter ρ tends to saturate at the value $\rho = -1$, making it somewhat of a redundant parameter.

Now that we have introduced the Rough Bergomi model, giving the theoretical derivation and explaining its parameters, we can move on and introduce the Rough Heston model.

3.2. rHeston

In this section we are going to present the Rough Heston model. In similar fashion for what we have done in the section on the Rough Bergomi, let us first introduce the classical Heston model, where the volatility process follows a Cox-Ingersoll-Ross process of the form under the equivalent martingale measure $\mathbb{Q}$, with starting condition $V_0 = \hat{V}_0$ [39, Pg.231]:

$$dV_t = \kappa(\bar{v} - V_t)\,dt + \theta\sqrt{V_t}dW_t$$

and the price process with starting condition $S_0 = \hat{S}_0$:

$$dS_t = rS_t dt + \sqrt{V_t}S_t dZ_t$$

where $\kappa, \bar{v}, \theta \in \mathbb{R}^+$, and correlated Brownian motions Z, W with correlation $\rho \in [-1, 1]$ and $r > 0$ is the risk-free interest rate of the market. Now we will introduce the rough counterpart, that was introduced first by El Euch and Rosenbaum, in [23]. They require the price process and the variance process to follow:

$$S_t = S_0 + \int_0^t \sqrt{V_s}S_s dZ_s$$
$$V_t = V_0 + \frac{1}{\Gamma(H+1/2)}\int_0^t (t-s)^{H-1/2}\lambda(\theta - V_s)\,ds + \frac{1}{\Gamma(H+1/2)}\int_0^t (t-s)^{H-1/2}\nu\sqrt{V_s}dW_s$$

Where $H \in (0, 1/2]$ is the Hurst parameter, $\lambda > 0$ is the mean reversion, $\nu > 0$ is the volatility of volatility, $\theta > 0$ is the long term mean mean reversion parameter and (W, Z) is a ρ-correlated Brownian motion with $\rho \in [-1, 1]$

Now, the novelty here stands in the presence of the fractional kernel inside the integrals that yields a rough behaviour of the volatility, that becomes Hölder continuous with Holder parameter equal to H. The fractional kernel $g(\tau) = \frac{1}{\Gamma(H+1/2)}\tau^{H-1/2}$ introduces a dependence structure in the volatility process V. However, this implies that the resulting process V is neither a Markov Process, nor a semi-martingale. This of course leads to problems in terms of both simulation and theoretical analysis. The popularity of the classical Heston model is due to the fact that it is able to reproduce different phenomena characterising the market, such as leverage effect, Zumbach effect, time varying volatility and fat tails. The most interesting aspect of the rough counterpart, is the micro-structural foundations [22],[32], where they show that the rough nature of volatility is emerging from the behaviour of market partecipants at high frequency. To do this in [22], they model the price process as a sum of bi-dimensional Hawkes process, i.e.:

$$P_t = N_t^+ - N_t^-$$

where $N_t := \left(N_t^+, N_t^-\right)$ is a bi-dimensional Poisson process with time-dependent and self-exciting intensities $\lambda_t := \left(\lambda_t^+, \lambda_t^-\right)$ defined by:

$$\begin{pmatrix}\lambda_t^+ \\ \lambda_t^-\end{pmatrix} = \begin{pmatrix}\mu_t^+ \\ \mu_t^-\end{pmatrix} + \int_0^t \begin{pmatrix}\varphi_1(t-s) & \varphi_3(t-s) \\ \varphi_2(t-s) & \varphi_4(t-s)\end{pmatrix} \cdot \begin{pmatrix}dN_t^+ \\ dN_t^-\end{pmatrix}$$

where $\mu^{\pm}$ are positive constant and the function:

$$\phi := \begin{pmatrix} \varphi_1(t-s) & \varphi_3(t-s) \\ \varphi_2(t-s) & \varphi_4(t-s) \end{pmatrix} : \mathbb{R}_+ \to \mathcal{M}^2(\mathbb{R}_+)$$

is a kernel matrix whose components φ_i are positive and locally integrable. The idea now, is that with a model like this, it is apparently easy to encode stylised properties of the market. We will present those characteristics but not how they are embedded in such a model, directly citing [22]:

- Markets are highly endogenous, meaning that most of the orders have no real economic motivation but are rather sent by algorithms in reaction to other orders.
- Mechanisms preventing statistical arbitrages take place on high frequency markets. In- deed, at the high frequency scale, building strategies which are on average profitable is hardly possible.
- There is some asymmetry in the liquidity on the bid and ask sides of the order book. This simply means that buying and selling are not symmetric actions.
- significant proportion of transactions is due to large orders, called meta-orders, which are not executed at once but split in time by trading algorithms.

After encoding those properties in the tick-by-tick price process, they manage to prove that this model, after a rescaling of the price process it converges weakly to a rough Heston model. Also, it has been shown both theoretically and numerically in [24] that the Rough Heston model displays the Zumbach effect, which also is a stylised fact of volatility and financial time series. Until now, we have discussed the motivations to use the Rough Heston model for describing volatility in financial time series.

Another interesting result, that will allow us in section 7 to perform precise numerical experiments is the fact that the characteristic function of the rough Heston model is known, and in particular it has been obtained exactly via the limit of Hawkes processes. To retrieve the characteristic function however, one must solve a fractional Riccati equation, compared to the Riccati-type ODE of the classical Heston. This means that we can use Fourier pricing methods such as the COS method [25] or the Carr-Madam option pricing [16].

3.3. Truncated Brownian Semistationary Processes

Let us define in this section both Brownian-Semi Stationary processes (BSS), and the truncated Brownian Semi-Stationary processes (TBSS) which are a large class of models, which include also processes like Fractional Brownian Motion. We take the definition of such processes by [10]. Let us consider a filtered probability space $(\Omega, \mathcal{F}, (F_t)_{t\in\mathbb{R}}, \mathbb{P})$, supporting a Brownian motion $(W_t)_{t\in\mathbb{R}^+}$, $(\sigma_t)_{t\in\mathbb{R}^+}$ is an $\mathcal{F}_t$ predictable stochastic process with locally bounded trajectories and g is a kernel function, positive, Borel measurable and in $\mathcal{L}^2(\mathbb{R}^+)$. Also, they assume σ to be covariance stationary with finite second moments, in the sense that

$$\mathbb{E}\left[\sigma_t^2\right] < \infty, \quad \mathbb{E}\left[\sigma_t\right] = \mathbb{E}\left[\sigma_s\right], \quad Cov\left[\sigma_t, \sigma_s\right] = Cov\left[\sigma_0, \sigma_{|t-s|}\right], \quad \forall t, s \in \mathbb{R}^+ \tag{3.13}$$

We will refer as the function g as the Kernel and more specifically we are interested in the case where g behaves like a power function near the origin, meaning

$$g(t) \propto x^{\alpha}, \quad \alpha \in (-1/2, 1/2) \tag{3.14}$$

under this assumption, the process X_t behaves locally like Fractional Brownian Motion with Hurst parameter $H = \alpha + 1/2 \in (0,1) \setminus \{1/2\}$. They assume, that the kernel function also satisfies those three additional hypotheses:

- $\exists \alpha \in (-1/2, 1/2) \setminus \{0\}$ such that:

$$g(x) = x^{\alpha} L_g(x), \quad x \in (0,1] \tag{3.15}$$

  where L_g is positive, continuously differentiable, slowly varying at 0 and bounded away from 0. Also $\exists C \geq 0$ s.t. the derivative L_g' satisfies:

$$\left|L_g'\right| \leq C\left(1 + x^{-1}\right), \quad x \in (0,1] \tag{3.16}$$

- g is continuously differentiable on $(0,\infty)$, with ultimately monotonic derivative such that $g' \in \mathcal{L}^2((1,\infty))$
- $\exists \beta \in (-\infty, -1/2)$ s.t.

$$g(x) = O(x^{\beta}), \quad x \to \infty \tag{3.17}$$

Definition 3.3.1 (Brownian Semi-Stationary Process)**.** With the assumptions (3.13)- (3.17) on g and σ, we define a Brownian Semi-Stationary process (BSS) as a process of the form:

$$X_t = \int_{-\infty}^{t} g(t-s)\,\sigma_s dW_s$$

However, for financial applications, Truncated Brownian Semi-Stationary processes (TBSS) are sufficient. The name Truncated, comes from the fact that the the integration is being done only on intervals starting from zero, while all the assumptions made on g, σ remains the same as for BSS:

Definition 3.3.2 (Truncated Brownian Semi-Stationary Process)**.** A Truncated Brownian Semi-Stationary Process is a process of the form:

$$X_t = \int_{0}^{t} g(t-s)\,\sigma_s dW_s$$

An example of Brownian Semi-Stationary process is the Riemann-Liouville fractional Brownian motion, in fact we have that, for $H \in (0, 1/2)$ and $\forall t > 0$:

$$W_t^H = \sqrt{2H} \int_0^t (t-s)^{H-1/2} dW_s$$

so that the kernel $g(\tau) = \sqrt{2H}(\tau)^{H-1/2}$, while in this case $\sigma_s \equiv 1$. One can easily check that the hypotheses (3.13)-(3.17) are in fact satisfied.

4

Simulating Rough Volatility

In this chapter, we will introduce two exact methods for simulating rough volatility models, as well as the state-of-the-art approach, which, to the best of our knowledge, remains the Hybrid Scheme of Bennedsen, Lunde, and Pakkanen [10]. First, in the next section, we will begin with exact types of schemes.

4.1. Exact Schemes

In this section, we will introduce two types of exact schemes. By "exact schemes," we refer to methods that simulate the covariance structure of a process precisely, without any approximations. The primary advantage of these schemes is that the main source of error stems only from discretization. However, they come with significant drawbacks: either the computational time increases quadratically with the number of discretization steps, or the memory usage grows quadratically with the number of discretization steps.

We will start by presenting the concept outlined in the paper where the Rough Bergomi model was first introduced [9]. Say that we want to simulate increments of fractional Brownian Motion with Hurst parameter $H \in (0, 1/2)$, then one has that:

$$W_t^H = \sqrt{2H} \int_0^t (t-s)^{H-1/2} \, dW_s$$

note that the dependence structure of W_t^H is fully specified, in fact $\forall v > u$ [9, Pg.892]:

$$\mathbb{E}\left[W_v^H W_u^H\right] = u^{2H} G\left(\frac{v}{u}\right) \tag{4.1}$$

$$G(x) = 2H \int_0^1 \frac{ds}{(1-s)^{1/2-H} (x-s)^{1/2-H}}, \quad \forall x > 1 \tag{4.2}$$

Then, always in [9] they propose a method for simulating such process on a discrete time grid $\{t_i\}_{i=0}^n$, with $0 =: t_0 < t_1 < \cdots < t_n$

- compute first the covariance matrix for the Fractional Brownian Motion W_t^H using (4.1), id est a symmetric matrix $A \in \mathbb{R}^{n \times n}$ with

$$A_{i,j} = A_{j,i} = \mathbb{E}\left[W_{t_i}^H W_{t_j}^H\right], \quad \forall i, j : \; i > j \in \{1, \ldots, n\}$$

- compute its Cholesky decomposition C, so a lower triangular matrix such that

$$A = CC^T$$

- For each time, generate i.i.d. standard normal random n-dimensional vector Z and multiply them by the lower triangular matrix obtained with the Cholesky decomposition. So that one has:

$$B = CZ$$

Then, the vector B is a Gaussian vector with zero mean and covariance equal to A. This means that B has in fact the same distribution of the increments of fractional Brownian Motion

So if we would want to compute the dependence structure for the It is clear that this naive algorithm suffers from two caveat: if one needs a fine time grid, then the Cholesky decomposition becomes computationally unfeasible. In fact remember that in order to compute the Cholesky decomposition of an $n \times n$ matrix, one need $O\left(n^3\right)$ floating point operations. Moreover, if one wants to simulate enough sample paths say M, then one has to compute M vector-matrix multiplications, which each one is of order $O\left(n^2\right)$. This results in a total complexity of the algorithm of n^3+Mn^2, which is clear that becomes not feasible even for a low number of simulations and time steps. To be more fair with the Cholesky approach, if one looks closely at the algorithm, then one realises that if values are cached during the computation, then only n^2 operations are needed.
Another exact method that also utilizes Cholesky decomposition is the one proposed by Davies and Harte [18]. However, this method has a significant limitation: it is only applicable to stationary processes. As a result, it cannot be applied to Riemann-Liouville fractional Brownian Motion, which is a non-stationary process. The main idea that comes into play is the fact that for circulant matrices, it is possible to compute the Cholesky decomposition using Fast Fourier Transform, hence improving the computational complexity of the algorithm from n^2 to $n \log n$ in the number of discretization steps. Let us give the definition of Circulant matrix:

Definition 4.1.1 (Circulant matrix)**.** An $m \times m$ Circulant matrix is a matrix that takes the form:

$$\begin{pmatrix} c_0 & c_{m-1} & \cdots & c_2 & c_1 \\ c_1 & c_0 & c_{m-1} & & c_2 \\ \vdots & c_1 & c_0 & \ddots & \vdots \\ c_{m-2} & & \ddots & \ddots & c_{m-1} \\ c_{m-1} & c_{m-2} & \cdots & c_1 & c_0 \end{pmatrix}$$

Say that, we want to simulate paths of the Mandelbrot-Van Ness fractional Brownian Motion with Hurst parameter H that we denote with $\hat{W}_t^H$ on a discrete time grid $\left\{t_k := \frac{k}{n}\right\}_{k=0}^n \subset [0,1]$. To see that $\hat{W}_t^H$ is stationary, note first that it is indeed a Gaussian process with mean zero, hence necessary and sufficient condition for it to be stationary, is that the variance of the increments is independent of time, i.e.:

$$\mathbb{E}\left[\left|\hat{W}_t^H - \hat{W}_s^H\right|^2\right] = \mathbb{E}\left[\left|\hat{W}_{t-s}^H\right|^2\right]$$

In fact one has:

$$\begin{aligned} \mathbb{E}\left[\left|\hat{W}_t^H - \hat{W}_s^H\right|^2\right] &= \mathbb{E}\left[\left|\hat{W}_t^H\right|^2 + \left|\hat{W}_s^H\right|^2 - 2\hat{W}_t^H \hat{W}_s^H\right] \\ &= t^{2H} + s^{2H} - \left(t^{2H} + s^{2H} - |t-s|^{2H}\right) = |t-s|^{2H} \\ &= \mathbb{E}\left[\left|\hat{W}_{t-s}^H\right|^2\right] \end{aligned}$$

Since it is a stationary process, and hence also covariance stationary, its auto-correlation structure will not depend on the time of evaluation, in fact one can define:

$$\begin{aligned} \rho_H^n(k) &:= \mathbb{E}\left[\left(W_{t_{i+1}}^H - W_{t_i}^H\right)\left(W_{t_{i+k+1}}^H - W_{t_{i+k}}^H\right)\right] \\ &= \mathbb{E}\left[W_{t_{i+1}}^H W_{t_{i+k+1}}^H - W_{t_{i+1}}^H W_{t_{i+k}}^H - W_{t_i}^H W_{t_{i+k+1}}^H + W_{t_i}^H W_{t_{i+k}}^H\right] \\ &= \frac{1}{2}(t_{i+1}^{2H} + t_{i+k+1}^{2H} - \left|\frac{k}{n}\right|^{2H} - t_{i+1}^{2H} - t_{i+k}^{2H} + \left|\frac{k-1}{n}\right|^{2H} - t_i^{2H} \\ &- t_{i+k+1}^{2H} + \left|\frac{k+1}{n}\right|^{2H} + t_i^{2H} + t_{i+k}^{2H} - \left|\frac{k}{n}\right|^{2H}) \\ &= \frac{1}{2n}\left(|k+1|^{2H} + |k-1|^{2H} - 2|k|^{2H}\right) \end{aligned}$$

Then one can put all those auto-correlation coefficients inside a matrix in the following way, as done in [41, Pg. 23]:

$$Aut(n) = \begin{pmatrix} \rho_H^n(0) & \rho_H^n(1) & \rho_H^n(2) & \cdots & \rho_H^n(n-1) \\ \rho_H^n(1) & \rho_H^n(0) & \rho_H^n(1) & \cdots & \rho_H^n(n-2) \\ \rho_H^n(2) & \rho_H^n(1) & \rho_H^n(0) & \cdots & \rho_H^n(n-3) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \rho_H^n(n-1) & \rho_H^n(n-2) & \rho_H^n(n-3) & \cdots & \rho_H^n(0) \end{pmatrix}.$$

Now then the task become to build a circulant matrix, such that in the top left corner lies the original auto-correlation matrix $Aut\,(n)$. So, if we consider a circulant matrix as in the definition 4.1.1, with $m = 2n$, and set, as in the definition:

$$\begin{aligned} c_i &= \rho_H\,(i)\,, \quad i = 0, \ldots, n-1 \\ c_i &= \rho_H\,(2n-i)\,, \quad i = n+2, \ldots, 2n-1 \end{aligned}$$

then we have obtained a circulant matrix, such that the top left $n \times n$ sub-matrix coincides with $Aut(n)$. Then, by the theorem from Brockwell [14], it is possible to compute the Cholesky decomposition of a circulant matrix C via the Fast Fourier Transform. In particular, $\exists U, \Lambda$ such that U us an unitary matrix and Λ is diagonal such that:

$$C = U\Lambda U^*$$

but then one can rewrite:

$$C = U\Lambda^{1/2}U^*U\Lambda^{1/2}U^*$$

With the matrix U such that:

$$U_{j,k} = \frac{1}{\sqrt{2n}} \exp\left(-\frac{2\pi\, i}{2n} jk\right), \quad 0 \leq j, k \leq 2n-1$$

Where i is the imaginary unit, and Λ is a diagonal matrix with diagonal $\{\lambda_k\}_{k=0}^{2n-1}$:

$$\lambda_k = \sum_{j=0}^{2n-1} c_j \exp\left(-\frac{2\pi\, i}{2n} jk\right) \tag{4.3}$$

So, if we consider the matrix $G := U\Lambda^{1/2}U^*$, and simulate a vector of length $2n$ of independent standard normal random variables, say Z, then one has that

$$X := GZ = U\Lambda^{1/2}U^*Z \sim N\,(0, C)$$

Now, since Z is standard normal distributed, then also $U^*Z =: S + iT$ will be a vector of normal random variables. Wood and Chan, in [42] prove the following properties on S, T:

Theorem 1. [42] The random vectors S, T satisfy:

1. $\mathbb{E}\,[S] = \mathbb{E}\,[T] = 0$, and $\mathbb{E}\left[ST^T\right] = 0$, so that they are in fact independent.
2. If $j \in \{0, n\}$ then $T_j = 0$ and
$$\mathbb{E}\,[S_j S_k] = \delta_{j,k}$$
where δ is the Kronecker delta.
3. If $j \notin \{0, n\}$ then
$$\mathbb{E}\,[S_j S_k] = \frac{1}{2}\delta_{j,k}, \quad \mathbb{E}\,[T_j T_k] = \frac{1}{2}\delta_{j,k} - \frac{1}{2}\delta_{2n-j,k}$$

So, using this theorem, then one can define a vector V of length $2n$ in the following way, as done in [41, Pg.25]:

$$\begin{aligned} &V_0 = A_0, \quad V_n = B_0 \\ &V_j = \frac{1}{\sqrt{2}}\left(A_j + iB_j\right), \quad V_{2n-j} = \frac{1}{\sqrt{2}}\left(A_j - iB_j\right), \quad j = 0, \dots, n-1 \end{aligned} \tag{4.4}$$

Where A, B are vectors of length n such that $A_j, B_j \sim N(0,1), \forall j = 0, \dots, n-1$. It is clear now that the vector V has the same distribution of the vector $U^* Z$ via theorem 1.

Finally, we need to compute $X = U\lambda^{\frac{1}{2}} V$, that is:

$$X_j = \sum_{k=0}^{2n-1} \frac{1}{\sqrt{2n}} \exp\left(-\frac{2\pi i}{2n} jk\right) \sqrt{\lambda_k} W_k \tag{4.5}$$

And this vector can be computed using Fast Fourier transform. So finally we give a recap of the algorithm, as done in [41]:

1. Compute $(\lambda_k)_{k=0}^{2n-1}$ using (4.3), via FFT.
2. Simulate two independent standard normal vectors A, B of size n.
3. Compute the vector V, using equation (4.4).
4. Compute X, using (4.5), using FFT.
5. Finally, retrieve the first n elements of X. Those will be fractional Brownian Motion increments.

With this method, one can simulate exactly paths of the Mandelbrot-Van Ness fractional Brownian Motion with a computational complexity of $\mathcal{O}(n \log n)$. However, in financial applications, we are interested in simulating the Riemann-Liouville fractional Brownian Motion. This process is not stationary, which prevents efficient simulation using circulant matrices.

To address this challenge, various algorithms have been proposed to approximate the Riemann-Liouville fractional Brownian Motion. Among these, the Hybrid Scheme, developed by Bennedsen et al. in [10] stands out, and to the best of our knowledge, is the state of the art for simulation of Brownian Semi-stationary processes and Truncated Brownian Semi-stationary processes. As we will see in the next section, this method has computational complexity of $\mathcal{O}(n \log n)$, and memory complexity of $\mathcal{O}(n)$, where n is the number of time steps used in the discretization.

4.2. Hybrid Scheme

This simulation method, is somewhat general in the sense that can be applied to every *Brownian Semi-Stationary* process (BSS). The main idea of the paper [10], is to capitalise on the regularity structure of the kernel and approximate its values near zero as a power law function, while using step functions elsewhere. Let us first present the method for general Brownian Semi-Stationary processes, introduced in section (3.3). In fact, let us rewrite the process X_t as done in [10, Pg.938], splitting the integral on a timebgrid $\mathcal{G}_t^n = \left\{t - \frac{k}{n}\right\}_{k \in \mathbb{N}}$. Then one can make a first approximation:

$$X_t = \sum_{k=1}^{\infty} \int_{t-\frac{k}{n}}^{t-\frac{k}{n}+\frac{1}{n}} g(t-s)\sigma_s dW_s \approx \sum_{k=1}^{\infty} \sigma_{t-\frac{k}{n}} \int_{t-\frac{k}{n}}^{t-\frac{k}{n}+\frac{1}{n}} g(t-s) dW_s$$

Reminding that

$$g(\tau) = \tau^{\alpha} L_g(\tau), \quad \tau > 0$$

where L_g is slowly varying and satisfy the hypotheses 3.13 - 3.17 then in the first terms of the sum, we can exploit the structure of g: since L_g is a slowly varying function, then one can approximate as a step function in a finite interval of type $\left[\frac{k-1}{n}, \frac{k}{n}\right]$, for some $n \in \mathbb{N} \setminus \{0\}, k \leq n$

$$g(t-s) \approx (t-s)^{\alpha} L_g\left(\frac{k}{n}\right), \quad t - s \in \left[\frac{k-1}{n}, \frac{k}{n}\right] \setminus \{0\}$$

while for larger values of k then one can just approximate g as a step function:

$$g(t-s) \approx g\left(\frac{b_k}{n}\right), \quad t-s \in \left[\frac{k-1}{n}, \frac{k}{n}\right]$$

where b_k can be optimally chosen, with the only constraint being that $b_k \in [k-1, k]$. Now, having chosen a κ small, in the range $\kappa \in \{1,2,3\}$ as they propose in their paper, one can approximate the process X_t by:

$$\begin{aligned} X_t \approx & \sum_{k=1}^{\kappa} L_g\left(\frac{k}{n}\right) \sigma_{t-\frac{k}{n}} \int_{t-\frac{k}{n}}^{t-\frac{k}{n}+\frac{1}{n}} (t-s)^{\alpha}\, dW_s + \\ & + \sum_{k=\kappa+1}^{\infty} g\left(\frac{b_k}{n}\right) \sigma_{t-\frac{k}{n}} \int_{t-\frac{k}{n}}^{t-\frac{k}{n}+\frac{1}{n}} dW_s \end{aligned}$$

The method is easily adapted to Truncated Brownian Semi-Stationary processes (TBSS), which are of most interest for our financial applications. So in this case the approximation, dropping all the terms of the sum for which the interval have non trivial intersection with the negative half real line then becomes:

$$\begin{aligned} Y_t \approx & \sum_{k=1}^{\min(\lfloor nt \rfloor, \kappa)} L_g\left(\frac{k}{n}\right) \sigma_{t-\frac{k}{n}} \int_{t-\frac{k}{n}}^{t-\frac{k}{n}+\frac{1}{n}} (t-s)^{\alpha}\, dW_s + \\ & + \sum_{k=\kappa+1}^{\lfloor nt \rfloor} g\left(\frac{b_k}{n}\right) \sigma_{t-\frac{k}{n}} \int_{t-\frac{k}{n}}^{t-\frac{k}{n}+\frac{1}{n}} dW_s \end{aligned}$$

Now, one needs to note that both integrals in the equation above, are in the end Gaussian random variables, with mean zero and for the first summand, we have a particular covariance structure for which the details are given in the original paper. While as for the second sum, the integrals are just increments of Brownian motion. So if we want to sample Y at a particular time, it can be computed as, [10, Pg.938]:

$$Y_{\frac{i}{n}}^n = \sum_{k=1}^{\min(i,\kappa)} L_g\left(\frac{k}{n}\right) \sigma_{\frac{i-k}{n}} W_{i-k,k}^n + \sum_{k=\kappa+1}^{i} g\left(\frac{b_k}{n}\right) \sigma_{\frac{i-k}{n}} W_{i-k}^n \tag{4.6}$$

Where

$$W_{i,j}^n := \int_{i/n}^{i/n+1/n} \left(\frac{i+j}{n} - s\right)^{\alpha} dW_s, \qquad \begin{aligned} & i = 0, \ldots, \lfloor nT \rfloor - 1 \\ & j = 1, \ldots, \kappa \end{aligned}$$

$$W_i^n := \int_{\frac{i}{n}}^{\frac{i+1}{n}} dW_s, \qquad i = 0, \ldots, \lfloor nT \rfloor - 1$$

Note that the the vectors

$$\mathbb{W}_i^n := \left(W_i^n, W_{i,1}^n, \ldots, W_{i,\kappa}^n\right), \quad i = 0, \ldots, \lfloor nT \rfloor - 1$$

Are independent realizations of a normal Gaussian random variable with mean zero and covariance matrix Σ which is known in closed form.
Another important remark, is the fact that the second sum of (4.6) can be seen as a discrete convolution, and so we can reduce the computational complexity to $O(n \log n)$. Note also that to be precise, one has also to compute the Cholesky decomposition of the covariance matrix Σ in order to generate correlated Gaussian random vectors. However, as they state in their paper, low values of κ such as $\kappa = 3$ is enough for Brownian Semi-Stationary processes, while it's even enough $\kappa \in \{1,2\}$ for Truncated Brownian Semi-Stationary processes. This means that in fact the Cholesky computational complexity does not influence the efficiency of the algorithm, and $n \log n$ remains the term with higher weight.

In its Turbocharged version, [37] the authors employ a variance reduction technique using as a control variate the value of a Timer Option, managing to speed up the simulation time up to a factor of 20. However, they do so conditioning first on the filtration generated by the Brownian motion driving the variance, and this has the caveat of loosing the relationship with Hedging strategies, as the authors them self remark.
As for the theoretical convergence of this scheme, we note that strong convergence is proven. In fact, theorem 2.5 of [10], gives the proof for at least the case that we are interested in. However, more than of the strong error, since our ultimate goal would be in the end option pricing, we need to take care of the weak error of such an approximation. In this sense, the paper of Paul Gassiat [27] and the paper [33] try to answer this question. In fact, in the latter they prove that there is indeed weak convergence of the hybrid scheme in the sense that they prove the approximation of fractional Brownian motion converges weakly to fractional Brownian motion, and also they guarantee that there is convergence of the approximation log-stock price and the hybrid scheme itself. However, proving weak convergence and establishing a precise rate of convergence is far from a trivial step. In this direction, Paul Gassiat proved that if the test function is a cubic polynomial, and the volatility has enough regularity,which in our practical case is guaranteed since they just require that

$$\sigma_t = f\left(t, W_t^H\right) \quad f \in \mathcal{C}_b^3$$

then the weak order of convergence is of $H + 1/2$.
Now that we have seen in this chapter different methods for simulating Rough Volatility models, we move on to the main topic of this work, which is the Markovian approximation of Rough Volatility models.

5

Markovian Approximation

In the previous section, we explored various simulation methods for rough volatility models. The literature on this topic is extensive, and we did not cover all available simulation schemes. We discussed two exact schemes and one approximation, namely the Hybrid scheme. While the Hybrid scheme performs well for various volatility models, it has the limitation of being applicable only to Brownian Semi-Stationary (BSS) processes and Truncated Brownian Semi-Stationary (TBSS) processes. Consequently, models like the Rough Heston model were not included in our discussion.

In a pioneering paper by Abi Jaber and El Euch [1], inspired by a well-known result from Carmona [15], a method was introduced to construct a Markovian approximation of rough volatility models. This method is both straightforward and flexible. In this section, we will present the construction of this approximation. Following that, we will discuss the weak and strong error bounds associated with it.

5.1. Construction of the approximation

Let us start with a definition, that ensures us the existence of solution, and it is flexible enough to contain every model that we have discussed in section 2 but the rough Heston Model, for which the existence of a weak, non-negative solution to the volatility process is also ensured, but via a different type of argument.
Let us give a couple of definitions that will be useful for the rest of the work.

Definition 5.1.1 (Complete Monotone Kernel)**.** A function $f : \mathbb{R}^+ \to \mathbb{R}^+$ is called completely monotone if there exists a measure μ on $\mathbb{R}^+$ such that:

$$\forall \tau \in \mathbb{R}^+ : \quad f(\tau) = \int_0^\infty e^{-\tau x} \mu(dx) \tag{5.1}$$

Now, we can fully specify the type of equation that we will work with for the rest of this work:

Definition 5.1.2 (Volterra Equation)**.** On a given filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\in[0,T]}, \mathbb{P})$, for some $T > 0$, we define to be a Volterra Equation an SDE of the form:

$$X_t = x_0 + \int_0^t g(t-s) b(X_s) ds + \int_0^t g(t-s) \sigma(X_s) dW_s, \tag{5.2}$$

where W is standard d-dimensional Brownian Motion, $x_0 \in \mathbb{R}^d$, $b : \mathbb{R}^d \to \mathbb{R}^d$, $\sigma : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ are globally Lipschitz continuous, i.e.:

$$\exists L > 0 : \ \forall x, y \in \mathbb{R}^d, |b(x) - b(y)| + ||\sigma(x) - \sigma(y)|| < L|x - y|$$

Also, we ask $g : \mathbb{R}^+ \to \mathbb{R}$ to be a completely monotone kernel such that there exist a bounded measurable function $M : \mathbb{R}^+ \to \mathbb{R}$ such that

$$g(\tau) = \int_{\mathbb{R}^+} e^{-\rho t} M(\rho) \mu(d\rho)$$

and:

$$\int_{\mathbb{R}^+} e^{-\rho t}\mu(d\rho) < \infty$$

Also, we assume that

$$\int_0^T |g(s)|^2\, ds < \infty, \quad \forall T \in \mathbb{R}_+$$

Under those assumptions one can apply Theorem 3.1 of [43], that states that in such case, there exists a unique strong solution to the equation (5.2), and also one has that:

$$\forall \tau \in [0,T], \quad \mathbb{E}\left[|X_\tau|^2\right] \leq C_{T,2,g,x_0}\left(1 + \int_0^T g\,(T-s)\, ds\right)$$

that implies:

$$\sup_{\tau\in[0,T]} \mathbb{E}\left[|X_\tau|^2\right] < \infty \tag{5.3}$$

Note that for our work, in Rough Volatility models, we have that the Volatility is driven by a Stochastic Volterra Equation. Now let us present proposition 2.1 of Alfonsi and Kebaier [35] that allows a representation of the solution to such Stochastic Volterra Equation in a particular, but useful setting.

Proposition 2. Let us assume that $\mu(d\rho) = \sum_{i=1}^n \alpha_i \delta_{\rho_i}(d\rho)$, with $\alpha_i \geq 0$ and $\rho_1 < \cdots < \rho_n$. In the setting of the definition of Stochastic Volterra Equations in (5.2) say that there exist $x_0^1, \ldots, x_0^n \in \mathbb{R}^d$ such that $\sum_{i=1}^n \alpha_i M(\rho_i) x_0^i = x_0$. Then, the solution of (5.2) is given by: $\sum_{i=1}^n \alpha_i M(\rho_i) X_t^i$ where $(X_t^1, \ldots, X_t^n)$ is the solution of the $n \times d-$dimensional stochastic differential equation defined by:

$$X_t^i = x_0^i - \int_0^t \rho_i (X_t^i - x_0^i) ds + \int_0^t b(\sum_{i=1}^n \alpha_i M(\rho_i) X_s^i) ds + \int_0^t \sigma(\sum_{i=1}^n \alpha_i M(\rho_i) X_s^i) dW_s \tag{5.4}$$

Proof. The SDE (5.2) has Lipschitz coefficients, and therefore there exists a unique strong solution. Now compute:

$$\begin{aligned}
d(e^{\rho_i t}\left(X_t^i - x_0^i\right)) &= \rho_i e^{\rho_i t}\left(X_t^i - x_0^i\right) dt + e^{\rho_i t} d\left(X_t^i - x_0^i\right) \\
&= \rho_i e^{\rho_i t}\left(X_t^i - x_0^i\right) dt - \rho_i e^{\rho_i t}\left(X_t^i - x_0^i\right) dt + e^{\rho_i t} b(\sum_{i=1}^n \alpha_i M(\rho_i) X_t^i) dt \\
&+ e^{\rho_i t}\sigma(\sum_{i=1}^n \alpha_i M(\rho_i) X_t^i) dW_t \\
&= e^{\rho_i t} b(\sum_{i=1}^n \alpha_i M(\rho_i) X_t^i) dt + e^{\rho_i t}\sigma(\sum_{i=1}^n \alpha_i M(\rho_i) X^i) dW_t
\end{aligned}$$

i.e., we get:

$$X_t^i = x_0^i + \int_0^t e^{-\rho_i(t-s)} b(\sum_{i=1}^n \alpha_i M(\rho_i) X_s^i) ds + \int_0^t e^{-\rho_i(t-s)}\sigma(\sum_{i=1}^n \alpha_i M(\rho_i) X_s^i) dW_s$$

Now, if we left multiply this equation by $\alpha_i M(\rho_i)$ and sum over i then $\sum_{i=1}^n \alpha_i M(\rho_i) X_t^i$ solves (5.2), and the strong uniqueness result gives the claim. In fact, (5.2) with $\lambda(d\rho) = \sum_{i=1}^n \alpha_i \delta_{\rho_i}(d\rho)$ becomes:

$$\begin{aligned}
X_t &= x_0 + \int_0^t \int_{\mathbb{R}_+} e^{-\rho(t-s)} M(\rho)\lambda(d\rho) b(X_s) ds + \int_0^t \int_{\mathbb{R}_+} e^{-\rho(t-s)} M(\rho)\lambda(d\rho)\sigma(X_s) dW_s = \\
&= \int_0^t \sum_{i=1}^n e^{-\rho_i(t-s)}\alpha_i M(\rho_i) b(X_s) ds + \int_0^t \sum_{i=1}^n e^{-\rho_i(t-s)}\alpha_i M(\rho_i)\sigma(X_s) dW_s
\end{aligned}$$

and if one considers $X_t = \sum_{i=1}^n \alpha_i M(\rho_i) X_t^i$ then it's clear that the two solutions coincides. □

If we specialise now this proposition to our case, i.e.: the fractional kernel case with Hurst parameter H then one has that M as in (5.7) is exactly $M \equiv 1$, and

$$\mu(x) = \frac{1}{\Gamma(H+1/2)\Gamma(1/2-H)} x^{-(H+1/2)},$$

in fact:

$$\frac{\tau^{H-1/2}}{\Gamma(H+1/2)} = \frac{1}{\Gamma(H+1/2)\Gamma(1/2-H)} \int_0^\infty e^{-\tau x} x^{-(H+1/2)} dx,$$

approximating g as a weighted sum of say, K exponentials, i.e.:

$$\hat{g}(\tau) = \sum_{i=1}^{K} w_i e^{-\tau x_i},$$

for some nodes $x_i \in \mathbb{R}^+, w_i \in \mathbb{R}, \forall i = 1, \dots, K$, then the approximation of Stochastic Volterra equation (5.2) becomes:

$$\hat{X}_t = x_0 + \int_0^t \hat{g}(t-s) b(\hat{X}_s) ds + \int_0^t \hat{g}(t-s) \sigma(\hat{X}_s) dW_s, \tag{5.5}$$

then we can rewrite Proposition 2 as Proposition 1.1 in [7]:

Proposition 3. Let $x_0^1, \dots, x_0^K \in \mathbb{R}^d$ such that $\sum_{i=1}^K w_i x_0^i = x_0$, Then the solution to (5.5) is given by $\sum_{i=1}^K w_i X_t^i$ where $\left(X_t^1, \dots, X_t^K\right)$ is the solution to the $N \times d$- dimensional system of SDEs defined by:

$$X_t^i = x_0^i - \int_0^t x_i \left(X_t^i - x_0^i\right) ds + \int_0^t b\left(\sum_{j=1}^K w_j X_s^j\right) ds + \int_0^t \sigma\left(\sum_{j=1}^K w_j X_s^j\right) dW_s \tag{5.6}$$

Note that this is an approximation of the original process X_t, and it is given by a sum of OU-processes driven by the same Brownian motion, but with different mean reverting speed. In fact, from a qualitative point of view, having a path with low Hölder regularity can be stylised with superimposing OU-processes with high mean reversion. The nodes of the quadrature appear in the SDE 5.6 only in the first integral, and they can be interpreted as the mean reversion speeds of the different Orhnstein-Uhlenbeck processes.

Now that we know how to simulate the Markovian Approximation given the nodes and weights $(x_i, w_i)_{i=1}^N$, let's explore in the next section what is the objective that one has to have in mind when choosing the nodes and the weights for the quadrature rule.

5.2. Strong And Weak Error Bounds

In this section, we are going to present both the strong and the weak error bounds for the Markovian Approximation of Rough Volatility models.

Our discussion will be divided in two parts, the first part will regard the strong error that one has when approximating Rough Volatility models, and we will see how this type of error can be bounded with the $\mathcal{L}^2([0,T])$ error between the original kernel and its approximation. Then in the second part of this section we are going to focus on the weak error for the Rough Heston Model.

In particular, we will need to distinguish in our discussion, the case of the Rough Heston model and all the other models, this is because, σ, as in (5.2) in the case of the Rough Heston model is equal to $\sigma(x) = \sqrt{x}$, that is a non-Lipschitz function over the positive real half line. We will see how for this model, the weak error is bounded by the $\mathcal{L}^1([0,T])$ error between the kernel and its approximation.

5.2.1. Strong Error Bounds

Let us take care now of the strong error that one has when using the Markovian Approximation. To do this, let us reproduce the proposition and its proof for the non-Rough Heston case, which corresponds to [35, Proposition 3.2]. They relate the strong $\mathcal{L}^2(0,T)$ between the approximated solution of (5.5)

and the solution of (5.2) to the $\mathcal{L}^2(0,T)$ error between the kernel g and its approximation $\hat{g}$. We will present a particular version of this proposition, perhaps in a bit less general setting, that will be sufficient for our purposes. In any case, the proposition is easily generalised to the multidimensional case, just replacing absolute values with euclidean norms, and for matrices with the Frobenious norm. To do this, we assume that g has the same properties as in Definition 5.1.2 id est.:

$$g(\tau) = \int_{\mathbb{R}_+} e^{-\rho\tau} M(\rho)\lambda(d\rho), \quad t \in (0,\infty), \tag{5.7}$$

with $M : \mathbb{R}_+ \to \mathbb{R}$, with M bounded i.e.:

$$\mathbf{M} := \sup_{\rho\in\mathbb{R}_+} ||M(\rho)|| < \infty. \tag{5.8}$$

For our case, let's assume that our approximating kernel $\hat{g}$ is such that:

$$\hat{g} \in \mathcal{L}^2_{loc}(\mathbb{R}_+, \mathbb{R}_+), \tag{5.9}$$

which for our case certainly holds since:

$$\hat{g}(\tau) = \sum_{i=1}^{N} w_i e^{-x_i t}.$$

Then, for $c > 0$, consider the resolvent of second kind $E_c(\tau)$ that solves the equation:

$$E_c(\tau) = \hat{g}^2(\tau) + \int_0^\tau c\hat{g}^2(\tau - s) E_c(s) ds. \tag{5.10}$$

Thanks to assumption (5.9), we know also that $\hat{g}^2 \in \mathcal{L}^1_{loc}(\mathbb{R}^+, \mathbb{R}^+)$ then, we have that $E_c(\tau)$ is well defined and also that $E_c(\tau) \in \mathcal{L}^1_{loc}(\mathbb{R}^+, \mathbb{R}^+)$. Now that we have all the tools, we can reproduce the proof of [35, Proposition 3.2]. In particular, the goal is to arrive at a Grönwall-type of inequality, and then apply the result of [30, Theorem 9.8.2]:

Proposition 4. [30, Theorem 9.8.2] Let k be a scalar kernel, i.e. a measurable function:

$$k : [0,T] \to \mathbb{R}$$

Assume also that $k \in \mathcal{L}^p_{loc}([0,T])$ Moreover, assume that $-k$ has a non-positive resolvent r of the same type. Let $x, f \in \mathcal{L}^p_{loc}(0,T)$ such that

$$x(t) \leq (k \star x)(t) + f(t) \ \ a.s.$$

where $\star$ is the convolution operator. Then, $x(t) \leq y(t)$, where y solves:

$$y(t) = (k \star y)(t) + f(t).$$

With this in mind, let's move on to [35, Proposition 3.2]

Proposition 5. [35, Proposition 3.2] Let $(\hat{X}_t)_{t\in[0,T]}$ be the strong solution of (5.5) and $(X_t)_{t\in[0,T]}$ the strong solution to (5.2). Then there exists a constant C, dependent on the kernel,T, b, σ such that:

$$\forall t \in [0,T], \quad \mathbb{E}\left[\left|\hat{X}_t - X_t\right|^2\right] \leq C \int_0^T |g(\tau) - \hat{g}(\tau)|^2 d\tau$$

Proof. Denote, in a similar fashion of [35, Proposition 3.2]:

$$\Delta(\tau) := g(\tau) - \hat{g}(\tau), \quad \forall \tau \in [0,T].$$

Then, $\forall t \in [0,T]$ one has:

$$\begin{aligned}\left|\hat{X}_t - X_t\right|^2 \leq\ & 4\left|\int_0^t \Delta(t-s) b(X_s) ds\right|^2 + 4\left|\int_0^t \Delta(t-s)\sigma(X_s) dW_s\right|^2 \\ & + 4\left|\int_0^t \hat{g}(t-s)\left[b(X_s) - b(\hat{X}_s)\right] ds\right|^2 + 4\left|\int_0^t \hat{g}(t-s)\left[\sigma(X_s) - \sigma(\hat{X}_s)\right] dW_s\right|^2\end{aligned}$$

where we used the inequality: $(a+b+c+d)^2 \leq 4\left(a^2+b^2+c^2+d^2\right)$. Now taking the expected value of the equation above, apply Ito's Isometry to the terms with σ while for the other two terms we use Jensen saying that:

$$\left|\int_0^t \Delta(t-s)b(X_s)ds\right|^2 \leq t\int_0^t \left|\Delta(t-s)b(X_s)\right|^2 ds,$$

then one has that:

$$\mathbb{E}\left[\left|\int_0^t \Delta\left(t-s\right)b\left(X_s\right)ds\right|^2\right] \leq \mathbb{E}\left[t\int_0^t \left|\Delta\left(t-s\right)b\left(X_s\right)\right|^2 ds\right] = t\int_0^t \left|\Delta\left(t-s\right)\right|^2 \mathbb{E}\left[\left|b\left(X_s\right)\right|^2\right] ds$$

As for the other term, repeating the same reasoning:

$$\begin{aligned}\mathbb{E}\left[\left|\int_0^t \hat{g}\left(t-s\right)\left[b\left(X_s\right)-b(\hat{X}_s)\right]ds\right|^2\right] &\leq \mathbb{E}\left[t\int_0^t \left|\hat{g}\left(t-s\right)\right|^2\left|b\left(X_s\right)-b(\hat{X}_s)\right|^2 ds\right] \\ &= t\int_0^t \left|\hat{g}\left(t-s\right)\right|^2 \mathbb{E}\left[\left|b\left(X_s\right)-b(\hat{X}_s)\right|^2\right] ds\end{aligned}$$

Putting all the pieces together we obtain:

$$\begin{aligned}\mathbb{E}\left[\left|\hat{X}_t - X_t\right|^2\right] \leq\ & 4t\int_0^t \left|\Delta\left(t-s\right)\right|^2 \mathbb{E}\left[\left|b\left(X_s\right)\right|^2\right] ds + 4\int_0^t \left|\Delta\left(t-s\right)\right|^2 \mathbb{E}\left[\left|\sigma\left(X_s\right)\right|^2\right] ds \\ &+ 4t\int_0^t \left|\hat{g}\left(t-s\right)\right|^2 \mathbb{E}\left[\left|b\left(X_s\right)-b(\hat{X}_s)\right|^2\right] ds + 4\int_0^t \left|\hat{g}\left(t-s\right)\right|^2 \mathbb{E}\left[\left|\sigma\left(X_s\right)-\sigma(\hat{X}_s)\right|^2\right] ds.\end{aligned} \tag{5.11}$$

Now, knowing that the coefficients σ and b are Lipschitz continuous one has that :

$$\forall x,y\in\mathbb{R},\ \ b(x) \leq |b(0)| + L\left|x\right|, \quad b(x-y) \leq L\left|x-y\right|, \tag{5.12}$$

$$\forall x,y\in\mathbb{R},\ \ \sigma(x) \leq |\sigma(0)| + L\left|x\right|, \quad \sigma(x-y) \leq L\left|x-y\right| \tag{5.13}$$

So, using (5.12) and (5.13) we have that (5.11) becomes:

$$\begin{aligned}\mathbb{E}\left[\left|\hat{X}_t - X_t\right|^2\right] &\leq 4t\int_0^t \left|\Delta\left(t-s\right)\right|^2 \mathbb{E}\left[\left(|b(0)| + L\left|X_s\right|\right)^2\right] ds \\ &\quad + 4\int_0^t \left|\Delta\left(t-s\right)\right|^2 \mathbb{E}\left[\left(|\sigma(0)| + L\left|X_s\right|\right)^2\right] ds \\ &\quad + 8L^2\int_0^t \left|\hat{g}\left(t-s\right)\right|^2 \mathbb{E}\left[\left|X_s-\hat{X}_s\right|^2\right] ds \\ &\quad \leq 8\left(T\vee 1\right)\left(|b(0)|^2 \vee |\sigma(0)|^2 + 2L^2 \sup_{\tau\in[0,T]} \mathbb{E}\left[\left|X_\tau\right|^2\right]\right)\int_0^t \left|\Delta\left(t-s\right)\right|^2 ds \\ &\quad + 8L^2\int_0^t \left|\hat{g}\left(t-s\right)\right|^2 \mathbb{E}\left[\left|X_s-\hat{X}_s\right|^2\right] ds\end{aligned} \tag{5.14}$$

Note that, we used (5.3) to bound:

$$\mathbb{E}\left[\left|X_s\right|^2\right] \leq \sup_{\tau\in[0,T]} \mathbb{E}\left[\left|X_\tau\right|^2\right] < \infty, \quad \forall s\in[0,T]$$

now that we have fully justified how to arrive at (5.14) then, the idea here in [35], is to use Grönwall-type of inequality of Lemma 4, to obtain:

$$\mathbb{E}\left[\left|\hat{X}_t - X_t\right|^2\right] \leq c_1\left(\int_0^t \left|\Delta\left(t-s\right)\right|^2 ds\right)\left(1+\int_0^T E_{c_2}(s)ds\right)$$

where E_c is defined as in (5.10) where the constant are defined:

$$c_1 = (T \vee 1)\left(|b(0)|^2 \vee |\sigma(0)|^2 + 2L^2 \sup_{\tau\in[0,T]} \mathbb{E}\left[|X_\tau|^2\right]\right)$$
$$c_2 = 8L^2$$

And $E_c(x)$ is defined as in (5.10) and also $E_c\left(\mathcal{L}^1_{loc}(\mathbb{R}^+,\mathbb{R}^+)\right)$. Then, one can define the constant C, to be:

$$C := c_1\left(1 + \int_0^T E_{c_2}(s)ds\right)$$

And then arrive at the thesis of the proposition:

$$\mathbb{E}\left[\left|\hat{X}_t - X_t\right|^2\right] \leq C\int_0^t |\Delta(t-s)|^2 ds$$

□

With this proposition then, one can reduce the initially very complex problem of determining how good it is the approximation of the solution to the Volterra Equation (5.2) with a much simpler problem, that is to minimise the $\mathcal{L}^2([0,T])$ error between the Kernel g and its approximation $\hat{g}$. Now, another thing that one has to take into account is the fact that the Markovian Approximation $\hat{X}_t, t \in [0,T]$ cannot always be simulated exactly. So one has also take into account the discretisation error when simulating Rough Volatility models. On the other side, models like the Rough Heston model, suffer less from this fact, since its characteristic function can be computed with arbitrary precision, and then one can both simulate exactly the model, and also can use Fourier Pricing methods.

5.2.2. Weak Error under Rough Heston

Now that we have discussed and presented the results in [35] on how one can relate the strong error of the Markovian approximation and the $\mathcal{L}^2([0,T])$ error between g and $\hat{g}$, we can go on in our discussion and see how things work for the Rough Heston model. In this section we are going to present the results on how one can bound the strong error and the weak error with the $\mathcal{L}^1([0,T])$ error between g and $\hat{g}$. As we have seen in section 3.2, the Rough Heston model reads:

$$dS_t = S_t\sqrt{V_t}dW_t \tag{5.15}$$
$$V_t = V_0 + \int_0^t g(t-s)(\theta(s) - \lambda V_s)ds + \int_0^t g(t-s)\nu\sqrt{V_t}dB_s \tag{5.16}$$

Where W, B are correlated Brownian motion, with correlation coefficient ρ, λ, ν are positive parameters, while $\theta(s)$ is a deterministic function that it's used to fit the forward variance curve at time zero. However, we can see how in the second term of (5.16) appears a square root, which is not Lipschitz continuous, so the volatility in the Rough Heston model does not meet the definition of Volterra equation (5.2). For this reason, the discussion for this model is a bit different, and it is mostly based on the fact that Abi Jaber and El Euch in [1]. They showed that the characteristic function for the Rough Heston model can be obtained via solving a fractional Riccati equation, i.e.:

$$\mathbb{E}\left[\exp\left(z\log\left(S_t/S_0\right)\right)\right] = \exp\left(\int_0^t F\left(z,\psi\left(t-s,z\right)\right)G(s)\,ds\right)$$

where:

$$G(t) = V_0 + \int_0^t g(t-s)\theta(s)\,ds$$

and ψ is the solution of a fractional Riccati-type equation:

$$\psi(t,z) = \int_0^t g(t-s)F\left(z,\psi(s,z)\right)ds$$

where

$$F(z,x) = \frac{1}{2}\left(z^2 - z\right) + (\rho\nu z - \lambda)\, x + \frac{\nu^2}{2} x$$

In the same fashion as before, Abi Jaber and El Euch showed that by replacing the kernel g with the kernel $\hat{g}$, then one can also find the characteristic function for the log-stock price via solving an ordinary Riccati equation. So denoting $\left(\hat{S}, \hat{V}\right)$ the Markovian approximation then one has:

$$\mathbb{E}\left[\exp\left(z \log\left(\hat{S}_t / S_0\right)\right)\right] = \exp\left(\int_0^t F\left(z, \hat{\psi}\left(t-s, z\right)\right) \hat{G}(s)\, ds\right)$$

where:

$$\hat{G}(t) = V_0 + \int_0^t \hat{g}(t-s)\, \theta(s)\, ds$$

and $\hat{\psi}$ is the solution of a Riccati-type equation:

$$\hat{\psi}(t,z) = \int_0^t \hat{g}(t-s) F\left(z, \hat{\psi}(s,z)\right) ds$$

It is shown, always in [1], that this equation can be solved by usual numerical integrators for ODEs. In particular, they also gave a bound for the absolute error between the characteristc function ψ and the one of the Markovian approximation $\hat{\psi}$. In particular they gave the following theorem:

Proposition 6. [1, Theorem 4.1] There exists a constant $C > 0$, such that $\forall a \in [0,1], b \in \mathbb{R}$ one has:

$$\sup_{t \in [0,T]} \left|\hat{\psi}(t, a+ib) - \psi(t, a+ib)\right| \leq C\left(1 + b^4\right) \int_0^T |\hat{g}(s) - g(s)|\, ds$$

using this result, they proved the following proposition:

Proposition 7. [1, Proposition 4.3] Denote $C(k,T)$ the price of a call option with log-strike k and expiration T under the Rough Heston model, and denote $\hat{C}(k,T)$ the price of the same derivative, under the Markovian approximation of the Rough Heston model. Then there exists a constant c such that:

$$\left|C(k,T) - \hat{C}(k,T)\right| \leq c \int_0^T |\hat{g}(s) - g(s)|\, ds$$

Finally, Bayer and Breneis, in a recent paper, managed to improve proposition (7) for a much more general payoff function, in fact Corollary 2.12 of [7] states:

Proposition 8. [8, Corollary 2.12] Let $h : \mathbb{R}^+ \to \mathbb{R}$ a 8 times weakly differentiable and compactly supported function, Then there exists a constant C, such that:

$$\left|\mathbb{E}\left[h(S_T) - h\left(\hat{S}_T\right)\right]\right| \leq c \int_0^T |g(s) - \hat{g}(s)|\, ds$$

The previous two results are per se very much interesting: as a first observation, we see that the weak error, is bounded now by the $\mathcal{L}^1([0,T])$ error instead of the $\mathcal{L}^2([0,T])$. This is way more convenient because especially for Hurst parameters $H \approx 0$, the kernel g becomes barely $\mathcal{L}^2([0,T])$, in the sense that the norm tends to explode as $H \to 0$, and this is not true for the $\mathcal{L}^1([0,T])$ norm. Also, as Bayer et Al. in [6] points out, solving a (System) of Riccati equation can be done in a more efficient way. In particular the fractional Riccati equation can be solved with Adam scheme, at a rate of convergence equal to $\mathcal{O}(\Delta t)$, while a predictor-corrector scheme is able to solve a Ordinary Riccati equation with $\mathcal{O}\left(\Delta t^2\right)$, where Δt is the stepsize.
Another advantage of using the Markovian approximation for the Rough Heston model is the following: if one were to use the approximation in a model like Rough Bergomi, where the characteristic function is

not available neither in closed form nor as a solution of some differential equation, then the only method for pricing that is available becomes the classical Monte Carlo simulation. Then, one has also to take into account the discretization error of the Euler Scheme, Alfonsi and Kebaier in [35] showed that it is of the order $\left(\frac{T}{N}\right)^{2H}$, where here N stands for the number of time steps used in the discretization of the SDE. In fact, [35, corollary 4.1] states:

Proposition 9. [35, corollary 4.1] In the setting of definition 5.1.2, let X_t the solution to (5.2), and Let $\hat{X}_t^N$ the discretised solution of (5.5), where N time steps have been used for the discretization. Also, let $g, \hat{g}$ the original kernel and its approximation respectively. Then we have a bound on the strong error of the discretization error for a Rough Kernel with $H \in (0, 1/2)$: for some $C > 0$ depending on b, σ, g such that:

$$\max_{k\in\{1,\dots,N\}} \mathbb{E}\left[\left|\hat{X}_{t_k}^N - X_{t_k}\right|^2\right] \leq C\left(\left(\frac{T}{N}\right)^{2H} + \frac{T}{N}\sum_{k=1}^{N}\|g(t_k) - \hat{g}(t_k)\|^2\right) \tag{5.17}$$

Note that this proposition tells us two things: The first term in (5.17) represents the discrtization error between the Markovian approximation, and its discretized counterparty. This error, as we can see has a power law decay in the number of discretization error, with exponent $2H$, that for financial applications remains very low. This means that if we are using Euler Scheme to simulate Stochastic Volterra Equations with fractional kernel, then one does not need a particularly good approximation for the kernel since the main source of the error will be in any case the discretization error. Then, the second term in (5.17), represents the error between the Markovian approximation and the real solution to (5.2). Note here that the $L^2([0,T])$ error has been replaced to its discretized version, which for fractional kernels tends to be much less than the continuous counterparty. The bottom line after those observations is that one can exploit the most a good Markovian approximation in two cases: when there is a possibility of simulating exactly the approximation, or when the characteristic function is known, and hence one can use Fourier Methods both for simulation and pricing.

In the next subsection we are going to present different quadrature rules in the literature. Both quadrature rules in Abi Jaber and El Euch [1] or Alfonsi, Kebayer [35] have a rate of convergence that behaves like a power law. In particular, in the paper of Alfonsi and Kebayer, they manage to find a quadrature rule that converge with rate N^{-H}. However, this rate of convergence is still very slow, since it was shown that usually the values of H tend to be close to zero. Then, Bayer and Breneis, in [6], they provide a set of nodes and weights that converges at an exponential rate. Let us start with the latter. In particular, Gaussian quadrature rule will be used in different papers, so let us give first a first insight of how Gaussian quadrature rule works.

5.3. Nodes and Weights

In this section, we are going to present different quadrature rules that we have found in the literature. We are going to present the main ideas behind Gaussian quadrature rule, since it has been used widely in the literature to approach this problem, see e.g. [9],[6],[7],[8]. Then we are going to see how the Gaussian quadrature was used in the literature. Finally, we are going to present quadrature rules that rely not on a deterministic and predefined rule like the Gaussian, but rely on optimization algorithms that work directly with the $\mathcal{L}^p([0,T])$ errors, for $p \in \{1, 2\}$.

At the end of this section one will have an overview of all the quadrature rules that are present in the literature for this specific problem.

5.3.1. Gaussian Quadrature Rule

In this section, we are going to present some basics but still important facts about Gaussian quadrature.

Let $[a, b]$ a finite non empty interval, and let $\omega : [a, b] \to \mathbb{R}^+$ a positive continuous weight function. Let instead $f : [a, b] \to \mathbb{R}$ the function that we are trying to integrate. Consider the following approximation:

$$\int_a^b f(x)dx \approx \sum_{i=1}^{m} w_i f(x_i) \tag{5.18}$$

Now, we have to choose m nodes and m weights, for a total of $2m$ degrees of freedom. Gaussian quadrature of level m is defined to be the quadrature that integrates exactly all polynomials up to level

$2m-1$ against the weight w. This means that for every polynomial up to degree $2m-1$, (5.18) is an equality.

Now, in order to get the nodes and weights for the Gaussian rule, we need to compute the first m orthogonal polynomials p_n with respenct to the inner product:

$$\langle f, g\rangle_\omega = \int_a^b f(x)g(x)\omega(x)\,dx$$

Such polynomials can be found via a recurrence relation:

$$p_{k+1}(x) = \left(x - \frac{\langle xp_k, p_k\rangle_\omega}{\langle p_k, p_k\rangle_\omega}\right)p_k(x) - \sum_{i=0}^{k-1}\frac{\langle xp_i, p_i\rangle_\omega}{\langle p_i, p_i\rangle_\omega}p_j(x)$$
$$p_0(x) = 1$$

Then, the nodes for the Gaussian quadrature of level m are the roots of p_m. It can be shown that all these roots are real and lie in the interval $[a,b]$. Finally, the weights can be computed as:

$$w_i = \frac{\langle p_{m-1}, p_{m-1}\rangle_\omega}{p'_m(x_i)p_{m-1}(x_i)}$$

In particular, we can find an almost explicit expression for the error given by the Gaussian quadrature rule. In fact [13, Theorem 4.2.3]

Proposition 10. Let $f:[a,b]\to\mathbb{R}$ be a $2m$ times continuously differentiable function and let w_i, x_i be the nodes and weights of the Gaussian quadrature rule for $i=1,\dots,m$, then:

$$\int_a^b f(x)\omega(x)dx - \sum_{i=1}^m w_i f(x_i) = \int_a^b f^{(2m)}(x)\kappa_{2m}(x)dx$$

where κ_{2m} is the Peano Kernel corresponding to the weight function ω.

5.3.2. Superpolynomial Rate

In this section, we are going to deep-dive into the paper of of Bayer and Breneis [7], where they give a quadrature rule that achieves a superpolynomial rate of convergence. In particular, their idea is the following: given the Laplace transform of the power-law Kernel

$$G(t) = \frac{t^{H-1/2}}{\Gamma(H+1/2)} = c_H\int_0^\infty e^{-\rho t}\rho^{-H-1/2}d\rho, \quad c_H = \frac{1}{\Gamma(H+1/2)\Gamma(1/2-H)}$$

they divide the positive real line into intervals, say $[\xi_i, \xi_{i+1}]$ for $i=0,\dots,n$, with $\xi_i > 0$. Then they apply Gaussian quadrature of level m to those intervals with the weight function being

$$\omega(x) := c_H x^{-H-1/2} \tag{5.19}$$

Now, let us specify the parameters for the quadrature that are used and set this as a definition.

Definition 5.3.1. Let $N\in\mathbb{N}$ the total number of nodes and $\alpha,\beta,a,b\in(0,\infty)$. Define $A := A_H = \left(\frac{1}{H}+\frac{1}{3/2-H}\right)^{1/2}$, where $H\in(0,1/2)$ is the Hurst parameter. Define:

$$m := \frac{\beta}{A}\sqrt{N}, \quad n := \frac{A}{\beta}\sqrt{N}$$
$$\xi_0 := a\exp\left(-\frac{\alpha}{(3/2-H)A}\sqrt{N}\right), \quad \xi_n := b\exp\left(-\frac{\alpha}{HA}\sqrt{N}\right)$$
$$\xi_i := \xi_0\left(\frac{\xi_n}{\xi_0}\right)^{i/n}, \quad ,i=0,\dots,n$$

Then, define Gaussian rule of type (H,N,α,β,a,b) to be the sets of nodes and weights $(x_i)_{i=1}^{nm}$,$(w_i)_{i=1}^{nm}$ of the Gaussian quadrature rule of level m applied to the intervals $[\xi_i,\xi_{i+1}]$ for $i=0,\dots,n-1$ with the weight function ω defined as before in (5.19). In addition, they add a node $x_0 := 0$ with weight $w_0 = c_H\int_0^{\xi_0} x^{-H-1/2}dx = \frac{c_H}{1/2-H}\xi_0^{1/2-H}$.

With this in mind, they first compute the error that is made using Gaussian quadrature on a generic interval $[a,b]$ with respect to the weight function $\omega(x) = c_H x^{-H-1/2}$. In particular [7, Lemma 2.8] reads:

Lemma 1. [7, Lemma 2.8] Let w_i, x_i the nodes and weights of the Gaussian quadrature rule of level m, on the interval $[a,b]$ with respect to the weight function $\omega(x) = c_H x^{-H-1/2}$. Then:

$$\left| c_H \int_a^b e^{-tx} x^{-H-1/2} dx - \sum_{i=1}^{m} w_i e^{-x_i t} \right| \leq \sqrt{\frac{5\pi^3}{18}} \frac{c_H}{2^{2m+1} m^H} t^{-1/2+H} \left(\frac{b}{a} - 1 \right)^{2m+1}$$

the proof of this lemma relies on the estimate of the sup norm of the Peano kernel. Then, with those estimates available for each sub-interval, they then use it for giving an estimate for the error on the interval $[\xi_0, \xi_n]$.

Lemma 2. [7, Lemma 2.9] With the Gaussian rule of type $(H, N, \alpha, \beta, a, b)$ then one has:

$$\begin{aligned} &\int_0^T \left| c_H \int_{\xi_0}^{\xi_n} e^{-tx} x^{-H-1/2} dx - \sum_{i=1}^{N} w_i e^{-tx_i} \right|^2 dt \\ &\leq \frac{5\pi^3}{36} \frac{c_H^2 T^{2H}}{H} \frac{n^2}{m^{2H}} \left(\frac{1}{2} \left(e^{\alpha\beta} - 1 \right) \right)^{4m+2}. \end{aligned}$$

Proof. recalling that

$$\frac{\xi_{i+1}}{\xi_i} = \left(\frac{\xi_n}{\xi_0} \right)^{1/n} = \exp \frac{\alpha\sqrt{N}}{An} A^2 = \exp \frac{\alpha A \sqrt{N}}{n} = e^{\alpha\beta}$$

Then, using triangular inequality and lemma 1:

$$\begin{aligned} &\int_0^T \left| c_H \int_{\xi_0}^{\xi_n} e^{-tx} x^{-H-1/2} dx - \sum_{i=1}^{N} w_i e^{-tx_i} \right|^2 dt \\ &\leq \int_0^T \left| \sum_{i=0}^{n-1} \sqrt{\frac{5\pi^3}{18}} \frac{c_H}{2^{2m+1} m^H} t^{-1/2+H} \left(\frac{\xi_{i+1}}{\xi_i} - 1 \right)^{2m+1} \right|^2 dt \\ &= \int_0^T \left| n \sqrt{\frac{5\pi^3}{18}} \frac{c_H}{2^{2m+1} m^H} t^{-1/2+H} \left(e^{\alpha\beta} - 1 \right)^{2m+1} \right|^2 dt \\ &= \frac{5\pi^3}{36} \frac{c_H^2 T^{2H}}{H} \frac{n^2}{m^{2H}} \left(\frac{1}{2} \left(e^{\alpha\beta} - 1 \right) \right)^{4m+2} \end{aligned}$$

□

Now, they give an estimate of the $\mathcal{L}^2([0,T])$ error:

Theorem 2. [7, Theorem 2.1] Let $(X_t)_{t\in[0,T]}$ be the strong solution to (5.2), and $(\hat{X}_t)_{t\in[0,T]}$ the strong solution of (5.5), using the Gaussian rule in Definition 5.3.1, with parameters $(H, N, \alpha, \beta, 1, 1)$, with $\alpha := 1.06418$ and $\beta := 0.4275$, then:

$$\mathbb{E}\left[\left| X_t - \hat{X}_t \right|^2 \right] \leq C_{H,T,\alpha,\beta} \exp\left(-2\alpha \left(\frac{1}{H} + \frac{1}{3/2 - H} \right)^{-1/2} \sqrt{N} \right)$$

Proof. First, note that we have presented with proposition 5 already the fact that one can bound the strong error between the solution X and $\hat{X}$ with the $\mathcal{L}^2([0,T])$ error between the kernel g and its ap-

proximation $\hat{g}$. Then let us compute the latter explicitly:

$$\int_0^T \left| c_H \int_0^\infty e^{-xt} x^{-H-1/2} dx - \sum_{i=0}^N w_i e^{-tx_i} \right|^2 dt \leq$$
$$3 \int_0^T \left| c_H \int_0^{\xi_0} e^{-xt} x^{-H-1/2} dx - w_0 \right|^2 dt +$$
$$+ 3 \int_0^T \left| c_H \int_{\xi_0}^{\xi_n} e^{-xt} x^{-H-1/2} dx - \sum_{i=1}^N w_i e^{-tx_i} \right|^2 dt +$$
$$+ 3 \int_0^T \left| c_H \int_{\xi_n}^\infty e^{-xt} x^{-H-1/2} dx \right|^2 dt$$

Now, for the first term:

$$\left| c_H \int_0^{\xi_0} e^{-xt} x^{-H-1/2} dx - w_0 \right| = c_H \int_0^{\xi_0} \left(1 - e^{-tx}\right) x^{-H-1/2} \leq c_H \int_0^{\xi_0} txx^{-H-1/2} dx$$

$$\left| c_H \int_0^{\xi_0} e^{-xt} x^{-H-1/2} dx - w_0 \right| \leq \frac{c_H}{3/2-H} t \xi_0^{3/2-H} = \frac{c_H}{3/2-H} t \exp\left(-\frac{\alpha}{A}\sqrt{N}\right)$$

Then integrating and squaring

$$\int_0^T \left| c_H \int_0^{\xi_0} e^{-xt} x^{-H-1/2} dx - w_0 \right|^2 dt \leq \frac{c_H^2 T^3}{3(3/2-H)^2} \exp\left(-\frac{2\alpha}{A}\sqrt{N}\right)$$

As for the last term,

$$\int_0^T \left| c_H \int_{\xi_n}^\infty e^{-xt} x^{-H-1/2} dx \right|^2 dt$$
$$= c_H^2 \int_0^T \int_{\xi_n}^\infty \int_{\xi_n}^\infty e^{-t(x+y)} x^{-H-1/2} y^{-H-1/2} dy dx dt \leq$$
$$\leq c_H^2 \int_{\xi_n}^\infty \int_{\xi_n}^\infty \int_0^\infty e^{-t(x+y)} dt x^{-H-1/2} y^{-H-1/2} dy dx =$$
$$= c_H^2 \int_{\xi_n}^\infty \int_{\xi_n}^\infty \frac{x^{-H-1/2} y^{-H-1/2}}{x+y} dy dx \leq$$
$$\leq \frac{c_H^2}{2} \int_{\xi_n}^\infty \int_{\xi_n}^\infty \frac{x^{-H-1/2} y^{-H-1/2}}{\sqrt{xy}} dy dx$$
$$= \frac{c_H^2}{2H^2} \xi_n^{-2H} = \frac{c_H^2}{2H^2} \exp\left(-\frac{2\alpha}{A}\sqrt{N}\right)$$

Finally, using lemma 2 and adding the pieces together:

$$\int_0^T \left| c_H \int_0^\infty e^{-xt} x^{-H-1/2} dx - \sum_{i=0}^N w_i e^{-tx_i} \right|^2 dt \leq$$

$$3 \left(\frac{c_H^2 T^3}{3(3/2-H)^2} \exp\left(-\frac{2\alpha}{A}\sqrt{N}\right) + \frac{5\pi^3}{36} \frac{c_H^2 T^{2H}}{H} \frac{n^2}{m^{2H}} \left(\frac{1}{2}\left(e^{\alpha\beta}-1\right)\right)^{4m+2} + \frac{c_H^2}{2H^2} \exp\left(-\frac{2\alpha}{A}\sqrt{N}\right) \right) =$$

And now inserting the values of n, m they get:

$$3c_H^2 \left(\frac{T^3}{3(3/2-H)^2} \exp\left(-\frac{2\alpha}{A}\sqrt{N}\right) + \frac{5\pi^3}{36} \frac{A^{2-2H} T^{2H}}{H\beta^{2-2H}} N^{1-H} \left(\frac{1}{2}\left(e^{\alpha\beta}-1\right)\right)^{4m+2} + \frac{1}{2H^2} \exp\left(-\frac{2\alpha}{A}\sqrt{N}\right) \right)$$

□

Also, recalling that $m = \frac{\beta}{A}\sqrt{N}$ then

$$\left(\frac{1}{2}\left(e^{\alpha\beta}-1\right)\right)^{4m} = \exp\left(\log\left(\frac{1}{2}(e^{\alpha\beta}-1)\right)\frac{4\beta}{A}\sqrt{N}\right)$$

Then, in order to find the optimal values for α, β, they maximise α, constrained to:

$$\log\left(\frac{1}{2}(e^{\alpha\beta}-1)\right)\frac{4\beta}{A}\sqrt{N} = -\frac{2\alpha}{A}\sqrt{N}$$

So that all the three terms have the same decay in N. The values that they find, for $H = 0.1, \alpha/A \approx 0.3251$.

5.3.3. Improvements on the rate of convergence

In this section we are going to present the quadrature rule given by Bayer and Breneis [8], what they call Geometric Gaussian (GG) and Non-Geometric Gaussian (NGG), for which theyt give the following asymptotic rates of convergence:

$$\text{for GG} : 2(\sqrt{2}+1)^{-2\sqrt{(H+1/2)N}}$$
$$\text{for NGG} : 60e^{-2.38\sqrt{(H+1/2)N}}$$

Let us start with the Geometric Gaussian. The main difference from what we have discussed in the previous section is the following: in the paper [7], they divide the half positive real line in the following way: given a partition of the half positive real line $\{\xi_i\}_{i=0}^n \subset \mathbb{R}^+$ they applied the following rule:

- from $[0, \xi_0]$ they approximate the integrand with a constant w_0.
- in the interval $[\xi_0, \xi_n]$ they apply Gaussian quadrature rule up to level m in each one of the sub-intervals $[\xi_i, \xi_{i+1}]_{i=1}^{n-1}$ with respect to the weight function as in (5.19).

Instead, in the last paper [8], their approach is slightly different:

- from $[0, \xi_0]$ they approximate the integral with a Gaussian Quadrature rule of level m with respect to the weight function as in (5.19).
- in the interval $[\xi_0, \xi_n]$ they apply Gaussian quadrature rule up to level m in each one of the sub-intervals $[\xi_i, \xi_{i+1}]_{i=1}^{n-1}$ with respect to the weight function $w \equiv 1$.

In fact, they use the following definition for Geometric Gaussian Quadrature rule [8, Def. 3.1]:

Definition 5.3.2 (Geometric Gaussian quadrature rule.)**.** Let $N \in \mathbb{N}$ the total number of nodes and $\alpha, \beta, a, b \in (0, \infty)$ the parameters of the scheme. Define:

$$m := \text{round}\left(\beta\sqrt{(H+1/2)\,N}\right), \quad n := \text{round}\left(\frac{1}{\beta}\sqrt{\frac{N}{H+1/2}}\right)$$

$$\xi_0 = 0, \quad \xi_n := b\exp\left(\frac{\alpha\sqrt{N}}{\sqrt{1/2+H}}\right), \quad \xi_i = a\left(\frac{\xi_n}{a}\right)^{\frac{i}{n}}, \quad i = 1, \dots, n$$

Let $\{x_j\}_{j=1}^m$,$\{\tilde{w}_j\}_{j=1}^m$ be respectively the nodes and the weights of a Gaussian quadrature of level m on the interval $[0, \xi_1]$ with respect to the weight function as in (5.19). Also let $\{x_j\}_{j=im+1}^{(i+1)m}$,$\{\tilde{w}_j\}_{j=im+1}^{(i+1)m}$ be the nodes and weights of a Gaussian quadrature of level m on the intervals $[\xi_i, \xi_{i+1}]$ for $i = 1, \dots, n-1$ with weight function $w \equiv 1$. Then define the Geometric Gaussian Rule of type $(H, N, \alpha, \beta, a, b)$ to be the set of nodes and weights $\{x_i, w_i\}_{i=1}^{mn}$ defined as:

$$w_j := \tilde{w}_j, \quad j = 1, \dots, m \tag{5.20}$$
$$w_j := c_H \tilde{w}_j x_j^{-H-1/2}, \quad j = m+1, \dots, mn \tag{5.21}$$

Where c_H is defined as in (6.9), and they assume, without affecting the convergence rate that the product mn is integer-valued.

Now, with this choice of nodes and weights Bayer proves the following Theorem that gives an asymptotic convergence rate.

Theorem 3. [8, Theorem 3.9]. Let $(x_i, w_i)_{i=1}^N$ the nodes and the weights of the Geometric Gaussian quadrature rule with $\alpha = \log\left(3 + 2\sqrt{2}\right), \beta = 1$, $a = \frac{10\sqrt{2}-14}{eT}\sqrt{(H+1/2)N}$, $b = \frac{10\sqrt{2}-14}{eT}$ then one has:

$$\int_0^T |g(t) - \hat{g}_N(t)|\ dt \leq \frac{c_H}{H+1/2}\left(\frac{eT}{10\sqrt{2}-14}\right)^{H+1/2}\left(\sqrt{2}+1\right)^{-2\sqrt{(H+1/2)N}} \tag{5.22}$$

Other than the Geometric Gaussian rule, in [8] they introduce a so called Non-Geometric Gaussian quadrature. The choice of weights used for the Gaussian rule is the same as for the Geometric, what changes is the partition of the positive half real line. In fact, they define:

Definition 5.3.3. Non-Geometric Gaussian quadrature rule [8, Def 3.11]. Let $N \in \mathbb{N}$ the total number of quadrature points used in the rule; let $\beta, a, c \in (0, \infty)$ define:

$$m := \text{round}\left(\beta\sqrt{(H+1/2)\,N}\right), \quad n := \text{round}\left(\frac{1}{\beta}\sqrt{\frac{N}{H+1/2}}\right)$$

$$\xi_0 = 0, \quad \xi_1 = a, \quad \xi_{i+1} := \left(\frac{c + \xi_i^{\frac{1/2+H}{2m}}}{c - \xi_i^{\frac{1/2+H}{2m}}}\right)\xi_i, \quad i = 1, \ldots, n-1$$

Here, they assume $c > \xi_i^{\frac{1/2+H}{2m}}$, $\forall i = 1, \ldots, n-1$. Then, the nodes and the weights for the quadrature are defined exactly the same as for the Geometric Gaussian rule, using Gaussian quadrature of level m with respect to the weight function $\omega(x) = c_H x^{-H-1/2}$ on the interval $[0, \xi_0]$, while for the others intervals the weight function considered becomes $\omega(x) \equiv 1$.

In this setting, they prove the following theorem, which again gives an asymptotic estimate for the error:

Theorem 4. [8, Theorem 3.16] Define $\beta_0 = 0.92993273$ and $c_0 = 3.6058502$, Let $\hat{g}_N$ a Gaussian approximation coming from a Non-Geometric Gaussian rule with parameters $c \geq c_0, \beta \geq \beta_0, a > 0$, where either β or c are strictly greater than β_0, c_0 respectively. Then:

$$\int_0^T |g(t) - \hat{g}_N(t)| \leq \frac{c_H}{H+0.5}\xi_n^{-H-1/2} \leq C\exp\left(-2.38\sqrt{(H+0.5)N}\right) \tag{5.23}$$

5.3.4. Optimal Algorithms

In this section, we are going to present three different algorithms, all from the paper by Bayer and Breneis [8]. The name of this section comes from the fact that those algorithms, do not come from a specific quadrature rule where the nodes and weights are specified to give a certain error bound but they are the result of an optimization algorithm.
For an approximation of the type:

$$\hat{g}_N(t) = \sum_{i=1}^N w_i e^{-x_i t}, \quad w_i \in \mathbb{R}, \quad x_i \in (0, \infty), \quad \forall i = 1, \ldots, N$$

The algorithms that we are going to present are the following:

- OL2: Optimal L2 error, for which it optimizes the $\mathcal{L}^2([0,T])$ between the kernel g and its approximation $\hat{g}$.
- BL2: Bounded Optimal L2 error, for which it optimizes the $\mathcal{L}^2([0,T])$ between the kernel g and its approximation $\hat{g}$, but penalizes large nodes.
- OL1: Optimal L1 error, for which it optimizes the $\mathcal{L}^1([0,T])$ between the kernel g and its approximation $\hat{g}$.

Let us start with the first algorithm. In order to have an optimization algorithm for the $\mathcal{L}^2([0,T])$ error, Bayer uses the following proposition [7], that is the result of a simple calculation of a definite integral:

Proposition 11. [7, Prop 2.12] Let $(x_i, w_i)_{i=0}^N$ the nodes and weights such that $x_i > 0, \forall i = 1, \dots, N$, $x_0 = 0$ and $w_i \in \mathbb{R}, \forall i = 0, \dots, N$. Let:

$$\hat{g}_N(t) = \sum_{i=0}^{N} w_i e^{-x_i t}, \quad g(t) = \frac{1}{\Gamma(H+1/2)} t^{H-1/2}$$

Then, the following equality holds:

$$\begin{aligned} \int_0^T |\hat{g}_N(t) - g(t)|^2 \, dt = & \frac{T^{2H}}{2H\Gamma(H+1/2)^2} + w_0^2 T \\ & + 2w_0 \sum_{i=1}^{N} \frac{w_i}{x_i} \left(1 - e^{-x_i T}\right) \\ & + \sum_{i,j=1}^{N} \frac{w_i w_i}{x_i + x_j} \left(1 - e^{-(x_i + x_j)T}\right) \\ & - \frac{2w_0 T^{H+1/2}}{\Gamma(H+3/2)} - \frac{2}{\Gamma(H+1/2)} \\ & \times \sum_{i=1}^{N} \frac{w_i}{x_i^{H+1/2}} \int_0^{x_i T} t^{H-1/2} e^{-t} dt \end{aligned} \tag{5.24}$$

with this proposition, the OL2 algorithm, is simply given by optimizing (5.24) with respect to the nodes and weights $(x_i, w_i)_{i=1}^N$, using some optimization algorithms.
As for the second algorithm, the Bounded L2, as we said before, the main idea is still to optimizing equation (5.24), but now penalizing having large nodes. Say we want a quadrature rule with N points, with $N > 1$.

In [8], Bayer and Breneis propose the following method to optimize the nodes and weights of a quadrature rule for a given Hurst parameter $H \in (-1/2, 1/2)$, time horizon $T > 0$, and number of nodes $N \in \mathbb{N}$. For a fixed $\epsilon \in (0,1)$ and for a fixed $q \in (1, \infty)$:

1. If $N = 1$ return the result from the OL2 method.
2. Set the side length L of the hypercube to 1.
3. Use an optimization algorithm to search for the nodes within the hypercube $[0, L]^N$, and save the rule, $rule_N$ and the error err_N
4. Use an optimization algorithm to search for the nodes within the hypercube $[0, L]^{N-1}$, and save the rule, $rule_{N-1}$ and the error err_{N-1}
5. if $err_N < (1 - \epsilon) err_{N-1}$, meaning that there is a relative improvement of the error of at least ϵ then return $rule_N$
6. otherwise, increase the size of the search space by setting the side of the hypercube to $L = qL$, for some $q \in (0, \infty)$ and then repeat the algorithm until termination from point 3

This method systematically increases the search space for the quadrature nodes and weights, iteratively adjusting the hypercube's side length to minimize the $\mathcal{L}^2([0,T])$ error. In the paper of Bayer and Breneis [8], this algorithm is the one that is capable to obtain the best results in every way possible: the size of the largest node remains relatively small, especially if compared with the nodes given by the OL2 algorithm. Both the strong and the weak convergence outperform every other quadrature rule that we have presented so far.
Lastly, we will present the OL1 algorithm, which is the last algorithm of this type introduced by Bayer and Breneis in [8]. The idea is still the same: optimize the $\mathcal{L}^1([0,T])$ error. The only difference is that there is no analytic expression available like for the $\mathcal{L}^2([0,T])$ error as in (5.24). In Appendix E of [8] they explain how to compute the following quantity:

$$\int_0^T |g(t) - \hat{g}_N(t)| \, dt$$

in particular the idea of the algorithm is the following: first compute all the intersections between g and $\hat{g}_N$, say that there are M intersection: $\{t_i\}_{i=1}^{M}$. Then, when one has all the intersections between the kernel and its approximation, then one can compute exactly the integral above in the following way:

$$\int_0^T |g(t) - \hat{g}_N(t)|\, dt = \sum_{i=1}^{M-1} \left| \int_{t_i}^{t_{i+1}} g(t) - \hat{g}_N(t) dt \right|$$

where in the right hand side of the equation, one can compute the integrals explicitly. The main problem of this method, is that the root finding algorithm, even if it is sure to converge and find all the roots, may take a long time to compute. Already for $N = 7$, the computational time exceeds the minute in computing time. However, this method, being far from fast, it's very efficient and yields very good results in terms both of weak and strong convergence. Also, the size of the nodes remains relatively small compared to the OL2.

6

Sinc: How it works

In this section, we will introduce the theory behind SINC quadrature. The main idea behind it, is to use first the Laplace transform of the rough kernel g, which is an integral over the real positive half line and transform it into an integral over the whole real line. After this change of variable, we successfully remove the singularity of the integrand in zero and obtain a function that is analytic over the whole real line. With this new integrand, we employ a simple trapezoidal rule. Thanks to the regularity of the new integrand, we can obtain error bounds that decay exponentially in the total number of nodes used in the quadrature rule.

We will start by presenting the main tool from complex analysis that we will use: the Residue Theorem. Following that, we will discuss the results from [36], where it is shown that for a relatively large class of functions defined in a domain in the complex plane, one can achieve high accuracy using a very simple quadrature rule: the trapezoidal rule. We will reproduce the main proofs given by Lund and Bowers, which will provide the key ideas for arriving at an efficient solution to our problem.

6.1. Preliminaries

In this section we introduce the key concepts and results from complex analysis that will be needed for the computation of the error bound of the SINC quadrature. We will not provide a proof of the Residue Theorem here, as it can be found in many introductory texts on complex analysis, such as [40].

Theorem 5 (Residue Theorem)**.** Let Ω a domain[1],if f is analytic in $\Omega \setminus \{z_j\}_{j=1}^n$ and $\{z_j\}_{j=1}^n$ is contained in the interior of the simple closed contour $\gamma \subset \Omega$ then:

$$\int_\gamma f(z)\, dz = 2\pi i \sum_{j=0}^{n} Res(f, z_j),$$

where if the singularity is a simple pole one can compute the residue as

$$Res(f, z) := \lim_{x \to z} f(x)(z - x)$$

Now the main idea, is that we need to compute an integral over the real line, so we that f is analytic in a domain containing $\mathbb{R}$.

Definition 6.1.1. We say that D_d denotes the infinite strip domain of height $2d$, $d > 0$ where D_d is given by:

$$D_d := \{z \in \mathbb{C} \mid z := x + iy,\ x \in \mathbb{R},\ |y| < d\} \tag{6.1}$$

Now, let's introduce the class of functions for which we can apply the SINC quadrature.

[1] For a Domain we use the standard definition being: a domain Ω is defined as an open, simply connected subset of $\mathbb{C}$, where simply connected means that $\mathbb{C} \cup \{\infty\} \setminus \Omega$ is connected.

Definition 6.1.2. Say that $f \in B(D_d)$ if f is analytic in D_d and satisfies:

$$\int_{-d}^{d} |f(x+iy)|\, dy = \mathcal{O}\left(|x|^{\alpha}\right), \quad x \to \infty, \quad \alpha \in [0,1) \tag{6.2}$$

Also, f must satisfy the following condition:

$$N(f, D_d) := \lim_{y \to d^-} \int_{-\infty}^{\infty} |f(x+iy)| + |f(x-iy)| dx < \infty. \tag{6.3}$$

Remark. Condition (6.2) asks the integral of the function f over vertical segments in the complex plane to grow at most like a power law function, with exponent less than one. This condition will be crucial in computing exactly the error that one has when using the trapezoidal rule. Condition (6.3) instead will be part of the estimate for the error.

Now that we have the main tools and definitions, we can move on and reproduce the proof of [36, Theorem 2.13]. First they give an analytic formula for the error that one has when using the SINC basis functions for interpolating a function.

Theorem 6. [36, Theorem 2.13] Let $f \in B(D_d)$ and $h > 0$, then:

$$\begin{aligned} f(x) - \sum_{k=-\infty}^{\infty} f(kh) sinc\left(\frac{(x-kh)}{h}\right) = & \\ = \frac{\sin\left(\frac{\pi x}{h}\right)}{2\pi i} \int_{-\infty}^{\infty} \left(\frac{F(x, u - id^-)}{\sin\left(\pi\left(u - id^-\right)/h\right)} - \frac{F(x, u + id^-)}{\sin\left(\pi\left(u + id^-\right)/h\right)} \right) du & \\ := S_h(x) I(f,h)(x) & \\ := \varepsilon(x) & \end{aligned} \tag{6.4}$$

Where, define:

$$\forall x \in \mathbb{R}, \quad sinc(x) = \frac{\sin(\pi x)}{\pi x} \chi_{\mathbb{R}\setminus\{0\}}(x) + \chi_{\{0\}}(x),$$

where $\chi_A(x)$ is the characteristic function of the set A, that yields one if $x \in A$ and 0 if $x \notin A$

$$F(x, u \pm iv) = \frac{f(u \pm iv)}{u \pm iv - x}$$

$$S_h(x) := \frac{\sin\left(\frac{\pi x}{h}\right)}{2\pi i}$$

$$I(f,h)(x) = \int_{-\infty}^{\infty} \left(\frac{F(x, u - id^-)}{\sin\left(\pi\left(u - id^-\right)/h\right)} - \frac{F(x, u + id^-)}{\sin\left(\pi\left(u + id^-\right)/h\right)} \right) du$$

Proof. Let $y_n = d - \frac{1}{n}$ and define the domain:

$$R_n := \left\{ z \in \mathbb{C} \ : \ z = x + iy \ \ |y| < y_n, \ -\left(n + \frac{1}{2}\right) h < x < \left(n + \frac{1}{2}\right) h \right\}$$

Define also, ∂R_n as the boundary of R_n. It is clear than that $R_n \to D_d$ as $n \to \infty$. Applying the Residue Theorem to the integral of

$$g(z) = \frac{\sin\left(\frac{\pi x}{h}\right) f(z)}{(z-x)\sin\left(\frac{\pi z}{h}\right)}$$

One obtains:

$$\begin{aligned} I_n &:= S_h(x) \int_{\partial R_n} \frac{f(z)dz}{(z-x)\,\mathsf{sin}\left(\frac{\pi z}{h}\right)} \\ &= f(x) - \sum_{k=-n}^{n} f(kh) sinc\left(\frac{(x-kh)}{h}\right) \end{aligned}$$

So, since I_n is a sequence of rectangles that converges to the infinite strip D_d and also from the last equation, one can easily see that

$$\lim_{n\to\infty} I_n = \varepsilon(x)$$

where $\varepsilon(x)$ being the same as in (6.4).

Also, integrating around ∂R_n we get exactly four terms, one for each side of the rectangle that we are integrating on:

$$\begin{aligned} I_n(x) = S_h(x) &\left\{ \int_{-y_n}^{y_n} \frac{F\left(x,(n+1/2)h+iy\right)idy}{\mathsf{sin}\left(\pi\left((n+1/2)h+iy\right)/h\right)} + \int_{-y_n}^{y_n} \frac{F\left(x,-(n+1/2)h+iy\right)idy}{\mathsf{sin}\left(\pi\left(-(n+1/2)h+iy\right)/h\right)} \right\} \\ + S_h(x) &\left\{ \int_{(n+1/2)h}^{-(n+1/2)h} \frac{F\left(x,u+iy_n\right)du}{\mathsf{sin}\left(\pi\left(u+iy_n\right)/h\right)} + \int_{-(n+1/2)h}^{(n+1/2)h} \frac{F\left(x,u-iy_n\right)du}{\mathsf{sin}\left(\pi\left(u-iy_n\right)/h\right)} \right\} \end{aligned} \tag{6.5}$$

Now, for the first two terms, i.e. the terms representing the vertical lines on the rectangle ∂R_n we have:

$$\begin{aligned} &\mathsf{sin}\left[\pi\left((n+1/2)h+iy\right)/h\right] \\ &= \pm\,\mathsf{sin}\left(\frac{2n+1}{2}\pi\right)\mathsf{cosh}\left(\pi y/h\right) + i\,\mathsf{cos}\left(\frac{2n+1}{2}\pi\right)\mathsf{sinh}\left(\pi y/h\right) \\ &= \pm(-1)^n\,\mathsf{cosh}\left(\pi y/h\right) \end{aligned}$$

Then:

$$\left|\mathsf{sin}\left[\pi\left((n+1/2)h+iy\right)/h\right]\right| = \mathsf{cosh}\left(\pi y/h\right) \geq 1$$

And also:

$$\left|\pm\left(n+1/2\right)h+iy-x\right| \geq \left|\pm\left(n+1/2\right)h-x\right|$$

So we can bound:

$$\begin{aligned} &\left| S_h(x) \left\{ \int_{-y_n}^{y_n} \frac{F\left(x,(n+1/2)h+iy\right)idy}{\mathsf{sin}\left(\pi\left((n+1/2)h+iy\right)/h\right)} + \int_{-y_n}^{y_n} \frac{F\left(x,-(n+1/2)h+iy\right)idy}{\mathsf{sin}\left(\pi\left(-(n+1/2)h+iy\right)/h\right)} \right\} \right| \\ &\leq \int_{-y_n}^{y_n} \frac{\left|f\left(x,(n+1/2)h+iy\right)\right|}{\left|(n+1/2)\,h-x\right|}dy + \int_{-y_n}^{y_n} \frac{\left|f\left(x,-(n+1/2)h+iy\right)\right|}{\left|(n+1/2)\,h+x\right|}dy \to 0 \end{aligned}$$

as $n \to \infty$, using the hypothesis (6.2). So that we have:

$$\begin{aligned} \varepsilon(x) &= \lim_{n\to\infty} I_n(x) \\ &= S_h(x) \lim_{n\to\infty} \int_{-(n+1/2)h}^{(n+1/2)h} \left\{ \frac{F\left(x,u-iy_n\right)}{\mathsf{sin}\left(\pi\left(u-iy_n\right)/h\right)} - \frac{F\left(x,u+iy_n\right)du}{\mathsf{sin}\left(\pi\left(u+iy_n\right)/h\right)} \right\} du \\ &= S_h(x) \int_{-\infty}^{\infty} \left\{ \frac{F\left(x,u-id^-\right)}{\mathsf{sin}\left(\pi\left(u-id^-\right)/h\right)} - \frac{F\left(x,u+id^-\right)du}{\mathsf{sin}\left(\pi\left(u+id^-\right)/h\right)} \right\} du \end{aligned}$$

And this shows the thesis (6.4).

Instead, if we want to have a bound for the modulus of the error, then if we consider the following two inequalities:

$$\left|u \pm id - x\right| \geq d \tag{6.6}$$

$$\left|\mathsf{sin}\left[\pi\left(u \pm id\right)/h\right]\right| = \left|\mathsf{sin}(\pi u/h)\,\mathsf{cosh}\left(\pi d/h\right) \pm i\,\mathsf{cos}\left(\pi u/h\right)\mathsf{sinh}\left(\pi d/h\right)\right| \geq \mathsf{sinh}\left(\pi d/h\right) \tag{6.7}$$

So using (6.6) and (6.7) then one gets:

$$\begin{aligned}|\varepsilon(x)| &= |S_h(x)| \left| \int_{-\infty}^{\infty} \left\{ \frac{F(x, u - id^-)}{\sin(\pi(u - id^-)/h)} - \frac{F(x, u + id^-)\, du}{\sin(\pi(u + id^-)/h)} \right\} du \right| \\ &\leq \frac{1}{2\pi d \sinh(\pi d/h)} \int_{-\infty}^{\infty} \left\{ \left| f\left(u + id^-\right) + f\left(u - id^-\right) \right| \right\} du \\ &\leq \frac{N(f, D_d)}{2\pi d \sinh(\pi d/h)}\end{aligned}$$

□

This theorem however, it's not directly useful for our purpose, but it will be used for showing the that in fact the error of the trapezoidal rule, decays in fact exponentially for all the functions in the class $B(D_d)$. In particular we have:

Theorem 7. [36, Theorem 2.20] If $f \in B(D_d)$and $h > 0$ then:

$$\int_{-\infty}^{\infty} f(x)\, dx - h \sum_{k=-\infty}^{\infty} f(kh) \equiv \eta \tag{6.8}$$

where:

$$\begin{aligned}\eta &= \int_{-\infty}^{\infty} \varepsilon(x)\, dx \\ &= \frac{e^{-\pi d/h}}{2i} \int_{-\infty}^{\infty} \left[\frac{f(u + id^-)\, e^{i\pi u/h}}{\sin(\pi(u + id^-)/h)} - \frac{f(u - id^-)\, e^{-i\pi u/h}}{\sin(\pi(u - id^-)/h)} \right] du\end{aligned}$$

Proof. Integrate equation (6.4), i.e.:

$$\begin{aligned}&\int_{-\infty}^{\infty} f(x)\, dx - \int_{-\infty}^{\infty} \sum_{k=-\infty}^{\infty} f(kh) sinc\left(\frac{(x - kh)}{h} \right) dx = \\ &= \frac{\sin\left(\frac{\pi x}{h}\right)}{2\pi i} \int_{-\infty}^{\infty} \left(\frac{F(x, u - id^-)}{\sin(\pi(u - id^-)/h)} - \frac{F(x, u + id^-)}{\sin(\pi(u + id^-)/h)} \right) du\end{aligned}$$

Now, the right hand side of the equation above can be elaborated using again the Residue Theorem, in fact:

$$\int_{-\infty}^{\infty} sinc\left(\frac{(x - kh)}{h} \right) = \int_{-\infty}^{\infty} \frac{\sin(\pi(x - kh)/h)}{\pi(x - kh)/h} = h$$

so that the right hand side becomes exactly the error between the integral of f and the integral computed with trapezoidal rule:

$$\int_{-\infty}^{\infty} \sum_{k=-\infty}^{\infty} f(kh) sinc\left(\frac{(x - kh)}{h} \right) dx = h \sum_{k=-\infty}^{\infty} f(kh)$$

On the other hand, one has:

$$\begin{aligned}&\int_{-\infty}^{\infty} \frac{\sin\left(\frac{\pi x}{h}\right)}{2\pi i} \int_{-\infty}^{\infty} \frac{F(x, u \pm id^-)}{\sin(\pi(u \pm id^-)/h)} du dx \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{\sin\left(\frac{\pi x}{h}\right)}{2\pi i(u \pm id^- - x)} \frac{f(u \pm id^-)}{\sin(\pi(u \pm id^-)/h)} dx du\end{aligned}$$

Integrating first in x, we have that:

$$\int_{-\infty}^{\infty} \frac{\sin(\pi x/h)}{u \pm id^- - x} dx = -\pi e^{i\pi\xi sign(\Im(\xi))}, \quad \xi = u \pm id^-$$

where $\Im$ is the imaginary part of a complex number. So that, putting all the pieces together, one obtains the thesis. Moreover, If we want to bound the absolute value of the error, then one has:

$$\begin{aligned}
|\eta| &\leq e^{-\pi d/h} \int_{-\infty}^{\infty} \left[\left| \frac{f\left(u+id^{-}\right) e^{i\pi u/h}}{\sin\left(\pi\left(u+id^{-}\right)/h\right)} \right| + \left| \frac{f\left(u-id^{-}\right) e^{-i\pi u/h}}{\sin\left(\pi\left(u-id^{-}\right)/h\right)} \right| \right] du \\
&= e^{-\pi d/h} \int_{-\infty}^{\infty} \left[\frac{\left|f\left(u+id^{-}\right)\right|}{\left|\sin\left(\pi\left(u+id^{-}\right)/h\right)\right|} + \frac{\left|f\left(u-id^{-}\right)\right|}{\left|\sin\left(\pi\left(u-id^{-}\right)/h\right)\right|} \right] du \\
&\leq \frac{e^{-\pi d/h}}{\sinh\left(\pi d/h\right)} \int_{-\infty}^{\infty} \left|f\left(u+id^{-}\right)\right| + \left|f\left(u-id^{-}\right)\right| du = \frac{e^{-\pi d/h} N(f, D_s)}{\sinh\left(\pi d/h\right)}
\end{aligned}$$

where it has been used the bound for the denominator as in (6.7) □

Now, this theorem made us clear that it is possible to achieve exponential convergence in the step-size h. However, one needs also to ask the integrand itself to decay such a way, that the tails of the integral can keep up with an exponential decay of the SINC quadrature error. In fact, we need to deal with truncated series of the trapezoidal rule.In order to give some heuristics about this we will reproduce the idea of the proof of Theorem 2.21 of [36]. Suppose that we have a positive function $f \in B\left(D_d\right)$ and also suppose that $\exists C, \alpha \in \mathbb{R}^{+}$ such that:

$$f : \mathbb{R} \to \mathbb{R}^{+}, \quad f(x) \leq Ce^{-\alpha|x|}, \quad x \to \pm\infty$$

and say we have two integers $M, N \in \mathbb{N}$. To compute the absolute error between the truncated trapezoidal rule with step size h and the integral of f one has:

$$\left| \int_{\mathbb{R}} f(x)dx - h \sum_{k=-M}^{N} f(kh) \right| \leq \left| \int_{\mathbb{R}} f(x)dx - h \sum_{k=-\infty}^{\infty} f(kh) \right| + h \sum_{k=M+1}^{\infty} f(-kh) + h \sum_{k=N+1}^{\infty} f(kh)$$

We already know, from theorem (7), that the first term of the equation above, has a decay that is proportional to $e^{-\pi d/h}$. So in this case, where f becomes exponentially small for $x \to \pm\infty$, then one can bound the tails of the series with[2]:

$$h \sum_{k=M+1}^{\infty} f(kh) \leq hC \sum_{k=M+1}^{\infty} e^{-\alpha|kh|} < \frac{hC}{1-e^{-\alpha h}} e^{-\alpha M h}$$

and with the same reasoning, one can bound the other tail of the series. In this way, one is sure to have exponential convergence rate also for the truncated trapezoidal rule.

6.2. SINC Quadrature

Now, we are going to introduce a quadrature rule for the fractional kernel that exploits a change of variable of the Laplace transform of $G(\tau) = \frac{\tau^{H-1/2}}{\Gamma(H+1/2)}$, for $H \in (-1/2, 1/2), \tau > 0$.

So, the Laplace transform reads:

$$\frac{\tau^{H-1/2}}{\Gamma(H+1/2)} = c_H \int_0^{\infty} e^{-\rho\tau} \rho^{-H-1/2} d\rho$$

$$c_H = \frac{1}{\Gamma(H+1/2)\Gamma(1/2-H)} \tag{6.9}$$

Now, change variables so that the integral on the positive real half-line becomes an integral on the whole real line:

$$log\rho = x \to \frac{d\rho}{\rho} = dx$$

[2]the inequality comes from the fact that $\forall r \in (0,1)$ and $\forall K \in \mathbb{N}$ we have that $\sum_{k=K+1}^{\infty} r^k = \frac{r^{K+1}}{1-r} < \frac{r^K}{1-r}$

So that we have:

$$G(\tau) = c_H \int_0^\infty e^{-\rho\tau}\rho^{-H-1/2+1}\frac{1}{\rho}d\rho = c_H \int_{-\infty}^\infty e^{-\tau e^x+(1/2-H)x}dx$$

i.e.:

$$G(\tau) = c_H \int_{-\infty}^\infty e^{-\tau e^x+(1/2-H)x}dx$$

First, fix $H \in (-1/2, 1/2)$, $T > 0$ and define the family of functions:

$$\mathcal{G}_H := \left\{ f_\tau : \mathbb{C} \to \mathbb{C} : f_\tau(z) = c_H e^{-\tau e^z+(1/2-H)z} \right\}_{\tau\in(0,T]} \tag{6.10}$$

with c_H as in (6.9) To apply the SINC quadrature, we need to check first that there exists a $d > 0$ such that $\mathcal{G}_H \subset B(D_d)$. We first note that $\forall f_\tau \in \mathcal{G}_H$, we have that f_τ is entire. Also, the modulo of f_τ reads:

$$\forall z \in \mathbb{C} : z = x + iy \Rightarrow |f_\tau(z)| = c_H|e^{-\tau e^{x+iy}+(1/2-H)(x+iy)}| = e^{-\tau e^x cosy}e^{(1/2-H)x}$$

So that in particular:

$$|f_\tau\left(x + iy\right)| = |f_\tau\left(x - iy\right)|$$

Now, we want a constant $d > 0$ such that for every $\tau > 0$ one has $f_\tau \in B(D_d)$. So this would require:

$$\exists d \in \mathbb{R}^+, \exists a \in (0, 1) : \quad \int_{-d}^d |f_\tau(x + iy)|dy = \mathcal{O}(|x|^a), \quad \forall\tau > 0$$

and in fact we have, $\forall 0 < d < \pi/2$:

$$\int_{-d}^d |f_\tau(x + iy)|dy = \int_{-d}^d e^{-\tau e^x cosy}e^{(1/2-H)x}dy \leq c_H 2de^{-\tau e^x cosd}e^{(1/2-H)x}$$

with the right hand side being $\mathcal{O}(|x|^a), \forall a \in (0, 1)$ since both x limits $\pm\infty$ tend to zero. Note that here it is necessary for d to be strictly less than $\pi/2$, otherwise the limit for $x \to +\infty$ would diverge more than exponentially. Moreover, since the modulo of f_τ depends only on the cosine of y in $[-\pi/2, \pi/2]$ (6.2) becomes

$$\begin{aligned} N(f_\tau, D_S) &= \lim_{y\to d^-} \int_{-\infty}^\infty |f_\tau(x + iy)| + |f_\tau(x - iy)|dx = 2 \lim_{y\to d^-} \int_{-\infty}^\infty |f_\tau(x + iy)|dx \\ &= 2 \lim_{y\to d^-} \int_{-\infty}^\infty c_H e^{-\tau e^x cosy}e^{(1/2-H)x}dx = 2c_H \lim_{y\to d^-} (\tau \cos y)^{H-1/2}\Gamma(1/2 - H) \end{aligned}$$

i.e.:

$$N(f_\tau, D_S) = 2c_H(\tau \cos d)^{H-1/2}\Gamma(1/2 - H) < \infty \quad \forall\tau > 0 \tag{6.11}$$

We just have shown then, that if $d \in (0, \pi/2)$ then $f_\tau \in B(D_d), \forall\tau > 0$.

Now, using triangular inequality one can obtain an estimate for the absolute error:

$$\left| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-M}^{N} f_\tau(kh) \right| = \left| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-\infty}^{\infty} f_\tau(kh) + h \sum_{k=-\infty}^{-M-1} f_\tau(kh) + h \sum_{k=N+1}^{\infty} f_\tau(kh) \right| \tag{6.12}$$

The first term, can be bounded using theorem 7 by:

$$\left| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-\infty}^{\infty} f_\tau(kh) \right| \leq \frac{N(f_\tau, D_S)e^{-\pi d/h}}{\sinh(\pi d/h)} = \frac{c_H(\tau \cos d)^{H-1/2}\Gamma(1/2 - H)e^{-\pi d/h}}{\sinh(\pi d/h)} \tag{6.13}$$

The second term in (6.12), instead we can easily bound it by:

$$x \in (-\infty, 0) \Rightarrow f_\tau(x) = c_H e^{-\tau e^x + (1/2-H)x} \leq c_H e^{(1/2-H)x} \Rightarrow$$

$$h \sum_{k=-\infty}^{-M-1} c_H e^{-\tau e^{kh} + (1/2-H)kh} \leq h c_H \sum_{k=M+1}^{\infty} e^{-(1/2-H)kh} \leq h c_H \frac{e^{-(1/2-H)Mh}}{(1/2-H)h} = c_H \frac{e^{-(1/2-H)Mh}}{(1/2-H)}$$

As for the last term, it is easier to compute the bound for the $\mathcal{L}^i([0,T]), i = 1, 2$ instead of the absolute error. In the end, we have obtained that the bound for the absolute error reads:

$$\left| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-M}^{N} f_\tau(kh) \right|$$

$$\leq \frac{c_H(\tau \cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{\sinh(\pi d/h)} + c_H \frac{e^{-(1/2-H)Mh}}{(1/2-H)} + h \sum_{k=N+1}^{\infty} f_\tau(kh) \tag{6.14}$$

Define each one of the terms as:

$$I := \frac{c_H(\tau \cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{\sinh(\pi d/h)} \tag{6.15}$$

$$II := c_H \frac{e^{-(1/2-H)Mh}}{(1/2-H)} \tag{6.16}$$

$$III := h \sum_{k=N+1}^{\infty} f_\tau(kh) \tag{6.17}$$

6.3. Computation of the L1 error

In this section we will compute the $\mathcal{L}^1([0,T])$ error for the SINC quadrature, in particular we want to compute such error for $H \in (-1/2, 1/2)$. Now we will compute, using (6.14):

$$\int_0^T \left| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-M}^{N} f_\tau(kh) \right| d\tau \leq \int_0^T (I + II + III)\, d\tau$$

As for the first two terms (6.15), (6.16), by basic integration rules:

$$\int_0^T \left[\frac{c_H(\tau \cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{\sinh(\pi d/h)} + c_H \frac{e^{-(1/2-H)Mh}}{(1/2-H)} \right] d\tau =$$

$$\frac{c_H T^{H+1/2}(\cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{(H+1/2)\sinh(\pi d/h)} + T c_H \frac{e^{-(1/2-H)Mh}}{(1/2-H)}$$

While for the last term of the sum (6.17), we need to apply Monotone Convergence Theorem, to exchange summation and integration:

$$\int_0^T h \sum_{k=N+1}^{\infty} f_\tau(kh)d\tau = \sum_{k=N+1}^{\infty} h \int_0^T f_\tau(kh)d\tau = \sum_{k=N+1}^{\infty} h \int_0^T c_H e^{-\tau e^{kh} + (1/2-H)kh} d\tau$$

$$\leq h c_H \sum_{k=N+1}^{\infty} \int_0^{\infty} e^{-\tau e^{kh} + (1/2-H)kh} d\tau = h c_H \sum_{k=N+1}^{\infty} e^{-(1/2+H)kh} \leq c_H \frac{e^{-(1/2+H)Nh}}{(1/2+H)}$$

So that the bound for the $\mathcal{L}^1([0,T])$ error reads:

$$\int_0^T \left| \int_{\mathbb{R}} f_\tau(x)dx - h\sum_{k=-M}^{N} f_\tau(kh) \right| d\tau$$

$$\leq \frac{c_H T^{H+1/2}(\cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{(H+1/2)\sinh(\pi d/h)} + Tc_H\frac{e^{-(1/2-H)Mh}}{(1/2-H)} + c_H\frac{e^{-(1/2+H)Nh}}{(1/2+H)}$$

Now, let us choose M, N such that the last two terms have the same exponential rate of convergence. For instance, take $\beta \in (0,\infty)$ and set $M = \lceil \beta N \rceil$, then one has:

$$(1/2-H)\lceil \beta N \rceil h \geq (1/2-H)\beta N h = (1/2+H)\, Nh$$

So if we choose

$$\beta = \frac{1/2+H}{1/2-H}, \tag{6.18}$$

then one can note that if $H \geq 0$, then $\beta \geq 1$ and vice versa. For now, we can just consider the case where H is positive. If H is negative, just consider $N = \lceil \frac{1}{\beta} M \rceil$

So with the choice of β as in (6.18) one has:

$$\begin{aligned} Tc_H\frac{e^{-(1/2-H)\lceil \beta N\rceil h}}{(1/2-H)} + c_H\frac{e^{-(1/2+H)Nh}}{(1/2+H)} &\leq Tc_H\frac{e^{-(1/2-H)\beta Nh}}{(1/2-H)} + c_H\frac{e^{-(1/2+H)Nh}}{(1/2+H)} \\ &= c_H\left(\frac{T}{1/2-H} + \frac{1}{1/2+H}\right)e^{-(1/2+H)Nh} \end{aligned}$$

Now, we can rewrite the hyperbolic sine as:

$$\forall x \in \mathbb{R}:\ \sinh(x) = \frac{1-e^{-2x}}{2e^{-x}} \ \Rightarrow\ (\sinh(\pi d/h))^{-1} = \frac{2e^{-\pi d/h}}{1-e^{-2\pi d/h}} \tag{6.19}$$

Then we can bound the $\mathcal{L}^1([0,T])$ norm of I in (6.15) as:

$$\begin{aligned} \int_0^T I\, d\tau &= \frac{c_H T^{H+1/2}(\cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{(H+1/2)\sinh(\pi d/h)} \\ &\leq \frac{c_H 2T^{H+1/2}(\cos d)^{H-1/2}\Gamma(1/2-H)}{(H+1/2)(1-e^{-2\pi d/h})}e^{-2\pi d/h} \end{aligned}$$

then, set equal convergence rate for all the terms:

$$\frac{2\pi d}{h} = (1/2+H)\, Nh \ \Rightarrow h = \sqrt{\frac{2\pi d}{(1/2+H)N}},$$

so that we have

$$(1/2+H)\, Nh = \sqrt{2\pi d(1/2+H)N}$$

Now, we have K nodes where, with the choice of beta as in (6.18):

$$\begin{aligned} &K = N + M + 1 = N + \lceil \beta N \rceil + 1 \leq N + \lceil \beta \rceil N + N \Rightarrow \\ &N \geq \frac{K}{\lceil \beta \rceil + 2} = K\frac{1/2-H}{3/2-H} \geq K(1/2-H) \end{aligned}$$

and using the fact that $N \geq K(1/2-H)$ we get:

$$\sqrt{2\pi d(1/2+H)N} \geq \sqrt{\pi d(1/2-2H^2)K}$$

The final estimate on the $\mathcal{L}^1([0,T])$ error reads then:

$$c_H\left(\frac{2T^{H+1/2}(\cos d)^{H-1/2}\Gamma(1/2-H)}{(H+1/2)(1-e^{-2\pi d/h})} + \frac{T}{1/2-H} + \frac{1}{1/2+H}\right)e^{-\sqrt{\pi d(1/2-2H^2)K}} \tag{6.20}$$

6.4. Computation of the Lp error

In this section, let us consider $p > 1$, $H \in (-1/2, 1/2)$. In particular, since the kernel $G(\tau)$ is a power-law of exponent $H - 1/2$, to guarantee integrability, we need to restrain p to be $p < \frac{1}{1/2-H}$. So in particular, for negative H, we have that $p < \frac{1}{1/2-H} < 2$.

With this choice, let us compute the $\mathcal{L}^p([0,T])$ norm of the error that we get from approximating our kernel with the SINC quadrature, reminding first that:

$$f_\tau(x) = C_H e^{-\tau e^x + \gamma x}, \quad \tau \in (0,T], \quad \gamma = 1/2 - H$$

Where c_H is as in (6.9)

$$\begin{aligned}
&\left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{K=-M}^{N} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \\
&= \left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-\infty}^{\infty} f_\tau(kh) + h \sum_{k=M+1}^{\infty} f_\tau(-kh) + h \sum_{k=N+1}^{\infty} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])}
\end{aligned}$$

using triangular inequality for $\mathcal{L}^p$ norms:

$$\leq \left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-\infty}^{\infty} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} + \left\| h \sum_{k=M+1}^{\infty} f_\tau(-kh) \right\|_{\mathcal{L}^p([0,T])} + \left\| h \sum_{k=N+1}^{\infty} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])}$$

Define now, for clarity:

$$A := \left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-\infty}^{\infty} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \tag{6.21}$$

$$B := \left\| h \sum_{k=M+1}^{\infty} f_\tau(-kh) \right\|_{\mathcal{L}^p([0,T])} \tag{6.22}$$

$$C := \left\| h \sum_{k=N+1}^{\infty} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \tag{6.23}$$

Let's evaluate first the left and right tail of the series, (6.22) and (6.23). Using the continuity of the $\mathcal{L}^p([0,T])$ norm:

$$\begin{aligned}
&\left\| h \sum_{k=M+1}^{\infty} f_\tau(-kh) \right\|_{\mathcal{L}^p([0,T])} + \left\| h \sum_{k=N+1}^{\infty} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \\
&= h \sum_{k=M+1}^{\infty} \|f_\tau(-kh)\|_{\mathcal{L}^p([0,T])} + h \sum_{k=N+1}^{\infty} \|f_\tau(kh)\|_{\mathcal{L}^p([0,T])} \\
&= c_H h \sum_{k=M+1}^{\infty} \left(\int_0^T e^{-p\tau e^{-kh} - p\gamma kh} d\tau \right)^{1/p} + c_H h \sum_{k=N+1}^{\infty} \left(\int_0^T e^{-p\tau e^{kh} + p\gamma kh} d\tau \right)^{1/p} \\
&= c_H h \sum_{k=M+1}^{\infty} e^{-\gamma kh} \left(\int_0^T e^{-p\tau e^{-kh}} d\tau \right)^{1/p} + c_H h \sum_{k=N+1}^{\infty} e^{\gamma kh} \left(\int_0^T e^{-p\tau e^{kh}} d\tau \right)^{1/p} \\
&= c_H h \sum_{k=M+1}^{\infty} e^{-\gamma kh} \left(\frac{e^{kh}}{p} \left(1 - e^{-pTe^{-kh}} \right) \right)^{1/p} + c_H h \sum_{k=N+1}^{\infty} e^{\gamma kh} \left(\frac{e^{-kh}}{p} \left(1 - e^{-pTe^{kh}} \right) \right)^{1/p}
\end{aligned} \tag{6.24}$$

Using now the fact that $1 - e^{-x} \leq x$ in the first term of (6.24) and $1 - e^{-x} \leq 1$ for the second term in

(6.24) one has:

$$\leq c_H h \sum_{k=M+1}^{\infty} e^{-\gamma kh} \left(\frac{e^{kh}}{p} pTe^{-kh} \right)^{1/p} + c_H h \sum_{k=N+1}^{\infty} e^{\gamma kh} \left(\frac{e^{-kh}}{p} \right)^{1/p}$$
$$= T^{\frac{1}{p}} c_H h \sum_{k=M+1}^{\infty} e^{-\gamma kh} + p^{-\frac{1}{p}} c_H h \sum_{k=N+1}^{\infty} e^{-(1/p-\gamma)kh}$$
$$= T^{\frac{1}{p}} c_H h \frac{e^{-\gamma Mh}}{1-e^{-\gamma h}} + p^{-\frac{1}{p}} c_H h \frac{e^{-(1/p-\gamma)Nh}}{1-e^{-(1/p-\gamma)h}}$$

Now, the first term converges, since $\gamma > 0$, as for the second term, we need to impose: $\gamma - 1/p < 0$ i.e. $p < 1/\gamma = \frac{1}{1/2-H}$. As for the term (6.21), thanks to theorem 7, we have an explicit expression, that is the one of (6.13) so that:

$$A := \left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{k=-\infty}^{\infty} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \leq \left\| \frac{c_H (\tau \cos d)^{H-1/2} \Gamma(1/2-H) e^{-\pi d/h}}{\sinh(\pi d/h)} \right\|_{\mathcal{L}^p([0,T])}$$
$$\leq \frac{c_H (\cos d)^{H-1/2} \Gamma(1/2-H) e^{-\pi d/h}}{\sinh(\pi d/h)} \frac{T^{H-1/2+1/p}}{(p(H-1/2)+1)^{1/p}}$$

As we did before, we can rewrite the Hyperbolic sine with (6.19):

$$= \frac{c_H (\cos d)^{H-1/2} \Gamma(1/2-H) e^{-\pi d/h}}{\sinh(\pi d/h)} \frac{T^{H-1/2+1/p}}{(p(H-1/2)+1)^{1/p}}$$
$$\leq \frac{2c_H (\cos d)^{H-1/2} T^{H-1/2+1/p} \Gamma(1/2-H)}{\left(1-e^{-2\pi d/h}\right)(p(H-1/2)+1)^{1/p}} e^{-2\pi d/h}$$

Putting together all the estimates for A, B, C we obtain:

$$\left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{K=-M}^{N} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \leq A + B + C$$
$$\leq \frac{2c_H (\cos d)^{H-1/2} T^{H-1/2+1/p} \Gamma(1/2-H)}{\left(1-e^{-\pi d/h}\right)(p(H-1/2)+1)^{1/p}} e^{-2\pi d/h} + T^{\frac{1}{p}} c_H h \frac{e^{-(1/2-H)Mh}}{1-e^{-(1/2-H)h}} + p^{-\frac{1}{p}} c_H h \frac{e^{-(1/p-(1/2-H))Nh}}{1-e^{-(1/p-(1/2-H))h}}$$

Now, we have three convergence rate for A, B, C i.e.:

$$A \propto e^{-2\pi d/h}$$
$$B \propto e^{-(1/2-H)Mh}$$
$$C \propto e^{-(1/p-1/2+H)Nh}$$

Since our benchmark will be the error in function of the total number of nodes, let's say that there are in our quadrature rule exactly K points. So the relation will be:

$$K = M + N + 1 \Rightarrow M = K - N - 1 \Rightarrow$$
$$B \leq T^{\frac{1}{p}} c_H h \frac{e^{-(1/2-H)(K-N-1)h}}{1-e^{-(1/2-H)h}} = T^{\frac{1}{p}} c_H h e^{(1/2-H)h} \frac{e^{-(1/2-H)(K-N)h}}{1-e^{-(1/2-H)h}}$$

Now, we can equate all the exponents, so that we have the same rate of convergence for all the terms:

$$2\pi d/h = (1/2-H)(K-N)h, \quad (1/2-H)(K-N)h = (1/p-1/2+H)Nh$$

$$h^2 = \frac{2\pi d}{(1/2-H)(K-N)}, \quad p(1/2-H)K = N \tag{6.25}$$

For the last equation to have sense, we need to impose that $p(1/2-H) \leq 1$, i.e.: $p \leq 1/(1/2-H)$, which is exactly what we asked p to be at the beginning of this section. If we now substitute h into the exponent $2\pi d/h$ we obtain:

$$2\pi d/h = \sqrt{2\pi d(1/2-H)\left(1-p\left(1/2-H\right)\right)K} \tag{6.26}$$

This is accomplished, with step size:

$$\begin{aligned} h &= \sqrt{\frac{2\pi d}{(1/2-H)(1-p(1/2-H))K}} \\ N &= \lceil (K(1/2-H)p) \rceil \\ M &= \lceil (K-N-1) \rceil \end{aligned}$$

Where with $\lceil \cdot \rceil$ we denote the ceil function, that takes a real integer and approximate it to the closest and nearest integer, then if we put all the pieces back together:

$$\begin{aligned} &\left\| \int_{\mathbb{R}} f_\tau(x)dx - h\sum_{k=-M}^{N} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \\ &\leq c_H \left(\frac{2(\cos d)^{H-1/2} T^{H-1/2+1/p}\Gamma(1/2-H)}{\left(1-e^{-2\pi d/h}\right)\left(p(H-1/2)+1\right)^{1/p}} + \frac{T^{\frac{1}{p}} h e^{(1/2-H)h}}{1-e^{-(1/2-H)h}} + \frac{p^{-\frac{1}{p}} h}{1-e^{-(1/p-(1/2-H))h}} \right) \\ &\times e^{-\sqrt{2\pi d(1/2-H)(1-p(1/2-H))K}} \end{aligned}$$

We note here that, this results coincides with the result previously found for $p = 1$. So, until now, we have shown the following theorem:

Theorem 8. For a fixed $T > 0$ and $H \in (-1/2, 1/2)$ and $1 \leq p < \frac{1}{1/2-H}$ and for a given $d \in (0, \pi/2)$. Then for every $K \in \mathbb{N}$ set:

$$h = \sqrt{\frac{2\pi d}{(1/2-H)(1-p(1/2-H))K}} \tag{6.27}$$

$$N = \lceil (K(1/2-H)p) \rceil \tag{6.28}$$

$$M = K - N - 1 \tag{6.29}$$

Let $(x_i, w_i)_{i=1}^K$ the nodes and weights of the SINC quadrature rule given by:

$$x_i = e^{(i-M)h}, \quad w_i = hc_H e^{(1/2-H)(i-M)}, \quad i = 0, \ldots, M+N$$

where rd denotes the function that rounds every real number to its nearest integer, and c_H the constant defined as in (6.9). Then one has, for this choice of nodes and weights the following error bound:

$$\begin{aligned} &\left\| \frac{\tau^{H-1/2}}{\Gamma(H+1/2)} - \sum_{k=-M}^{N} w_i e^{-\tau x_i} \right\|_{\mathcal{L}^p([0,T])} \\ &\leq c_H \left(\frac{2(\cos d)^{H-1/2} T^{H-1/2+1/p}\Gamma(1/2-H)}{\left(1-e^{-2\pi d/h}\right)\left(p(H-1/2)+1\right)^{1/p}} + \frac{T^{\frac{1}{p}} h e^{(1/2-H)h}}{1-e^{-(1/2-H)h}} + \frac{p^{-\frac{1}{p}} h}{1-e^{-(1/p-(1/2-H))h}} \right) \\ &\times e^{-\sqrt{2\pi d(1/2-H)(1-p(1/2-H))K}} \end{aligned} \tag{6.30}$$

Another path that one can take, instead of bounding the Hyperbolic sin with (6.19) and obtaining:

$$\begin{aligned} &= \frac{c_H(\cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{\sinh(\pi d/h)} \frac{T^{H-1/2+1/p}}{\left(p(H-1/2)+1\right)^{1/p}} \\ &\leq \frac{c_H(\cos d)^{H-1/2} T^{H-1/2+1/p}\Gamma(1/2-H)}{\left(1-e^{-2\pi d/h}\right)\left(p(H-1/2)+1\right)^{1/p}} e^{-2\pi d/h} \end{aligned}$$

one, can also use the following inequality [3]:

$$\sinh(z) > z\cosh^{4/3}(z/2) > ze^{2/3z}, \quad \forall z \in (0,\infty).$$

Then, one obtains:

$$\begin{aligned}
&= \frac{c_H(\cos d)^{H-1/2}\Gamma(1/2-H)e^{-\pi d/h}}{\sinh(\pi d/h)} \frac{T^{H-1/2+1/p}}{(p(H-1/2)+1)^{1/p}} \\
&\leq \frac{hc_H T^{H-1/2+1/p}(\cos d)^{H-1/2}\Gamma(1/2-H)}{\pi d\,(p(H-1/2)+1)^{1/p}} e^{-\pi\frac{5}{3}d/h}
\end{aligned}$$

Putting together all the estimates for A, B, C we obtain:

$$\begin{aligned}
&\left\| \int_{\mathbb{R}} f_\tau(x)dx - h\sum_{K=-M}^{N} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \leq A+B+C \\
&\leq \frac{hc_H T^{H-1/2+1/p}(\cos d)^{H-1/2}\Gamma(1/2-H)}{\pi d\,(p(H-1/2)+1)^{1/p}} e^{-\pi\frac{5}{3}d/h} + T^{\frac{1}{p}}c_H h\frac{e^{-(1/2-H)Mh}}{1-e^{-(1/2-H)h}} + p^{-\frac{1}{p}}c_H h\frac{e^{-(1/p-(1/2-H))Nh}}{1-e^{-(1/p-(1/2-H))h}}
\end{aligned}$$

Remark. Note that in the bounds that we have just given, we have that only one of the two tail has a dependency on time. So this means that for smaller T, we will need a lower number for M

Now, we have three convergence rate for A, B, C i.e.:

$$\begin{aligned}
A &\propto e^{-\pi\frac{5}{3}d/h} \\
B &\propto e^{-(1/2-H)Mh} \\
C &\propto e^{-(1/p-1/2+H)Nh}
\end{aligned}$$

doing the same reasoning as before we have:

$$\begin{aligned}
&K = M+N+1 \Rightarrow M = K-N-1 \Rightarrow \\
&B \leq T^{\frac{1}{p}}c_H h\frac{e^{-(1/2-H)(K-N-1)h}}{1-e^{-(1/2-H)h}} = T^{\frac{1}{p}}c_H h e^{(1/2-H)h}\frac{e^{-(1/2-H)(K-N)h}}{1-e^{-(1/2-H)h}}
\end{aligned}$$

Now, we can equate all the exponents, so that we have the same rate of convergence for all the terms:

$$\frac{5}{3}\pi d/h = (1/2-H)(K-N)h, \quad (1/2-H)(K-N)h = (1/p-1/2+H)Nh$$

$$h^2 = \frac{5\pi d}{3(1/2-H)(K-N)}, \quad p(1/2-H)K = N$$

For the last equation to have sense, we need to impose that $p(1/2-H) \leq 1$, i.e.: $p \leq 1/(1/2-H)$, which is exactly what we asked p to be at the beginning of this section. If we now substitute h into the exponent $5\pi d/3h$ we obtain:

$$\frac{5}{3}\pi d/h = \sqrt{\frac{5}{3}\pi d(1/2-H)\left(1-p\left(1/2-H\right)\right)K} \tag{6.31}$$

This is accomplished, with step size:

$$\begin{aligned}
h &= \sqrt{\frac{5\pi d}{3(1/2-H)(1-p(1/2-H))K}} \\
N &= \lceil (K(1/2-H)p) \rceil \\
M &= \lceil (K-N-1) \rceil
\end{aligned}$$

then if we put all the pieces back together, setting $\gamma = (1/2 - H)$:

$$\begin{aligned}
&\left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{K=-M}^{N} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \\
&\leq c_H \left(\frac{hc_H T^{\gamma+1/p} (\cos d)^\gamma \Gamma(\gamma)}{\pi d \left(p\gamma + 1\right)^{1/p}} + \frac{T^{\frac{1}{p}} h e^{\gamma h}}{1 - e^{-\gamma h}} + \frac{p^{-\frac{1}{p}} h}{1 - e^{-(1/p - \gamma)h}} \right) \\
&\times e^{-\sqrt{\frac{5}{3}\pi d\gamma(1-p\gamma)K}}
\end{aligned} \tag{6.32}$$

In the next section, we are going to compare the two different quadrature rules that we have given here in (6.32) and in (6.30)

7

Numerics

In this section, first we are going to present two algorithms that we have developed to obtain the nodes and weights of the SINC quadrature rule, secondly we are going to present the numerical experiments that we have performed. In particular, we will evaluate the goodness of the SINC quadrature rule, via different metrics, inspired by the paper [8]. In their paper, they evaluate different quadrature rules via different methods:

- For different Hurst parameters H, plot the relative $\mathcal{L}^p\left([0,T]\right), p = 1,2$ error between the real kernel and its approximation against the number of quadrature points.
- Numerically verify that the theoretical convergence rate that we found is in line with the one that we found in chapter 6.
- Compare the computational time to get the quadrature nodes of the SINC compared with the ones reported by the various paper that we analysed in the literature review.
- Compare the largest node between different quadrature rules. As remarked in [8], having nodes that are small, improves the numerical stability of the pricing method and lower computational times.
- Compute implied volatility smiles for European call options under the Rough Heston model for different values of the Hurst parameter and for different maturities and compare it with the other quadrature rules presented in chapter 5.
- Compute Implied Volatility surfaces under the Rough Heston model and under its Markovian approximation, using different quadrature rules and comparing the results obtained.
- Compute prices for continuously monitored Geometric Asian options under the Rough Heston model.

7.1. Algorithms

In this section, we will provide two algorithms for optimally choosing the step-size h used in the SINC quadrature rule.

For a fixed Hurst parameter $H \in (-1/2, 1/2)$, let's fix a $p \in [1, 1/(1/2-H))$. Say that we want K nodes in the quadrature rule, then, we have seen in the previous section that if we choose

$$\begin{aligned} N &= p(1/2-H)K \\ M &= K - N - 1 \\ h &= \sqrt{\frac{2\pi d}{(1/2-H)(1-p(1/2-H))K}} \end{aligned} \tag{7.1}$$

for a given $d \in (0, \pi/2)$, then one has the error bound for the $\mathcal{L}^p([0,T])$ error

$$c_H \left(\frac{(\cos d)^{H-1/2} T^{H-1/2+1/p} \Gamma(1/2-H)}{\left(1-e^{-2\pi d/h}\right)\left(p(H-1/2)+1\right)^{1/p}} + \frac{T^{\frac{1}{p}} h e^{\gamma h}}{1-e^{-\gamma h}} + \frac{p^{-\frac{1}{p}} h}{1-e^{-(1/p-\gamma)h}} \right) \times e^{-\sqrt{2\pi d(1/2-H)(1-p(1/2-H))K}} \tag{7.2}$$

However, d remains a parameter that needs to be selected. We have developed an algorithm to optimally choose d:

1. For a fixed total number of nodes K, we computed M, N by setting:
$$N = \lceil p(1/2-H)K \rceil, \quad M = K - N - 1$$
2. For h as in (7.1), minimise expression (7.2), with respect to d, M, N as a constrained optimisation problem with:
$$N \in [0, \lceil p(1/2-H)K \rceil], \quad M \in [0, K-N-1], \quad d \in (0, \pi/2)$$
to do this, we used Scipy's `minimise` function for constrained optimisation.
3. Recompute the step size h with the optimised value of d using (7.1)
4. Finally, compute the nodes and the weights $(x_i, w_i)_{i=1}^K$ of the quadrature:
$$x_k = e^{(k-M)h}, \quad w_k = c_H h e^{(1/2-H)(k-M)h}, \quad \forall k = 0, \dots, M+N \tag{7.3}$$
with c_H as in (6.9)

We have observed that for small values of T, our algorithm does not perform well. Specifically, the constrained optimization tends to select the largest possible values for M. To address this issue, we propose a new algorithm that provides exactly K nodes and weights for the SINC quadrature while accounting for small T. Instead of selecting the step size h using a fixed rule as in (7.1), we leave this task to an optimization procedure. The previous algorithm appeared to be almost "time-independent," with the dependence on the final time T hidden within the function being optimized. The new algorithm aims at selecting the step size h by explicitly considering the time T. We will now describe the developed algorithm in detail, which utilizes the error bound we have previously established:

$$\left\| \int_{\mathbb{R}} f_\tau(x)dx - h \sum_{K=-M}^{N} f_\tau(kh) \right\|_{\mathcal{L}^p([0,T])} \leq A + B + C \tag{7.4}$$

$$\leq \frac{2c_H (\cos d)^{H-1/2} T^{H-1/2+1/p} \Gamma(1/2-H)}{\left(1-e^{-\pi d/h}\right)\left(p(H-1/2)+1\right)^{1/p}} e^{-2\pi d/h} + T^{\frac{1}{p}} c_H h \frac{e^{-(1/2-H)Mh}}{1-e^{-(1/2-H)h}} + p^{-\frac{1}{p}} c_H h \frac{e^{-(1/p-(1/2-H))Nh}}{1-e^{-(1/p-(1/2-H))h}} \tag{7.5}$$

Where

$$f_\tau(x) = c_H e^{-\tau e^x + (1/2-H)x}$$

For a given starting tolerance, say $\varepsilon_0 = 0.1$ and for $q > 1$, say that we want to find K nodes.

1. Optimize expression A in (7.4) with respect to h, d, until it reaches tolerance $\varepsilon_0/3$.
2. For the optimised parameters h, d, find the least integer M such that expression B in (7.4) is less or equal than $\varepsilon_0/3$.
3. Now, with the value of h found in the first step, compute the least integer $N \in \{0, \dots, M-K-1\}$ such that C in (7.4) is less than $\varepsilon_0/3$.
4. If the values of M, N are such that $M+N+1 = K$ then the algorithm stops. Otherwise, reassign to the hyper parameter ε_0 $\varepsilon_0 = \varepsilon_0/q$ and repeat from step 1.

Note that this algorithm will terminate. In fact, one can note that both B, C in (7.4) are decreasing with respect to M, N respectively. Also, choosing $N \in \{0, \dots, M - K - 1\}$ ensures that the algorithm does not pick values for M, N such that $M + N + 1 > K$. An important note on the first step of this algorithm is the following: we want to choose h, d such that A in (7.4) is less than $\varepsilon_0/3$. However, A tends to zero if h tends to zero, so our task is to choose the biggest h such that $A < \varepsilon_0/3$. In this way, we can achieve the same accuracy of each term A, B, C in expression (7.4). In our implementation, we chose to solve this problem in the following way:

- fix $h_0 > 0$, $d \in (0, \pi/2)$. If there is a couple (h, d) in $(h_0, \infty) \times (0, \pi/2)$ such that $A < \varepsilon_0/3$ then the algorithm terminates.
- if there is no values for (h, d) such that $A < \varepsilon_0/3$ then decrease h_0 by a fixed factor and repeat step 1 until termination.

Note that this algorithm is guaranteed to come to a stop since $\lim_{h \to 0} A = 0$. The second algorithm has two more hyper parameters that need to be optimised, id est the starting tolerance ε_0 and q. We noticed in fact that ε_0 influences the convergence rate and has to be chosen accordingly to the total number of nodes present in the quadrature rule. In fact, we empirically observed that for a low number of nodes, it is necessary for ε_0 not to be too small, because if this is the case, then the algorithm will choose a value for the step size h that is too low to compensate for the error given by the tails of the integral B, C in (7.4). In fact we see that if the step size is too small, then the exponentials in the terms B, C will have a low rate of convergence in the number of nodes M, N. So, after a lot of trial and error, we noted that for small T, in particular for $T < 0.1$ algorithm 2 would outperform Algorithm 1, with the following choice of ε_0, q:

$$\varepsilon_0 = \frac{2\sqrt{T}}{K}, \quad q = 1.1$$

Where K is the total number of nodes. With this choice, we can improve the $\mathcal{L}^1([0.T])$ errors by almost a factor of 2 in the asymptotic regime. As for $T > 0.1$, we noticed that the first algorithm tends to outperform algorithm 1 for a number of nodes $K > 10$. To support our thesis, let us plot in Figure 7.1-7.2 the $\mathcal{L}^1([0, T])$ errors between the kernel and its approximation for different values of T and Hurst parameter H.

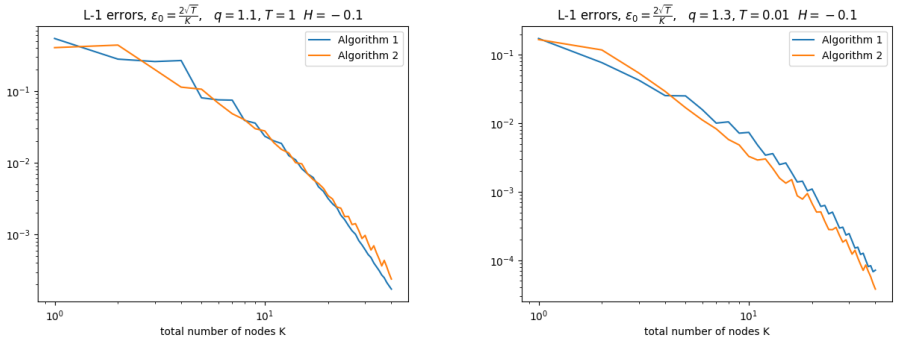

Figure 7.1: $\mathcal{L}^1([0, T])$ errors for algorithm 1 and algorithm 2 for different values of T, $H = -0.1$

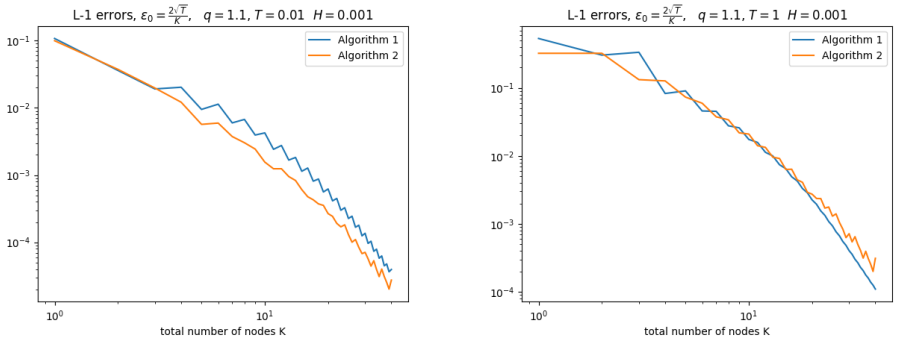

Figure 7.2: $\mathcal{L}^1([0, T])$ errors for algorithm 1 and algorithm 2 for different values of T, $H = 0.001$

So, in light of this observations, we decided to proceed in the following way: to use algorithm 2 for computing implied volatility smiles with $T < 0.1$. Instead, for computing all the other implied volatility smiles and implied volatility surfaces we will use algorithm 1.

7.1.1. Node in Zero

Now, in this subsection, we are going to briefly present how we could use an idea given in [6] to improve the SINC quadrature rule.

Recall that, for the $\mathcal{L}^2([0,T])$ error, we have an analytic expression, i.e.:

$$\begin{aligned}\int_0^T \left|\hat{g}_N(t) - g(t)\right|^2 dt =& \frac{T^{2H}}{2H\Gamma(H+1/2)^2} + w_0^2 T \\ &+ 2w_0 \sum_{i=1}^{K} \frac{w_i}{x_i}\left(1 - e^{-x_i T}\right) \\ &+ \sum_{i,j=1}^{K} \frac{w_i w_i}{x_i + x_j}\left(1 - e^{-(x_i+x_j)T}\right) \\ &- \frac{2w_0 T^{H+1/2}}{\Gamma(H+3/2)} - \frac{2}{\Gamma(H+1/2)} \\ &\times \sum_{i=1}^{K} \frac{w_i}{x_i^{H+1/2}} \int_0^{x_i T} t^{H-1/2} e^{-t} dt\end{aligned} \tag{7.6}$$

where

$$g(t) = \frac{t^{H-1/2}}{\Gamma(H+1/2)}$$

$$\hat{g}(t) = w_0 + \sum_{i=1}^{K} w_i e^{-x_i t}, \quad x_i \in (0,\infty), w_i \in \mathbb{R}, \forall i = 1,\dots,K$$

then, one realises that the expression in 7.6 is a polynomial of second degree in w_0 and then one can also easily optimise equation (7.6) expression with respect to w_0, by setting:

$$w_0 = \frac{1}{T}\left(\frac{T^{H+1/2}}{\Gamma(H+3/2)} - \sum_{i=1}^{N} \frac{w_i}{x_i}\left(1 - e^{-x_i T}\right)\right) \tag{7.7}$$

The only problem of this, is that we cannot give an estimate of the error of the SINC quadrature with the additional node and weight $(0, w_0)$, since the nodes of the SINC are expressed via an exponential as one can see in (7.3), they are always strictly positive. However, by adding this node in zero, one can boost even more the performance of the SINC quadrature, in particular for $H \in (0, 1/2)$

Now, instead we plot the $\mathcal{L}^p([0,T])$ relative error for the SINC quadrature, with and without the node in zero, as specified in 7.7

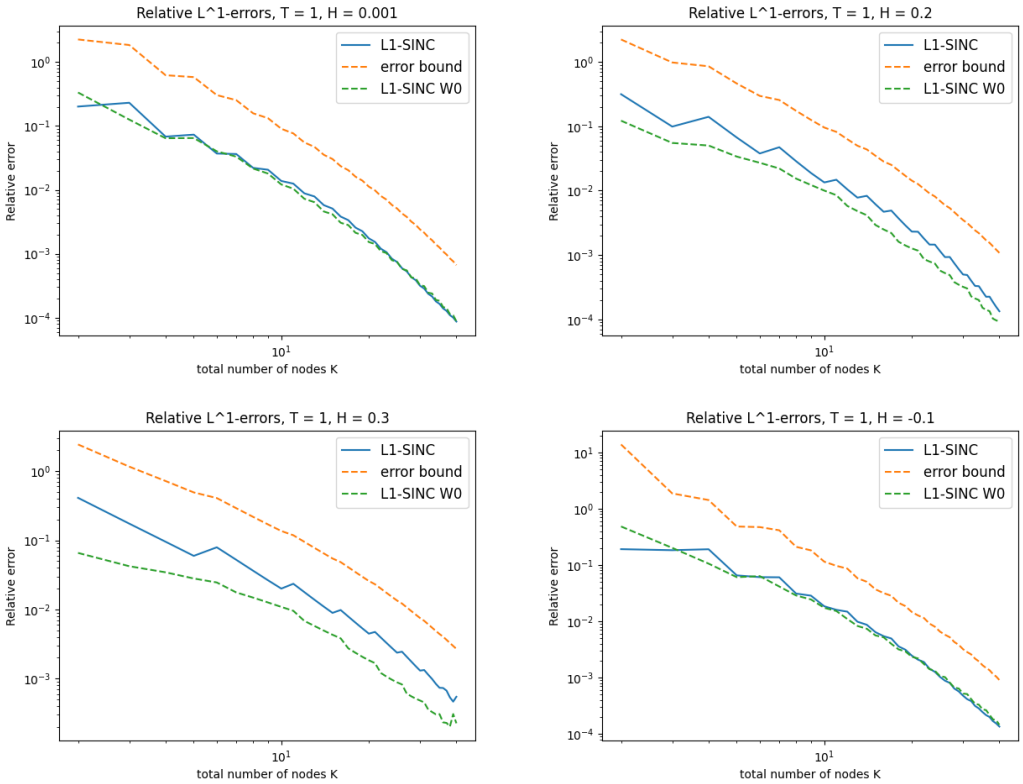

Figure 7.3: Relative $\mathcal{L}^1([0,T])$ errors for different values of the Hurst parameter H in function of the number of nodes in the quadrature rule with and without the node in zero

As we can see in the Figure 7.3, for some values of the number of nodes, adding a node in zero in fact yields a better approximation of the kernel. In particular, we noted that for higher values of the Hurst parameter,$H \approx 1/2$, the approximation yields even better results. Instead, for negative values of the Hurst parameter or $H \approx 0$, having or not having the node in zero does not make that big of a difference as we can see from the Figure 7.3.

As for the $\mathcal{L}^2([0,T])$ error we have the following results:

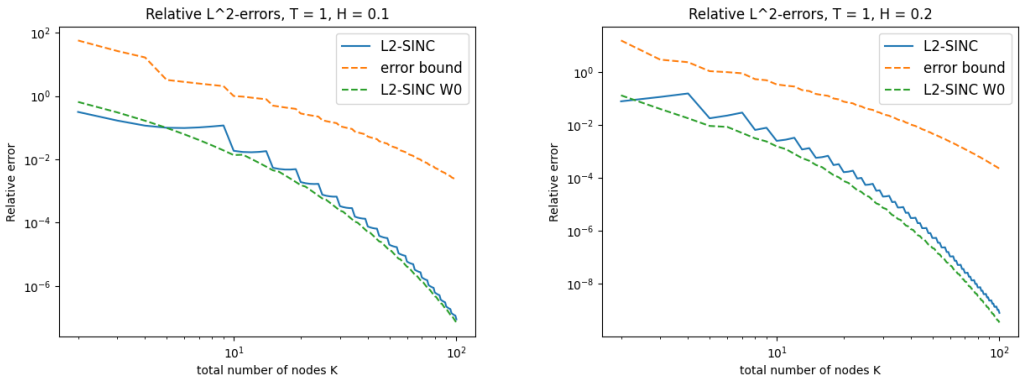

Figure 7.4: Relative $\mathcal{L}^2([0,T])$ errors for different values of the Hurst parameter H in function of the number of nodes in the quadrature rule

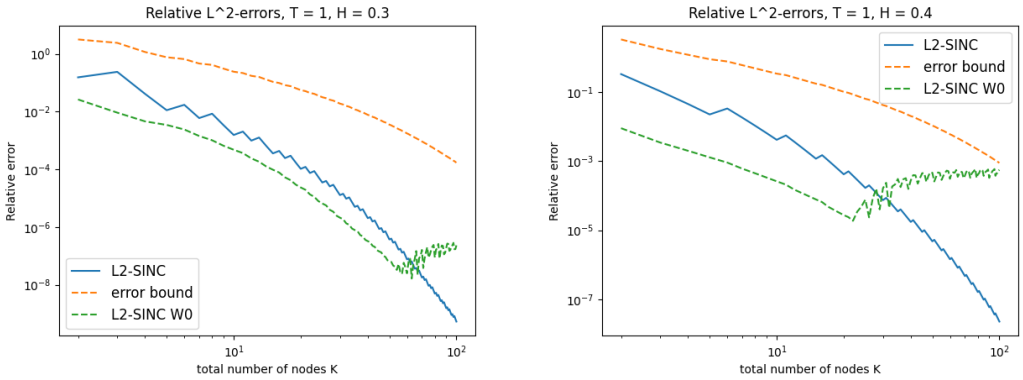

Figure 7.5: Relative $\mathcal{L}^2([0,T])$ errors for different values of the Hurst parameter H in function of the number of nodes in the quadrature rule

The surprising fact here, is the fact that one expects a better result for the $\mathcal{L}^2([0,T])$ error, as we are choosing w_0 by optimizing exactly this quantity, as we can clearly see in 7.4 and 7.5. However, from the figure one can see how an improvement is also made for the $\mathcal{L}^1([0,T])$ error. As we will see, adding a node in zero, gives an impressive boost on the performances of the SINC quadrature, both regarding the strong and the weak error.

7.2. Empirical Numerical Convergence

In this section, we present the empirical numerical convergence of our scheme. We will plot both the $\mathcal{L}^1([0,T])$ and the $\mathcal{L}^2([0,T])$ errors as functions of the total number of nodes.

To compute the $\mathcal{L}^2([0,T])$ error, we used Proposition 2.12 from Bayer [7]. This proposition provides a straightforward method to compute $\mathcal{L}^2([0,T])$ errors given the nodes and weights of the quadrature.

For the $\mathcal{L}^1([0,T])$ error, we used the method described in Appendix E of [8], which we have outilined in Section 5.3.4.

Let us plot in Figure 7.6 and 7.7 first the relative error with respect to the $\mathcal{L}^p([0,T])$ the norm, for $p = 1, 2$ and the error against the theoretical convergence rates found in (7.2). When we say relative error it means that we consider the following quantities: the theoretical $\mathcal{L}^p([0,T])$ norm for the kernel:

$$\|g\|^p_{\mathcal{L}^p([0,T])} = \int_0^T |g(s)|^p \, ds = \frac{T^{((H-1/2)p+1)}}{\Gamma(H+0.5)((H-1/2)p+1)}$$

and for relative error we mean:

$$e_N = \frac{\|\hat{g}_N - g\|_{\mathcal{L}^p([0,T])}}{\|g\|_{\mathcal{L}^p([0,T])}}$$

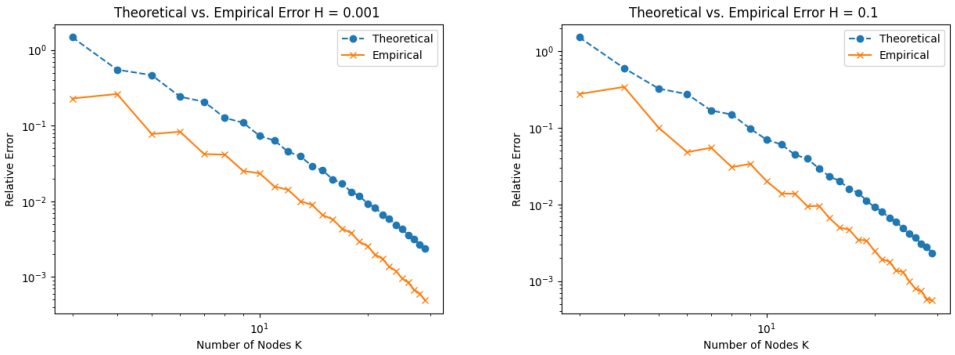

Figure 7.6: In dotted lines, is plotted expression (7.2), for different values of H, that is, our theoretical error bound, the other line represents the actual $\mathcal{L}^1([0,T])$ error

Note here two things, that the precision of our approximation is higher for values of the Hurst parameters H that are close to zero, while deteriorates progressively for $H \to \pm 0.5$. Also, we can note how the error bound that we have found is very much in line with the actual error. The situation seems much different for the case $p = 2$.

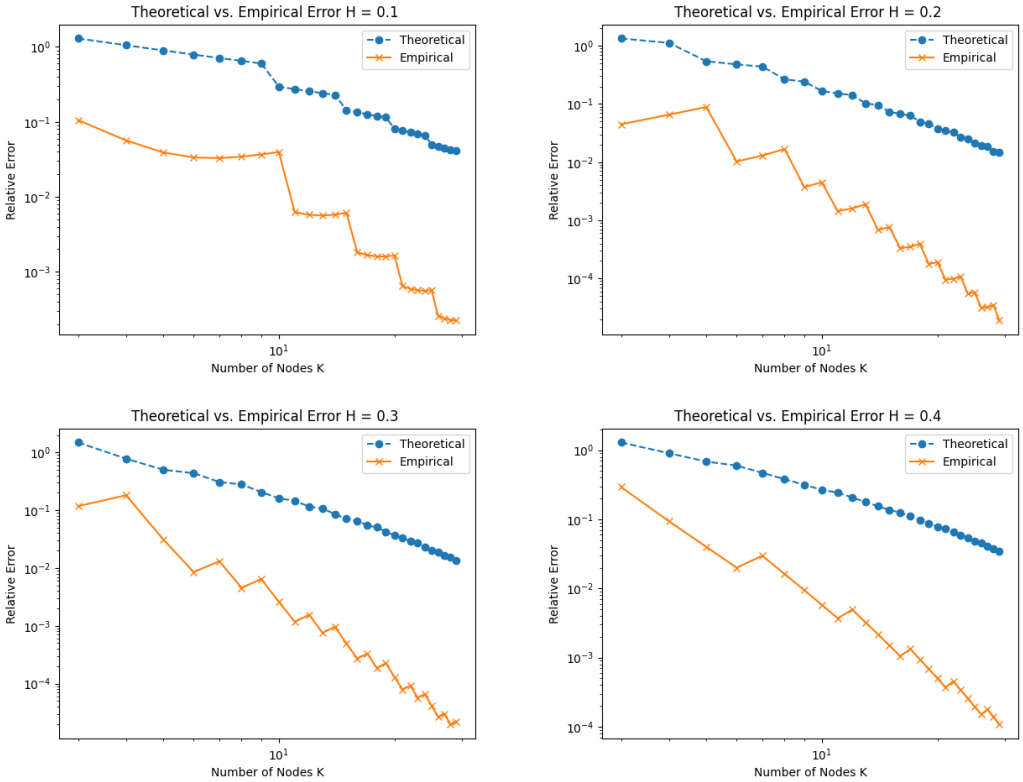

Figure 7.7: In dotted lines, is plotted expression (7.2), for different values of H and $p = 2$, that is, our theoretical error bound

Finally, we can estimate the asymptotic rate for the SINC quadrature. Through several experiments, we observed that as the total number of nodes increases, the optimal choice of d (the height of the strip in the complex plane, as explained in section 6) tends towards $\pi/2$. To support this observation, we plotted the difference between d and $\pi/2$ as a function of the total number of nodes for different values of H.

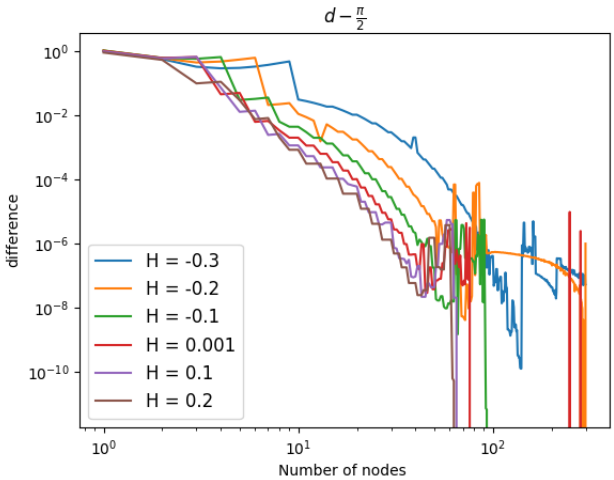

Figure 7.8: Difference between the length d of the strip in the complex plane and $\pi/2$ for the L1-SINC and L1-SINC-W0

The situation is a bit different instead for the L2-SINC, in fact, for values of H that are close to zero, then there is such an evident convergence as for the case of the L1-SINC. In what follows, we plot the same difference for d, but for the L2-SINC algorithm.

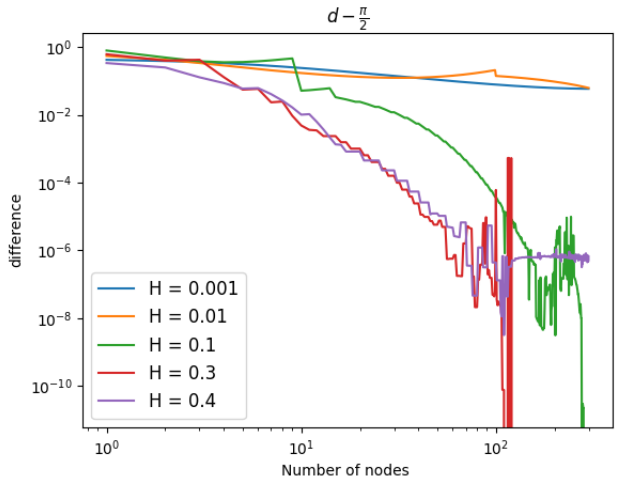

Figure 7.9: Difference between the length d of the strip in the complex plane and $\pi/2$ for the L2-SINC

In this way, we can assert that the asymptotic convergence rate for the SINC quadrature is exponential in the number of nodes used in the quadrature K with exponent:

$$\sqrt{\pi^2(0.5-H)(H+0.5)K}$$

Now that we have shown accordance with the theoretical error bound that we have previously given, we move on to the next section where we will compare the SINC quadrature rule to the state of the art.

7.3. Comparison with state of the art

In this section, we are going to compare our choice of nodes and weights with the state of the art. Specifically, we will first compare the computational time for computing these nodes and the size of the largest node, similar to [8]. Once we have done this, we will compare the convergence rate of our quadrature. Finally, at the end of this section, we will compute option prices in the Rough Heston model via Fourier inversion techniques, and then compare the Implied Volatility smile of the approximated Markovian Rough Volatility model and the original model.

Throughout this section, we will refer to our proposed algorithms with this notation:

- L1-SINC, id est the SINC method, optimised for reducing the $\mathcal{L}^1([0,T])$ error, as in (7.2), with $p=1$.
- L1-SINC-W0, id est the SINC method, optimised for reducing the $\mathcal{L}^1([0,T])$ error, as in (7.2), with $p=1$, with an additional node in 0, as specified before in (7.7).
- L2-SINC, id est the SINC method, optimised for reducing the $\mathcal{L}^2([0,T])$ error, as in (7.2), with $p=2$.
- L2-SINC-W0, id est the SINC method, optimised for reducing the $\mathcal{L}^2([0,T])$ error, as in (7.2), with $p=2$, with an additional node in 0, as specified before in (7.7).

In general, for computing the $\mathcal{L}^p([0,T])$ errors, we are going to use the corresponding Lp-SINC. Moreover, for computing option prices and implied volatility smiles, we are going to use the L1-SINC.

7.3.1. Largest node and computational time

In this section, we present the largest node and the computational times for all the algorithms proposed above.

In Table 7.1, we present the maximum value of the node in $\log_{10}$ scale for different algorithms, different

values of the Hurst parameter, and for fixed $T = 1$. Note that we did not include values for the algorithms Lp-SINC-W0 for $p \in 1, 2$ to avoid redundancies. When we consider K nodes, the first $K - 1$ nodes are identical to the Lp-SINC, and we add a node at zero as described in (7.7). Thus, the maximum node for the Lp-SINC-W0 with K total nodes is exactly equal to the maximum node of the Lp-SINC with $K - 1$ nodes.

It is clear from the Table 7.1 that the L1-SINC algorithm has very small nodes compared even to an algorithm like $BL2$, which is constructed by optimizing the $\mathcal{L}^2([0, T])$ error and penalizing large nodes in favor of smaller ones. In contrast, comparing L1-SINC with an algorithm like AE is not meaningful since the AE algorithm converges at a rate of N^{-H}, meaning that having 10 nodes will result in very low precision. As observed, the L1-SINC algorithm maintains relatively bounded node sizes even for a large number of nodes.

	H = -0.1				H = 0.001					H = 0.1				
K	L1-SINC	GG	NGG	OL1	SINC	GG	NGG	OL1	AK	SINC	GG	NGG	OL1	AK
1	0.00	**0.18**	**0.05**	**0.24**	0.00	**0.12**	0.00	**0.14**	-	0.00	**0.06**	-0.07	**0.02**	-
2	1.24	1.17	0.95	**1.40**	1.37	1.02	0.95	1.28	**3.32**	1.54	0.92	0.95	1.18	1.48
3	2.54	1.59	1.70	2.24	2.66	1.39	1.70	2.07	**3.32**	2.97	1.25	1.70	1.93	1.48
4	3.65	1.94	2.49	2.98	2.46	1.70	**2.49**	**2.75**	**7.10**	2.72	1.58	2.49	2.57	**3.13**
5	3.38	2.24	3.32	**3.66**	3.68	2.02	3.32	3.35	**7.10**	2.26	1.81	1.09	**3.14**	**3.13**
6	4.50	2.57	4.16	4.24	3.05	2.26	1.86	**3.91**	**11.5**	3.40	2.04	1.86	**3.65**	**4.40**
7	5.62	2.82	2.66	4.79	4.06	2.48	2.66	**4.42**	**11.5**	2.95	2.24	2.66	**4.28**	**4.40**
8	4.63	3.04	2.66	**5.32**	3.52	2.68	2.66	**4.90**	**15.5**	3.94	2.42	2.66	**4.58**	**5.45**
9	5.56	3.24	2.66	**6.03**	4.41	2.86	2.66	**5.35**	**15.5**	3.52	2.58	2.66	**5.20**	**5.45**
10	4.82	3.44	3.49	**6.65**	3.94	3.04	3.49	**6.02**	**20.0**	3.21	2.75	3.49	**5.74**	**6.35**

Table 7.1: Size of the largest node for different quadrature rules, expressed in log base. **Bold** values are values that are greater than the one given by the L1-SINC quadrature. Values for the other algorithms are taken from [8, Table 1, Section 4]

As for the computational times, every algorithm we provided runs in at most 20 milliseconds for every total number of nodes K. This is in clear contrast with algorithms like OL1, as given in [7], where the computational time for 10 nodes reaches 10 minutes.

Now that we have shown that our algorithm, particularly L1-SINC and L1-SINC-W0, performs better than the state of the art in terms of the largest node and computational time, we are ready to demonstrate that we also achieve better performance in terms of convergence. This comparison is made at least when considering all algorithms that do not rely on a direct optimization procedure of the $\mathcal{L}^p([0, T])$ error like OL1,OL2 and BL2 proposed in the literature.

To corroborate our findings, we plotted the computational times for different quadrature rules, except OL1,OL2,BL2. What we found is that the SINC quadrature has a computational time that is comparable to all the other quadrature rules, with no significant improvement or loss in the running time:

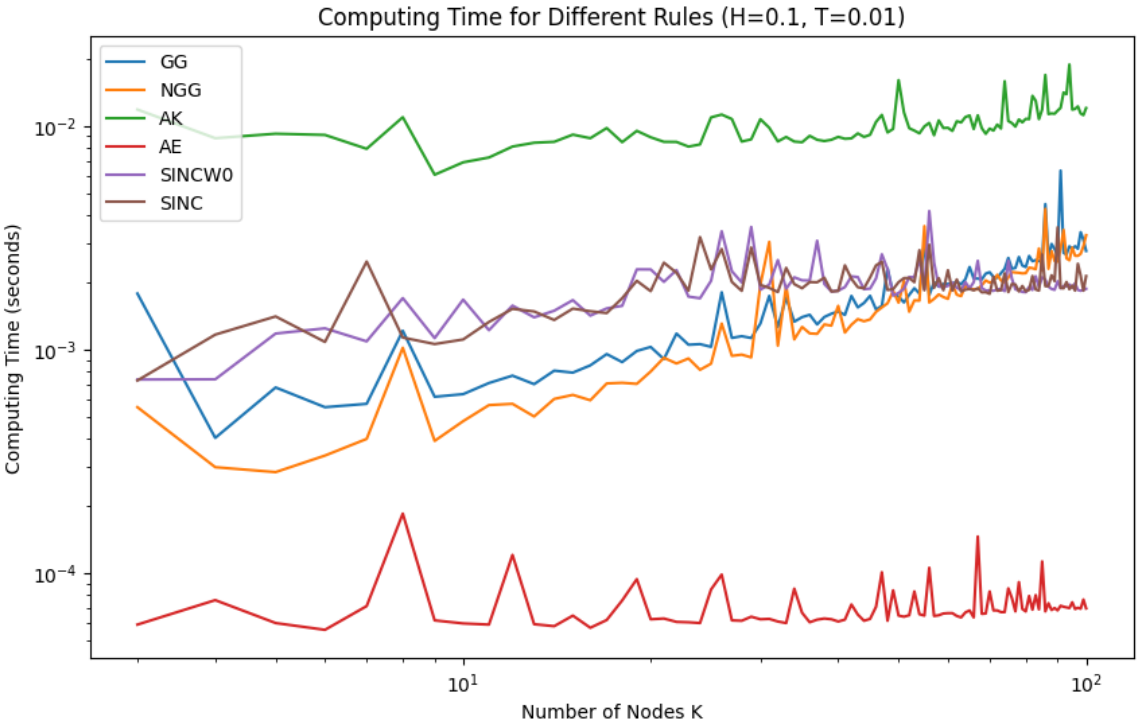

Figure 7.10: Computational time for different quadrature rules

First, we noted that neither the Hurst parameter H, nor the final time T are influential on the run time of every quadrature. Secondly, the only quadrature rule that is much faster than the other ones is the one given by Abi Jaber and El Euch in [1]. However, as we will see in the next section, this quadrature rule performs the worst, meaning that any time saved during computation is subsequently lost due to reduced accuracy.

7.3.2. Convergence rate comparison

In this subsection, we are going to compare different quadrature rules, with the respective convergence rates. First, we plot the errors obtained by using the quadrature rules from Alfonsi and Kebaier [35] that we will denote with 'AK', the one from the initial paper from Abi Jaber et al [34], denoted with 'AE', and the one introduced by Bayer et al in [8],[6], id est both the Non-Geometric Gaussian 'NGG', and Geometric Gaussian 'GG' that we have introduced in chapter 5.

First, we are going to plot the relative $\mathcal{L}^2([0,T])$ errors for different quadrature rules and different values of the Hurst parameter in Figures 7.11-7.12

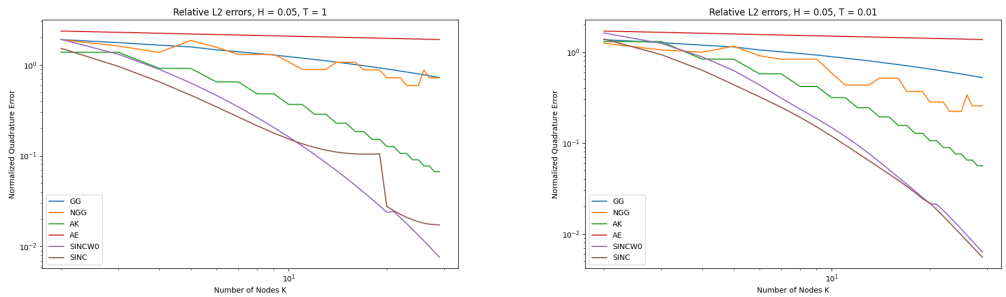

Figure 7.11: Relative $\mathcal{L}^2$ errors for different quadrature rules and different terminal values T.

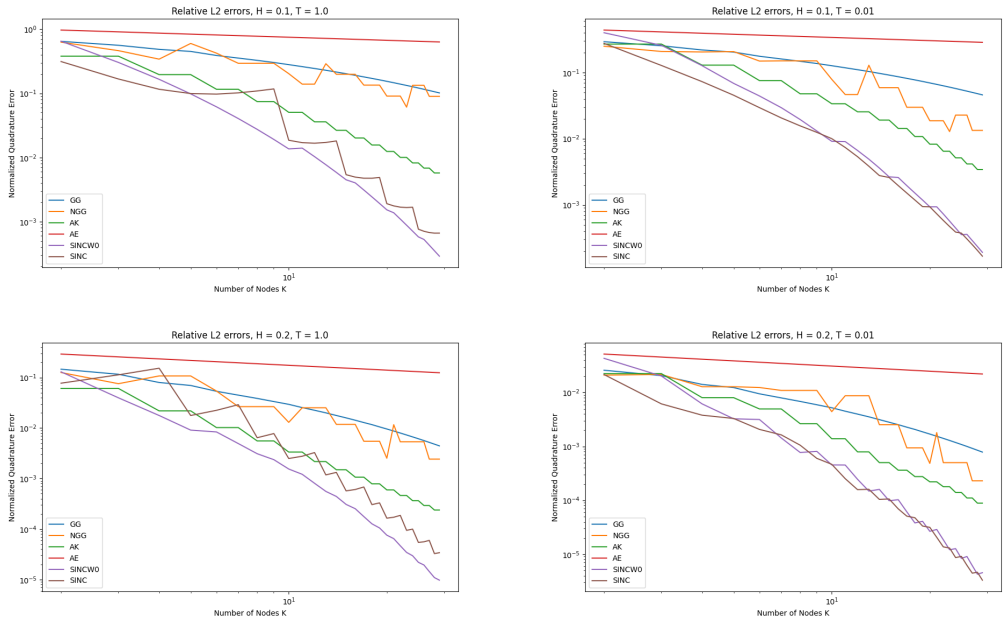

Figure 7.12: Comparison for the relative $\mathcal{L}^2([0,T])$ error between SINC and different quadrature rules introduced in chapter 5. On the left column, we chose $T = 1$, while for the right column we have $T = 0.01$

First we note that the only case where the SINC quadrature is actually not able to perform well with respect to the other quadrature rules is the case where the Hurst parameter H is greater than 0.3. On the other hand, the SINC quadrature outperforms every other rule in all the other cases, both in the pre-asymptotic range and in the asymptotic range. The situation instead for the $\mathcal{L}^1([0,T])$ error is quite different. In Figure 7.13 we plotted the $\mathcal{L}^1([0,T])$ error for different terminal values of T, and focusing on negative values of the Hurst parameter. In those cases, the sinc quadrature is able to outperform the state of the art. In the right column of Figure 7.13, we choose to plot the relative error for $T = 0.01$, and in addition, we used also the second Algorithm that we have developed for small T.

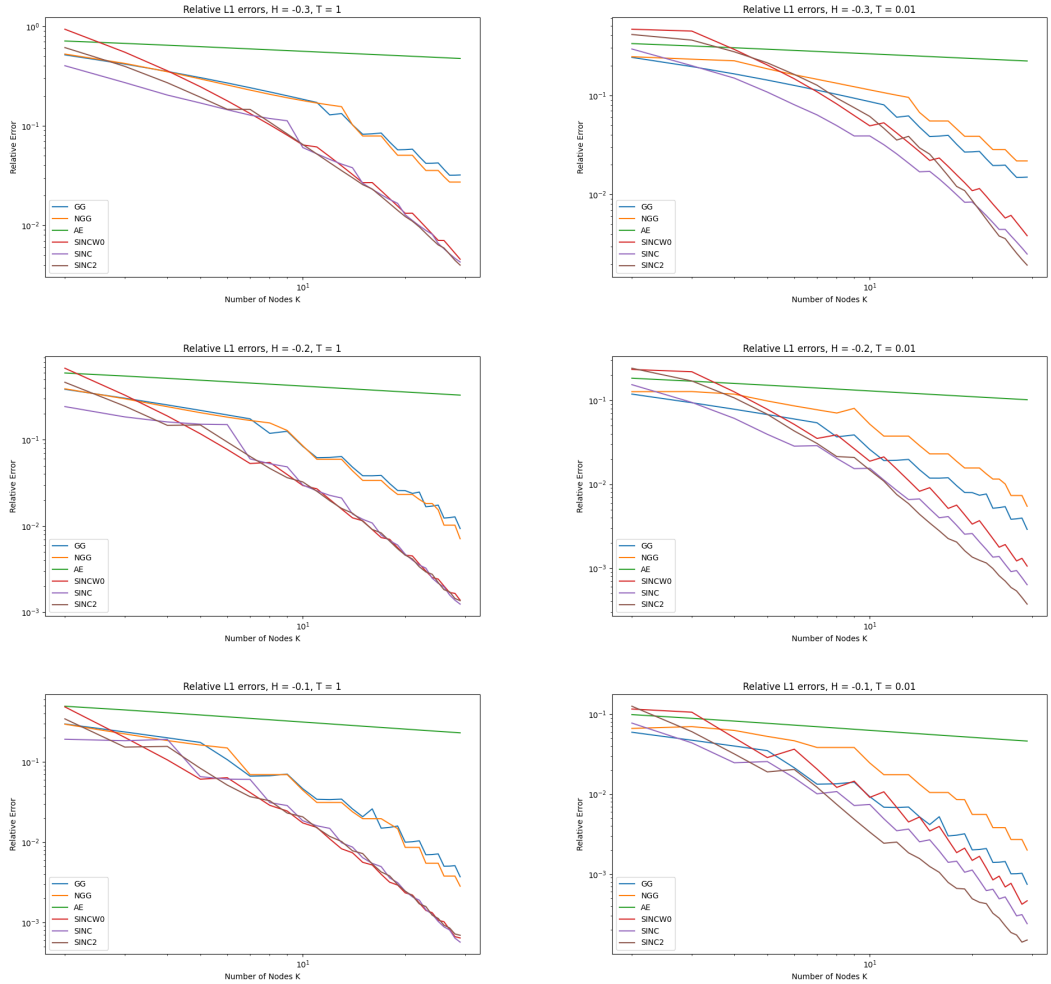

Figure 7.13: Comparison for the relative $\mathcal{L}^1([0,T])$ error between SINC the different quadrature rules introduced in chapter 5. On the right column we consider $T = 1$ while on the left $T = 0.01$ and we consider different negative Hurst parameters.

As for Figure 7.14, we plotted the same type of error, while focusing on the range of Hurst parameter for which the SINC quadrature is not fully able to outperform quadratures like NGG or GG. As we can see, from $H = 0.2$, we already have some underperforming of the quadrature rule, in particular for the small terminal time T.

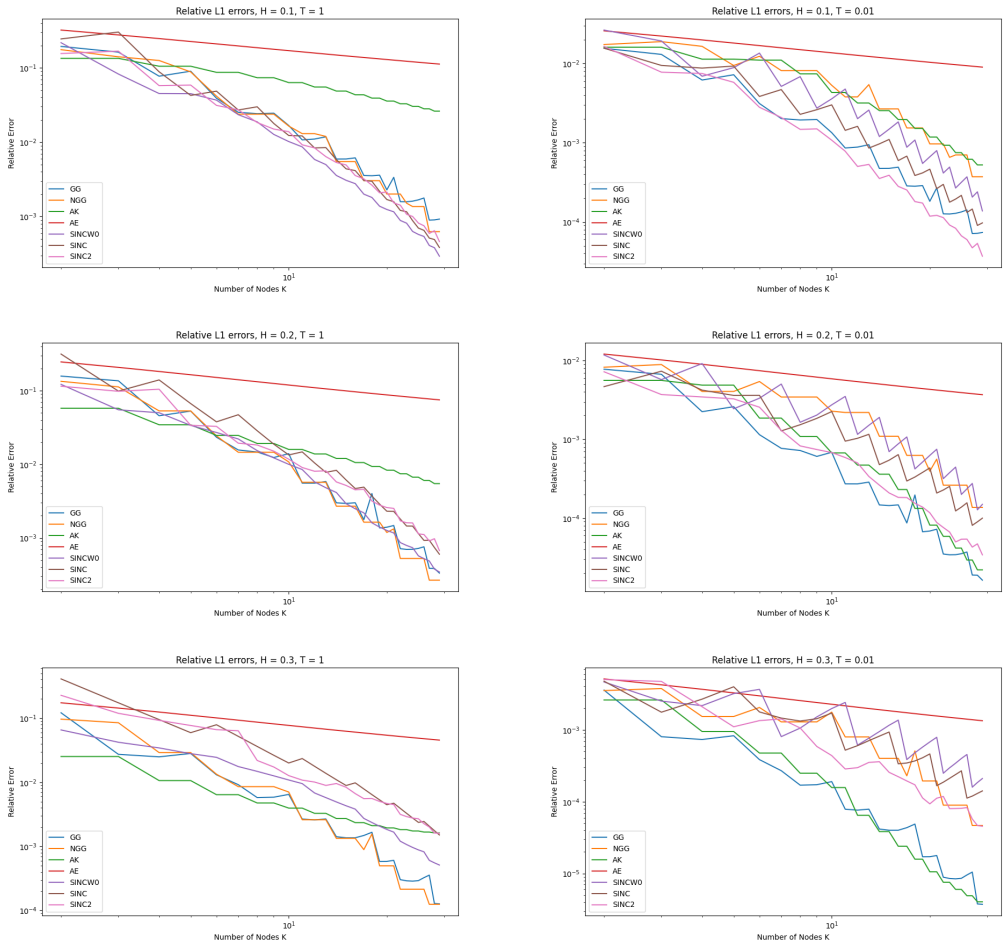

Figure 7.14: Comparison for the relative $\mathcal{L}^1([0,T])$ error between SINC the different quadrature rules introduced in chapter 5. Here $T = 0.01$

In figure 7.14 we can see how the second algorithm, denoted in the plots with 'SINC2' is able to outperform every quadrature rule for $H < 0.2$ and small T, while the first algorithm that we have given remains behind. Let us plot (7.1), with $K = 1$, for different values of H, against the convergence rate given by the NGG and the GG rules.

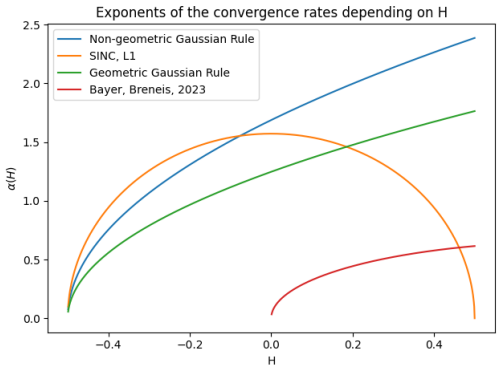

Figure 7.15: coefficient in front of the $\sqrt{N}$ inside the exponential.

Here, in figure 7.15 we plotted:

$$2.38\sqrt{(H+0.5)}, \quad \text{NGG rate}$$
$$2\log(1+\sqrt{2})\sqrt{(H+0.5)}, \quad \text{GG rate}$$
$$\sqrt{\pi^2(0.5-H)(H+0.5)}, \quad \text{L1-SINC rate}$$

One one hand, it seems that the SINC would under perform both Geometric Gaussian and Non-Geometric Gaussian quadrature rules for positive values of the Hurst parameter H. However, as we have seen in this section, using SINCW0 would give an impressive boost to the performance of the algorithm for this case. Since also we noted that we could outperform up until $H \approx 0.2$, we computed the $\mathcal{L}^1([0,1])$ errors and then estimated the coefficient α in front of the exponential of the convergence rate $e^{-\alpha\sqrt{N}}$ using a simple linear regression. What we have found is the following:

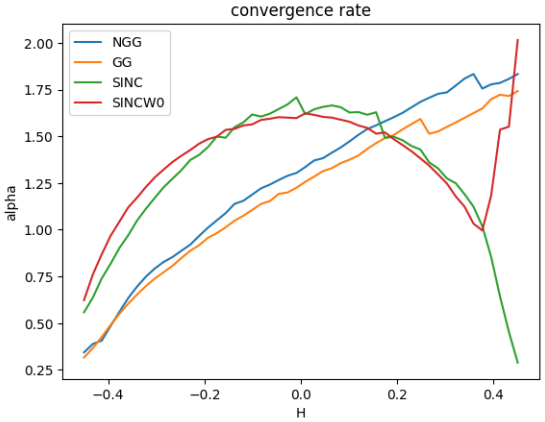

Figure 7.16: estimated coefficient α for different quadrature rules

As we can see, the estimated convergence rate surpasses the one of the Geometric Gaussian rule and the Non Geometric Gaussian rule for every $H \in (-0.5, 0.2)$.

7.3.3. Option Pricing under Rough Heston

In this section we are going to benchmark our method in computing option prices under the Rough Heston model. To compute option prices, we used the function `rHestonFourieriveurcall` of the library provided by Bayer in the GitHub repository in [8]. The function uses Fourier Inversion technique to compute option prices and implied volatility.

In particular, we are in the following setting for the Rough Heston dynamics:

$$\begin{aligned} dS_t &= S_t\sqrt{V_t}dW_t \\ V_t &= V_0 + \int_0^t g(t-s)(\theta(s) - \lambda V_s)\,ds + \int_0^t g(t-s)\,\nu\sqrt{V_t}dB_s \\ dW_t dB_t &= \rho dt \end{aligned}$$

and we use the same set of parameters that are used in [8], id est:

$$\lambda = 0.3, \quad \theta \equiv 0.02, \quad V_0 = 0.02, \quad S_0 = 1, \quad \nu = 0.3 \tag{7.8}$$

With this set of parameters for the Rough Heston model, we are going to compute option prices and implied volatility smiles for maturity $T = 1$ using 451 linearly-spaced log-strikes in the interval $[-1.5, 0.75]$. Then, we are going to compare the results obtained using the algorithms proposed with the state of the art. First, let us plot the implied volatility smile for different values of quadrature points used in the SINC:

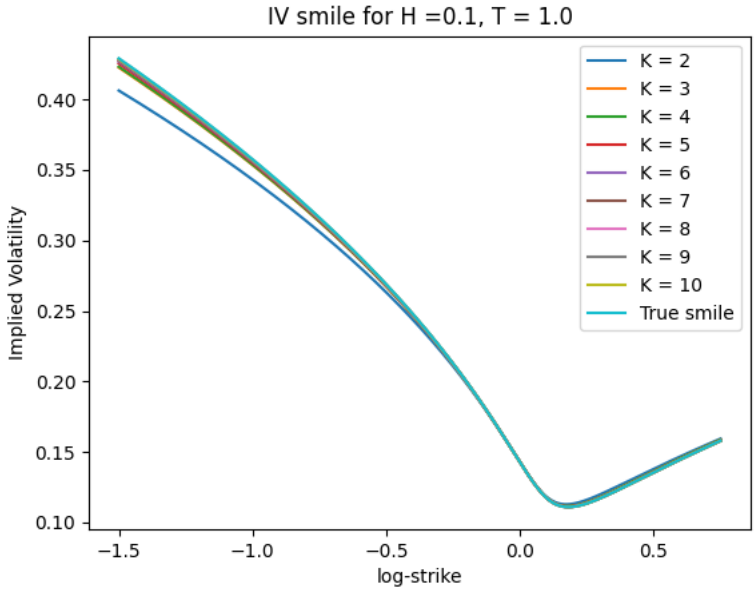

Figure 7.17: Markovian volatility smiles given by L1-SINC-W0 plotted against the true smile given by the Rough Heston model.

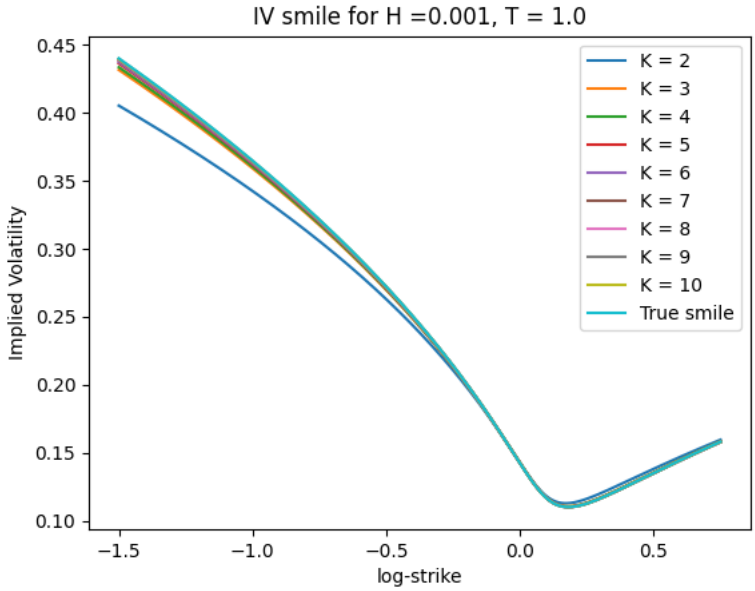

Figure 7.18: Markovian volatility smiles given by L1-SINC-W0 plotted against the true smile given by the Rough Heston model.

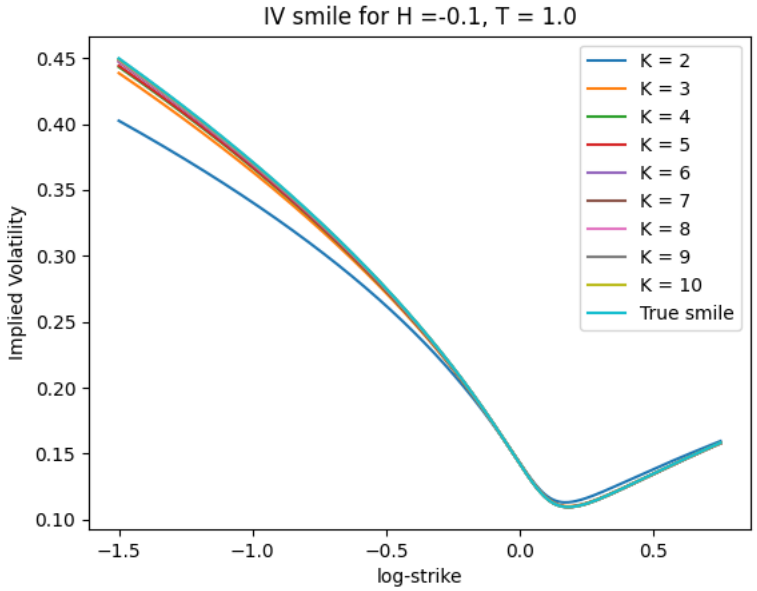

Figure 7.19: Markovian volatility smiles given by L1-SINC-W0 plotted against the true smile given by the Rough Heston model.

As one can observe, the smile given by the Markovian approximations become almost indistinguishable from the true smile for every K greater than 5 for various values of the Hurst parameter.

To have a better understanding on how our proposed methods compare with the state of the art, let us present Table given in [8, Sect. 4 pg. 34] that gives the maximal relative errors in % for the Markovian approximations for implied volatility smiles of the European call option:

	$H=-0.1$						$H=0.001$							$H=0.1$						
K	SINC	GG	NGG	OL1	BL2	AE	SINC	GG	NGG	OL1	OL2	BL2	AE	AK	SINC	GG	NGG	OL1	OL2	BL2	AE	AK
3	2.608	**13.35**	**3.46**	2.527	0.112	**34.46**	1.644	**8.704**	**15.75**	1.147	**6.394**	0.101	**21.56**	**8.138**	2.737	**6.017**	**12.31**	0.738	0.553	0.066	**17.60**	1.112
4	0.748	**10.58**	**14.16**	**1.281**	0.012	**31.01**	1.927	**7.066**	**13.16**	0.525	**5.848**	0.007	**19.28**	**24.38**	1.519	**4.405**	**9.812**	0.306	0.427	0.005	**15.92**	**2.004**
5	0.896	**8.802**	**5.84**	**0.681**	0.002	**28.56**	0.829	**7.599**	**10.74**	0.262	**5.395**	0.001	**17.65**	**24.38**	0.523	**5.058**	**6.501**	0.140	0.319	0.001	**14.67**	**2.004**
6	0.746	**6.109**	**13.38**	0.399	0.000	**26.72**	0.461	**3.161**	**10.83**	0.138	**5.051**	0.000	**16.41**	**29.50**	0.656	**2.121**	**9.107**	0.069	0.236	0.000	**13.68**	**1.659**
7	0.432	**3.707**	**11.78**	0.238	0.000	**25.27**	0.508	**1.965**	**7.282**	0.077	**4.740**	0.000	**15.43**	**29.50**	0.413	**1.371**	**5.525**	0.029	0.174	0.000	**12.88**	**1.659**
8	0.333	**3.697**	**11.78**	0.147	0.001	**24.07**	0.314	**1.898**	**7.282**	0.044	**4.469**	0.000	**14.63**	**31.27**	0.370	**1.245**	**5.525**	0.019	0.128	0.000	**12.21**	**0.999**
9	0.288	**3.844**	**11.78**	0.079	0.002	**23.07**	0.297	**1.932**	**7.282**	0.026	**4.228**	0.000	**13.95**	**31.27**	0.277	**1.206**	**5.525**	0.008	0.094	0.000	**11.64**	**0.999**
10	0.217	**2.482**	**7.052**	0.047	0.003	**22.22**	0.192	**1.263**	**4.476**	0.012	**4.010**	0.000	**13.38**	**30.54**	0.218	**0.804**	**3.414**	0.004	0.070	0.000	**11.15**	**0.503**

Table 7.2: Maximal Relative errors (in %) of IV smiles for different quadrature rules. Results of the other algorithms are taken from [8, Table 3, Section 4]

- In blue, our proposed quadrature rule
- In bold **red** are the values that we are able to outperform
- In black, the values that we cannot outperform

With maximal relative error we refer to the following quantity: for a given Hurst parameter H and a given maturity T, we will have the so-called true smile, which is given by the real Rough Heston model. We will denote as the True smile with respect to the log-strikes $\{K_i\}_{i=1}^n$ and with maturity T the set of all implied volatilities $IV_r(T, K_i)$ with the different strikes under the Rough Heston model, i.e.:

$$IV_r := \{IV_r(T, K_i)\}_{i=1}^n$$

While we will denote the Implied volatility smile under the Markovian approximation as

$$IV_M := \{IV_M(T, K_i)\}_{i=1}^n .$$

. Then one can define the maximal relative error between the true smile and its approximation by:

$$\max_{i=0,\dots,n} \frac{|IV_r(T, K_i) - IV_M(T, K_i)|}{IV_r(T, K_i)}$$

So if we compare our results contained in Table 7.2, we can see how our method, in particular L1-SINC is able to outperform every quadrature rule that is not the result of a direct optimisation of the $\mathcal{L}^p([0,T])$ error for $p \in \{1,2\}$. In fact the only algorithms that consistently perform better than the SINC rule are OL1 and BL2. For values of H close to zero, our method is even able to perform better than the OL2 algorithm proposed by Bayer in [8]. Then, we computed implied volatility smiles under the Rough Heston model for European Call Options with maturity $T = 1$, what we found is presented in Figure 7.24.

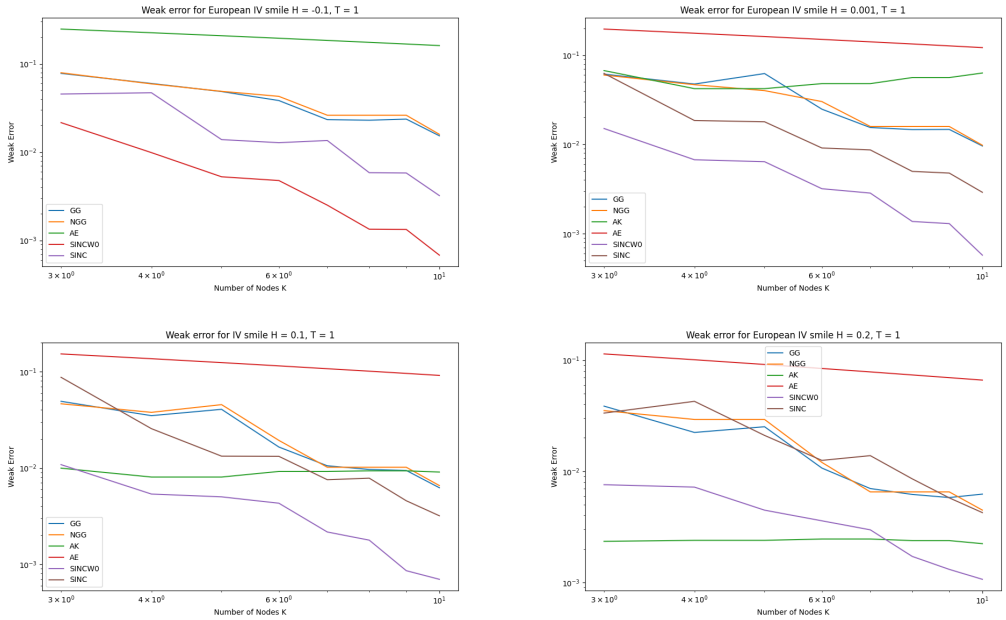

Figure 7.20: Implied Volatility Smile relative errors for different quadrature rules and different values of the Hurst parameter H.

Also in this case, we manage to outperform every quadrature rule.

Then, to give a full picture, we also reproduced the experiment done in [8], where they computed the whole Implied Volatility surface in the Rough Heston case and then they compared it with the Implied Volatility surface given by the Markovian approximation. In particular, without changing the set of parameters given in (7.8) for the Rough Heston model. For maturities and strikes, we used the same as in [8] for the sake of reproducibility of the results. In particular we consider implied volatility surfaces, using 25 maturities linearly spaced in $[0.04, 1]$, where we define $T_{min} := 0.04$ and $T_{max} := 1$. For each maturity T we take 301 linearly spaced values of log-moneyness in the interval $[-1, 0.5]\cdot\sqrt{T}$. Then, similarly to what Bayer has done in [8], we calibrated the nodes and the weights of the SINC quadrature by using a fixed maturity T_0. In particular we have chosen T_0 in the same way of Bayer, i.e.: given T_{min}, T_{max} respectively the minimum and the maximum maturity in the Implied Volatility surface, then defined T_0 as:

$$T_0 = T_{min}^{\alpha(K)} T_{max}^{1-\alpha(K)}$$

where K is the total number of nodes and $\alpha : \mathbb{N} \to \mathbb{R}$ defined as:

$$\alpha(1) = 3/5, \quad \alpha(2) = \frac{1}{2}, \quad \alpha(3) = \frac{1}{3}, \quad \alpha(4) = \frac{1}{4}$$

$$\alpha(5) = \frac{1}{6}, \quad \alpha(6) = \frac{1}{10}, \quad \alpha(N) = 0 \quad \forall N \geq 7$$

Note that this choice for T_0 is not straightforward and we have not optimised this choice because of the long computational times of implied volatility surfaces. In general we noticed that in our case, for the SINC quadrature, it might be beneficial to choose T_0 closer to zero, since we have observed that for lower maturities, the approximation performs the worst, while on higher maturities, we have good

results, this behaviour can be easily seen by looking at the heatmaps that we have plotted in Figure 7.21,7.22,7.23. Also, it is worth noticing that we have been using the T_0 for calibrating our model on the whole volatility surfaces equal to the one chosen by Bayer. However, it is very likely that this is not the optimal choice for our method.

	$H=-0.1$						$H=0.001$								$H=0.1$							
K	SINC	GG	NGG	OL1	BL2	$\overline{\text{AE}}$	SINC	GG	NGG	OL1	OL2	BL2	$\overline{\text{AE}}$	AK	SINC	GG	NGG	OL1	OL2	BL2	AE	AK
3	5.085	18.14	18.14	4.975	0.806	52.76	5.601	15.12	12.87	3.203	14.52	0.948	40.45	19.10	2.884	14.22	11.09	2.632	6.808	1.012	33.71	4.089
4	5.085	14.68	13.81	2.881	0.644	53.02	2.217	12.26	9.995	1.583	11.88	1.044	40.74	24.45	1.429	12.27	8.611	1.233	3.679	0.807	33.07	2.426
5	3.331	12.38	11.15	1.620	0.176	53.61	2.872	12.43	8.213	0.900	14.55	0.274	41.42	26.41	1.316	11.67	19.72	0.676	1.505	0.194	34.57	2.153
6	2.753	11.12	9.455	1.043	0.012	54.04	1.291	8.618	11.90	0.531	12.52	0.056	41.94	26.59	0.776	8.732	9.581	0.372	0.623	0.036	35.03	1.218
7	1.371	8.756	10.84	0.720	0.012	55.16	1.599	7.406	6.733	0.338	16.00	0.012	43.36	26.98	0.834	7.845	5.410	0.188	0.414	0.008	36.31	1.473
8	0.642	7.604	10.84	0.720	0.002	54.58	0.672	5.997	6.733	0.196	16.02	0.002	42.60	26.56	0.450	6.253	5.410	0.126	0.284	0.002	35.62	1.197
9	0.823	6.933	10.84	0.229	0.001	54.02	0.877	5.077	6.733	0.117	16.00	0.001	41.92	26.56	0.316	5.148	5.410	0.053	0.289	0.000	35.01	1.197
10	0.439	5.121	5.967	0.136	0.001	53.50	0.420	3.829	3.557	0.056	15.96	0.001	41.29	26.56	0.190	3.940	2.831	0.026	0.228	0.000	34.45	0.994

Table 7.3: Maximal Relative errors (in %) of IV surfaces for different quadrature rules. Results of the other algorithms are taken from [8, Table 4, Section 4]

It is worth noticing that our proposed algorithm is able to ouperform every quadrature rule, besides the one that are the result of an optimisation procedure, id est OL1,OL2 and BL2. Also, we have noticed that the argmax of the relative error for implied volatility surfaces comes always from short maturities. In fact it is quite interesting that even for a low number of quadrature points, we can already achieve high accuracy of $\approx 0.8\%$ on most of the IV surface, however, for small maturities, and also for deep in the money options the error becomes predominant, this is clear in Figure 7.21, where even for 6 total nodes in the quadrature rule, we have errors that are below 1% for every maturity $T > 0.08$. Interestingly enough, is the fact that for small maturities, in this case for $T = 0.04$ then the Markovian approximation given by the L1-SINC algorithm fails to approximate both deep in the money options, out of the money options but also at the money options. In fact the maximal error for $N = 6$ is attained for $T = 0.04$ and for log-strike $K \approx 1$. Approximately the same phenomenon appears in Figure 7.22, where we plotted the relative errors for the implied volatility surface for $H = -0.1$. For reference, even though we have a maximal error of 0.75%, we computed also the mean relative error for the whole surface and obtained 0.196%.

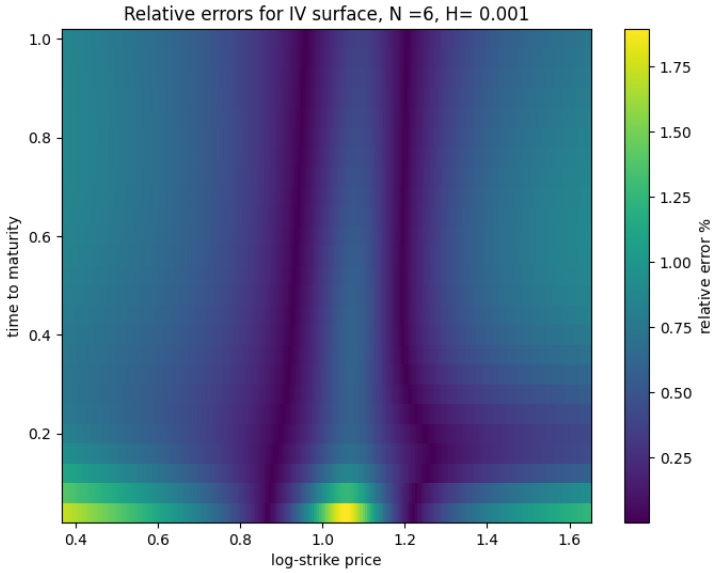

Figure 7.21: Relative errors for Implied volatility surface in the Rough Heston model, for a total number of nodes $N = 6$, using the L1-SINC for Hurst parameter $H = 0.001$

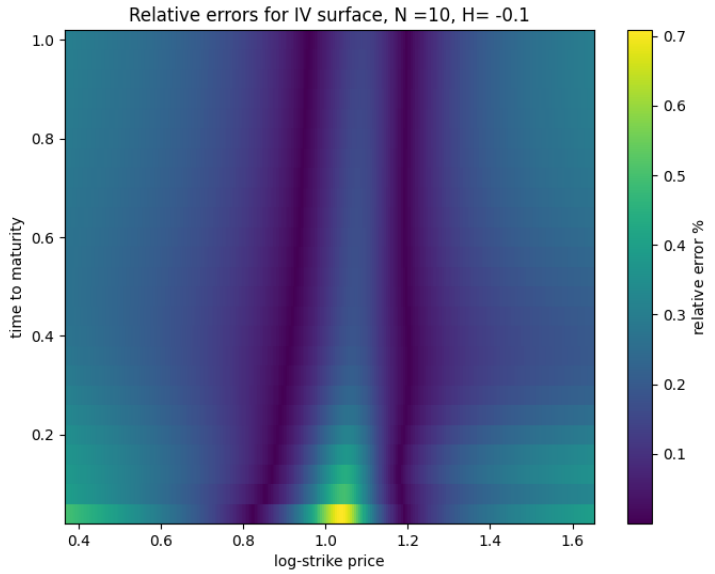

Figure 7.22: Relative errors for Implied volatility surface in the Rough Heston model, for a total number of nodes $N = 10$, using the L1-SINC for Hurst parameter $H = -0.1$

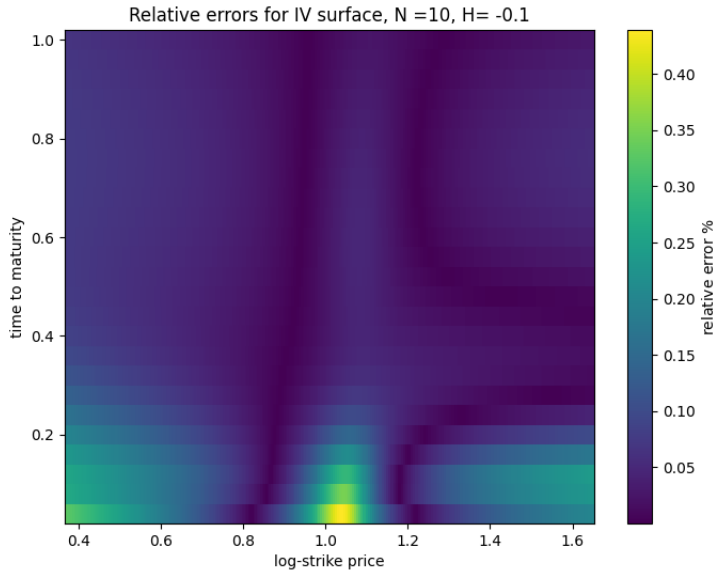

Figure 7.23: Relative errors for Implied volatility surface in the Rough Heston model, for a total number of nodes $N = 10$, using the L1-SINC-W0 for Hurst parameter $H = -0.1$

As a final experiment, we computed prices for continuously monitored geometric Asian options, which can still be evaluated using Fourier pricing methods. Specifically, a continuously monitored geometric Asian option is an option where the underlying asset is the following quantity:

$$A(0,T) = \exp\left(\frac{1}{T}\int_0^T \log\left(S_t\right) dt\right).$$

Then, the payoff of such instrument will be:

$$(A(0,T)-K)^+ \qquad \text{for a Call}$$
$$(K-A(0,T))^+ \qquad \text{for a Put}$$

The goal of this experiment, along with computing the implied volatility surfaces, is to determine if the Markovian Approximation of Rough Volatility models using SINC quadrature can accurately capture the probability distributions of the underlying asset and volatility not only at the terminal time but also at intermediate maturities. While this property of the Markovian approximation has been verified by computing the entire implied volatility surface without adjusting the quadrature rule for each expiration, we aim to perform an additional check for thoroughness.

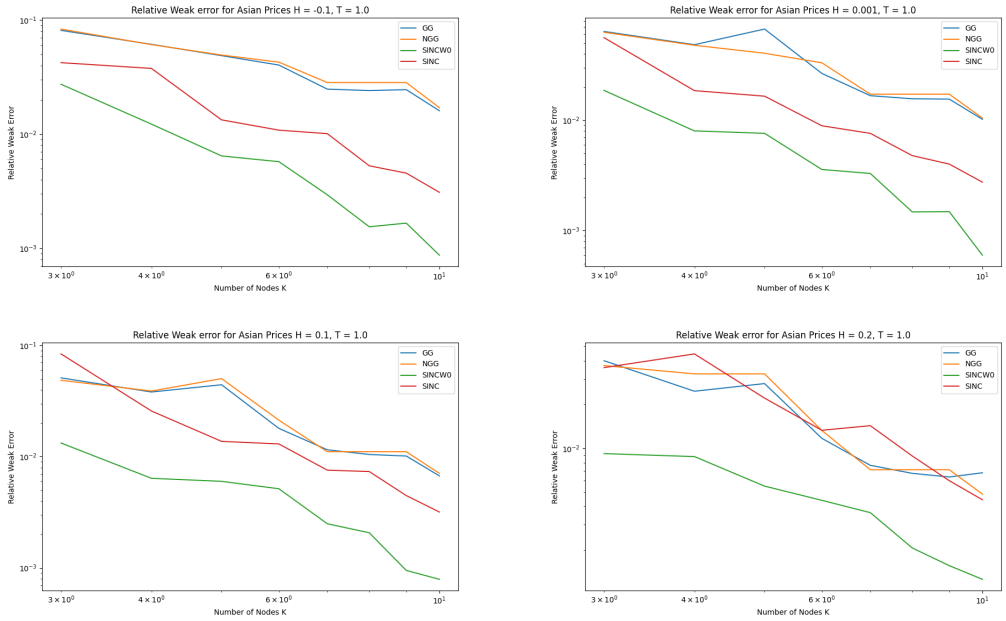

Figure 7.24: relative weak errors for asian call options for different quadrature rules and different values of the Hurst parameter H.

As we can see, also in this experiment, our proposed quadrature rule is able to outperform both the Geometric and Non-Geometric Gaussian quadrature.

8

Conclusions and future work

Finally, we have come to the conclusion of this work. First, we have introduced Rough Volatility models and their ability to describe Market's properties. Then in chapter 5 we deep-dived into the Markovian approximation for such models. In chapter 6 we introduce a new quadrature rule, the SINC quadrature rule, computing theoretical error bounds for the SINC quadrature, in particular regarding $\mathcal{L}^p([0,T])$ errors, for $p \in \{1,2\}$. With those error bounds, we developed two algorithms in chapter 7, one for the quadrature rule when the final time considered T is less than 0.1, and the other one for $T > 0.1$. Finally, in chapter 7, we compared the SINC quadrature with different quadrature rules present in the literature. What we found is that the SINC quadrature is able to outperform every other quadrature rule both from the point of view of the weak error and the strong error. The boosted performance are restricted to choices of values of the Hurst parameter $H \in (-0.5, 0.2)$, which is a range which is the most interesting for financial applications. The main advantage of our quadrature rule is the following: despite it is extremely easy to implement, especially if compared to Gaussian Quadrature, where an orthogonal basis of polynomials must be constructed for obtaining the nodes and weights of the quadrature, we showed how we can achieve state of the art results in various experiments.

To get to those results, we heavily relied on the error bounds given for the SINC quadrature in [36].

As future research, one could investigate more in the directions of approximating Rough Volatility models with $H \in (0.2, 1)$, we have seen that despite the fact that the SINC quadrature rule is able to have extremely good performances for $H \in (-0.5, 0.2)$, we have observed a decay in the convergence rate when $H > 0.2$, and in particular, we did not consider at all the non-fractional case, where $H > 0.5$.

Moreover, one could try to investigate another class of change of variables that might be beneficial to an improvement of the convergence rate of the SINC quadrature, in particular using the so called Double Exponential transformations. In particular, with double exponential we suggest that instead of using the change of variables:

$$\rho = e^x$$

instead one would deal with

$$\rho = e^{sinh(x)}$$

where ρ is the variable that is being integrated in the integral:

$$\int_0^T e^{-\rho\tau} \rho^{-(H+1/2)} d\rho$$

We have tried to investigate this possibility, however the challenges are two: first, the error analysis is not as easy as the one of the standard SINC quadrature that we have presented in this work. In fact one has to compute integrals on the complex plane that have no analytical solutions and have no straightforward bounds. With this in mind, we have tried to deal with the double exponential but it led to extremely large nodes and weights, and hence if one has no control on the error bound, then choosing an appropriate step-size can become an extremely difficult task, as one has to deal with underflow and overflow errors that arise when the step-size is too large.

9

Aknowledgments

Finally, after nine months, here we are at the acknowledgments. Firstly, I would like to thank my supervisor Kristin Kirchner, for helping me in the moments of discomfort and stress, always giving me a positive feedback and reminding me to keep my head up when I was way too much self critical about my work. Secondly I want to thank Antonis Papapantoleon, for the guidance and recommendations.

Per quanto riguarda i riconoscimenti non accademici e non istituzionali, vorrei ringraziare con tutto il cuore i miei genitori, Laura Ciusani e Davide Pezzoli, per il prezioso sostegno che mi hanno dato in questi anni, appoggiando sempre le mie decisioni, aiutandomi nei momenti bui, ma soprattutto per la fiducia che hanno in me. Voglio ringraziarvi per tutti i sacrifici che avete fatto, per l'impegno e la pazienza che avete avuto e che avete tuttora. Voglio dire che non c'è giorno in cui non pensi a quello che avete fatto per me e per tutto il bene che mu volete. Vi sono e vi sarò sempre grato. Vorrei poi ringraziare poi della mia famiglia mia Zia Paola, per tutte le volte che abbiamo pranzato insieme a Milano, per rendere possibile un florido scambio intergenerazionale.

Per i miei amici infine vorrei ringraziare tutte le persone che ho conosciuto a Delft, con cui abbiamo condiviso momenti e esperienze incredibili. Tra di loro vorrei ringraziare in particolare Albert, Albin, Alessandro, Fernando, Francesco, Gideon, Matthias, Panos, Songyang e Tomas. Grazie a voi che mi avete sia fatto compagnia in questi anni, sia mi avete motivato e trainato in questo viaggio.

Un ringraziamento va ai miei coinquilini, Laura, Lourens, Niels e Karim. In particolare grazie a Laura per l'anno trascorso insieme, per aver esserci sostenuti l'un l'altro. Grazie a Lourens per tutta la compagnia che ci siamo fatti, per le discussioni e gli scambi di opinione sempre costruttivi. Grazie a Niels infine per tutta la musica che abbiamo condiviso e cantato a squarciagola.

Un ringraziamento a tutti i miei amici in Italia e nel mondo, a tutto il gruppo di Chemminchiabici. Tra di loro non posso non ringraziare esplicitamente Gio per tutti i passaggi e tutti i momenti insieme sempre genuini, (M) Dario per tutte le discussioni sulla filosofia che ogni uomo di tecnica e di scienza non dovrebbe mai lasciar nell'oblio, Mea per tutte le esperienze condivise.Un ringraziamento ai miei compagni di triennale, Nicola e Carlo, con cui ho passato tre anni fantastici grazie a voi. Un ringraziamento speciale infine alle persone più vicine con cui ho sempre avuto discussioni che mi fanno metter in dubbio le mie opinioni e che mi son state affianco per quasi una decade ormai: grazie Isa per il tuo supporto e sostegno, grazie a Joris a alla tua famiglia, che mi ha regalato indimenticabili momenti e che mi ha fatto sempre sentire a casa, grazie a Edo per le infinite discussioni e per tutti i giri in montagna . Infine vorrei ringraziare in particolare Alessandro, che nonostante la distanza continua a rimanere un punto di riferimento per la costanza, la motivazione e la sua infinita voglia di eccellere e forza di volontà.

References

[1] Eduardo Abi Jaber and Omar El Euch. “Multifactor approximation of rough volatility models”. In: *SIAM journal on financial mathematics* 10.2 (2019), pp. 309–349.

[2] Eduardo Abi Jaber and Shaun Xiaoyuan Li. “Volatility models in practice: Rough, Path-dependent or Markovian?” In: *Path-Dependent or Markovian* (2024).

[3] Yogesh J Bagul and Christophe Chesneau. “Some new simple inequalities involving exponential, trigonometric and hyperbolic functions”. In: *Cubo (Temuco)* 21.1 (2019), pp. 21–35.

[4] Peter Bank et al. “Rough PDEs for local stochastic volatility models”. In: *arXiv preprint arXiv:2307.09216* (2023).

[5] Christian Bayer, Chiheb Ben Hammouda, and Raúl Tempone. “Hierarchical adaptive sparse grids and quasi-Monte Carlo for option pricing under the rough Bergomi model”. In: *Quantitative Finance* 20.9 (2020), pp. 1457–1473.

[6] Christian Bayer and Simon Breneis. “Efficient option pricing in the rough Heston model using weak simulation schemes”. In: *arXiv preprint arXiv:2310.04146* (2023).

[7] Christian Bayer and Simon Breneis. “Markovian approximations of stochastic Volterra equations with the fractional kernel”. In: *Quantitative Finance* 23.1 (2023), pp. 53–70.

[8] Christian Bayer and Simon Breneis. “Weak Markovian approximations of rough Heston”. In: *arXiv preprint arXiv:2309.07023* (2023).

[9] Christian Bayer, Peter Friz, and Jim Gatheral. “Pricing under rough volatility”. In: *Quantitative Finance* 16.6 (2016), pp. 887–904.

[10] Mikkel Bennedsen, Asger Lunde, and Mikko S Pakkanen. “Hybrid scheme for Brownian semistationary processes”. In: *Finance and Stochastics* 21 (2017), pp. 931–965.

[11] Lorenzo Bergomi. “Smile dynamics II”. In: *Available at SSRN 1493302* (2005).

[12] Pierre Blanc, Jonathan Donier, and J-P Bouchaud. “Quadratic Hawkes processes for financial prices”. In: *Quantitative Finance* 17.2 (2017), pp. 171–188.

[13] Helmut Brass and Knut Petras. *Quadrature theory: the theory of numerical integration on a compact interval*. 178. American Mathematical Soc., 2011.

[14] Peter J Brockwell et al. “The Spectral Representation of a Stationary Process”. In: *Time Series: Theory and Methods* (1991), pp. 114–165.

[15] Philippe Carmona and Laure Coutin. “Fractional Brownian motion and the Markov property”. In: (1998).

[16] Peter Carr and Dilip Madan. “Option valuation using the fast Fourier transform”. In: *Journal of computational finance* 2.4 (1999), pp. 61–73.

[17] Rémy Chicheportiche and Jean-Philippe Bouchaud. “The fine-structure of volatility feedback I: Multi-scale self-reflexivity”. In: *Physica A: Statistical Mechanics and its Applications* 410 (2014), pp. 174–195.

[18] Robert B Davies and David S Harte. “Tests for Hurst effect”. In: *Biometrika* 74.1 (1987), pp. 95–101.

[19] Jules Delemotte, Stefano De Marco, and Florent Segonne. “Yet Another Analysis of the SP500 At-The-Money Skew: Crossover of Different Power-Law Behaviours”. In: *Available at SSRN 4428407* (2023).

[20] *deribit.com*.

[21] Giulia Di Nunno et al. “From Constant to Rough: A Survey of Continuous Volatility Modeling”. In: *Mathematics* 11.19 (2023). ISSN: 2227-7390. URL: https://www.mdpi.com/2227-7390/11/19/4201.

[22] Omar El Euch, Masaaki Fukasawa, and Mathieu Rosenbaum. “The microstructural foundations of leverage effect and rough volatility”. In: *Finance and Stochastics* 22 (2018), pp. 241–280.

[23] Omar El Euch and Mathieu Rosenbaum. “The characteristic function of rough Heston models”. In: *Mathematical Finance* 29.1 (2019), pp. 3–38.

[24] Omar El Euch et al. “The Zumbach effect under rough Heston”. In: *Quantitative finance* 20.2 (2020), pp. 235–241.

[25] Fang Fang and Cornelis W Oosterlee. “A novel pricing method for European options based on Fourier-cosine series expansions”. In: *SIAM Journal on Scientific Computing* 31.2 (2009), pp. 826–848.

[26] Masaaki Fukasawa. “Asymptotic analysis for stochastic volatility: martingale expansion”. In: *Finance and Stochastics* 15 (2011), pp. 635–654.

[27] Paul Gassiat. “Weak error rates of numerical schemes for rough volatility”. In: *SIAM Journal on Financial Mathematics* 14.2 (2023), pp. 475–496.

[28] Jim Gatheral, Thibault Jaisson, and Mathieu Rosenbaum. “Volatility is rough”. In: *Commodities*. Chapman and Hall/CRC, 2022, pp. 659–690.

[29] Jim Gatheral and Martin Keller-Ressel. “Affine forward variance models”. In: *Finance and Stochastics* 23 (2019), pp. 501–533.

[30] Gustaf Gripenberg, Stig-Olof Londen, and Olof Staffans. *Volterra integral and functional equations*. 34. Cambridge University Press, 1990.

[31] Julien Guyon and Mehdi El Amrani. “Does the Term-Structure of Equity At-the-Money Skew Really Follow a Power Law?” In: *Available at SSRN 4174538* (2022).

[32] Ulrich Horst, Wei Xu, and Rouyi Zhang. “Convergence of Heavy-Tailed Hawkes Processes and the Microstructure of Rough Volatility”. In: *arXiv preprint arXiv:2312.08784* (2023).

[33] Blanka Horvath, Antoine Jacquier, and Aitor Muguruza. “Functional central limit theorems for rough volatility”. In: *arXiv preprint arXiv:1711.03078* (2017).

[34] Eduardo Abi Jaber et al. “Volatility models in practice: Rough, Path-dependent or Markovian?” In: *arXiv preprint arXiv:2401.03345* (2024).

[35] Ahmed Kebaier et al. *Approximation of Stochastic Volterra Equations with kernels of completely monotone type*. Tech. rep. 2022.

[36] John Lund and Kenneth L Bowers. *Sinc methods for quadrature and differential equations*. SIAM, 1992.

[37] Ryan McCrickerd and Mikko S Pakkanen. “Turbocharging Monte Carlo pricing for the rough Bergomi model”. In: *Quantitative Finance* 18.11 (2018), pp. 1877–1886.

[38] Robert C Merton. “Theory of rational option pricing”. In: *The Bell Journal of economics and management science* (1973), pp. 141–183.

[39] Cornelis W Oosterlee and Lech A Grzelak. *Mathematical modeling and computation in finance: with exercises and Python and MATLAB computer codes*. World Scientific, 2019.

[40] Walter Rudin. *Real and Complex Analysis*. McGraw-Hill Science/Engineering/Math, 1986. ISBN: 0070542341.

[41] Yat Chun Chester Wong and Paul Bilokon. “Simulation of Fractional Brownian Motion and Related Stochastic Processes in Practice: A Straightforward Approach”. In: *Available at SSRN* (2024).

[42] Andrew TA Wood and Grace Chan. “Simulation of stationary Gaussian processes in [0, 1] d”. In: *Journal of computational and graphical statistics* 3.4 (1994), pp. 409–432.

[43] Xicheng Zhang. “Stochastic Volterra equations in Banach spaces and stochastic partial differential equation”. In: *Journal of Functional Analysis* 258.4 (2010), pp. 1361–1425.