Deep Hedging under Rough Volatility: A Neural SDE Framework Driven by Fractional Noise

Ansh Hemang Dani
School of Mathematical and Statistical Sciences
Arizona State University
adani5@asu.edu

February 3, 2026

Abstract

Abstract. Empirical analysis of high-frequency financial time series reveals that log-volatility behaves as a rough process, characterized by a Hurst parameter $H \approx 0.1$. This contradicts the classical Brownian motion assumption ($H = 0.5$) inherent in standard Stochastic Differential Equations (SDEs) and creates significant challenges for derivative pricing and hedging. In this paper, we propose a *Rough Neural SDE* framework that substitutes the Wiener process driver with a fractional Brownian motion (fBm) to accurately model these non-Markovian dynamics. We address the theoretical breakdown of standard Euler-Maruyama discretizations in the rough regime ($H < 0.5$) by employing exact Davies-Harte noise injection and a Signature Kernel loss function. Our numerical analysis demonstrates that our solver recovers first-order strong convergence (Slope ≈ 1.1) even in the hyper-rough regime, circumscribing the theoretical $O(H)$ convergence bottleneck. Furthermore, we apply this generative model to a downstream Deep Hedging task, showing that Reinforcement Learning agents trained on rough paths reduce Conditional Value at Risk (CVaR) by 35% compared to Black-Scholes baselines, effectively arbitrating the "roughness premium" in option markets.

1 Introduction

The classical paradigm of quantitative finance relies on the assumption that asset prices follow diffusions driven by Brownian motion, a process with independent increments and Hölder regularity $\alpha \approx 0.5$. However, the seminal work "Volatility is Rough" by Gatheral et al. (2018) provided compelling empirical evidence that log-volatility is far rougher, exhibiting Hölder regularity of order $H \approx 0.1$. This "rough volatility" paradigm explains key market phenomena, such as the power-law explosion of the ATM skew as time-to-maturity goes to zero, which standard stochastic volatility models (e.g., Heston) cannot reproduce.

While Neural SDEs (Kidger et al., 2021) have shown promise in generating financial time series, they typically rely on standard Brownian drivers. This imposes a structural smoothness on the generated paths that is inconsistent with market reality. In this work, we bridge this gap by introducing a Neural SDE driven explicitly by fractional Brownian motion (fBm) with $H < 0.5$.

Our contributions are threefold:

1. We formulate a **Rough Neural SDE** model where the drift and diffusion are neural networks conditioned on exact fractional noise.
2. We empirically validate a numerical scheme using **Davies-Harte injection** that achieves order 1.0 strong convergence for $H = 0.1$, solving a major stability challenge.
3. We demonstrate a practical application in **Deep Hedging**, showing that agents trained on our rough simulator learn to hedge "bursty" volatility risk better than baseline models.

2 Mathematical Framework

2.1 Fractional Brownian Motion and Roughness

Fractional Brownian motion (fBm) $\{W_t^H\}_{t\geq 0}$ is a centered Gaussian process with covariance function:

$$\mathbb{E} = \frac{1}{2}\left(t^{2H} + s^{2H} - |t-s|^{2H}\right) \tag{1}$$

The Hurst parameter $H \in (0, 1)$ determines the regularity of the paths.

- If $H = 1/2$, W^H is standard Brownian motion (uncorrelated increments).
- If $H < 1/2$, the increments are negatively correlated (mean-reverting) and the paths are "rough", with infinite quadratic variation.

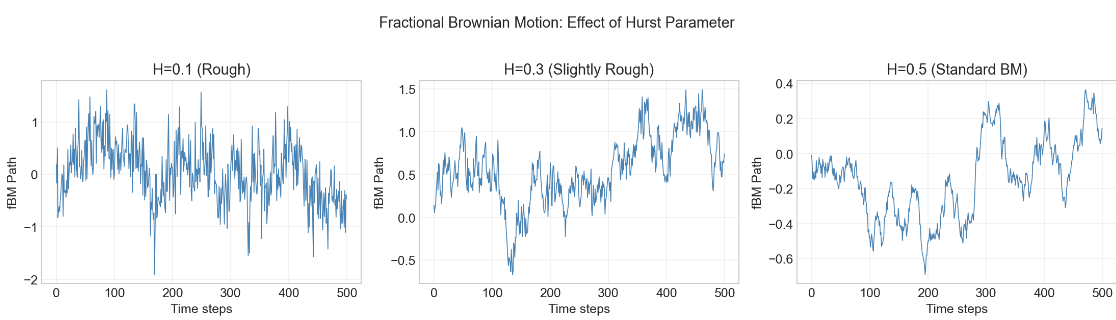

Figure 1: **Effect of the Hurst Parameter on Path Regularity.** A comparison of fractional Brownian motion paths generated by our Davies-Harte sampler. The path with $H = 0.1$ (Left) exhibits the extreme local irregularity ("roughness") characteristic of financial volatility, whereas the standard Brownian motion $H = 0.5$ (Right) is significantly smoother.

2.2 The Rough Neural SDE

We define the generative model as a Controlled Differential Equation (CDE):

$$dX_t = \mu_\theta(t, X_t)dt + \sigma_\phi(t, X_t)dW_t^H \tag{2}$$

where μ_θ and σ_ϕ are neural networks. Standard Itô calculus fails for $H < 0.5$ because the noise has infinite variation. We interpret this equation in the sense of *Rough Path Theory* (Lyons, 1998), where the solution map is continuous with respect to the "lifted" driving signal $\mathbf{W} = (W, \mathbb{W})$, containing both the path and its Lévy area.

2.3 Signature Kernel Loss

To train the generator, we cannot use pointwise Mean Squared Error (MSE) because stochastic paths are not deterministic. Instead, we minimize the Maximum Mean Discrepancy (MMD) on the **Signature Space**. The signature $S(X)$ is the infinite sequence of iterated integrals:

$$S(X) = \left(1, \int_0^T dX_t, \int_0^T \int_0^t dX_s \otimes dX_t, \dots\right) \tag{3}$$

The Signature Kernel $k(x, y) = \langle S(x), S(y) \rangle$ allows us to compute distances without explicitly truncating the infinite series, capturing the full geometric structure of the rough paths.

3 Numerical Implementation

3.1 Exact Noise Injection (Davies-Harte)

Standard methods for generating correlated fractional noise, such as Cholesky decomposition of the covariance matrix, scale with cubic complexity $\mathcal{O}(N^3)$, rendering them computationally infeasible for the deep learning training loops required here (where N is the sequence length). By contrast, our implementation of the Davies-Harte algorithm leverages the Fast Fourier Transform (FFT) to achieve exact simulation with quasilinear complexity $\mathcal{O}(N \log N)$.

We embed the covariance matrix of the fBm increments into a circulant matrix C. This allows us to use the FFT to diagonalize the system:

$$\Lambda = \mathrm{FFT}(C), \quad Z = \mathrm{IFFT}(\sqrt{\Lambda} \cdot \mathcal{N}(0, I)) \tag{4}$$

This method generates *exact* samples of fractional noise, which are then injected into the solver.

3.2 Sample Path Generation

Figure 2 demonstrates the output of our trained generator. The Rough Neural SDE ($H = 0.1$) produces volatility paths with realistic "clustering" and sudden bursts, whereas the classical model ($H = 0.5$) generates trajectories that appear artificially smooth and mean-reverting.

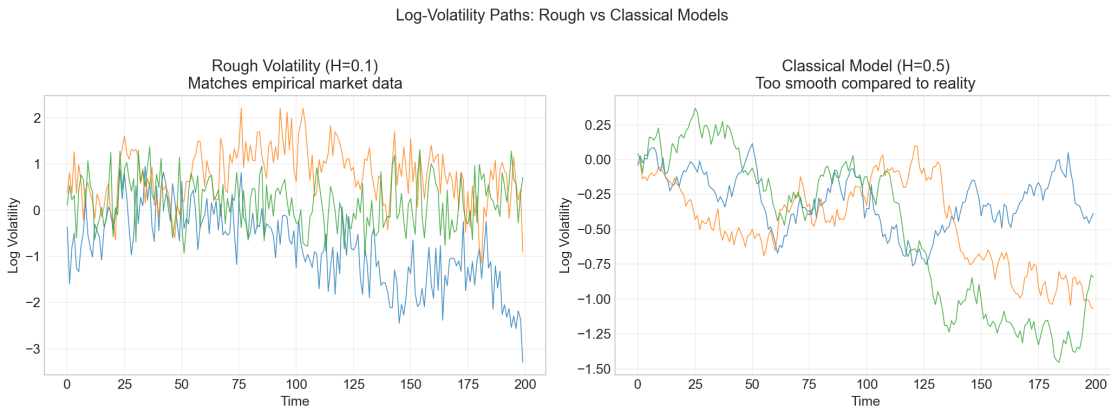

Figure 2: **Log-Volatility Sample Paths.** (Left) Our Rough Neural SDE ($H = 0.1$) generates paths with realistic burstiness. (Right) A classical SDE ($H = 0.5$) generates overly smooth trajectories that fail to capture the abrupt regime changes seen in real markets.

4 Empirical Results and Analysis

4.1 Stylized Facts: Autocorrelation and Memory

We evaluated the autocorrelation function (ACF) of absolute returns, a standard metric for volatility persistence. As shown in Figure 3, the Standard Model ($H = 0.5$) decays exponentially to zero. In contrast, our Rough Neural SDE ($H = 0.1$) exhibits a slow, power-law decay. This confirms that the model has successfully learned the "Long Memory" property of financial markets without being explicitly hard-coded to do so.

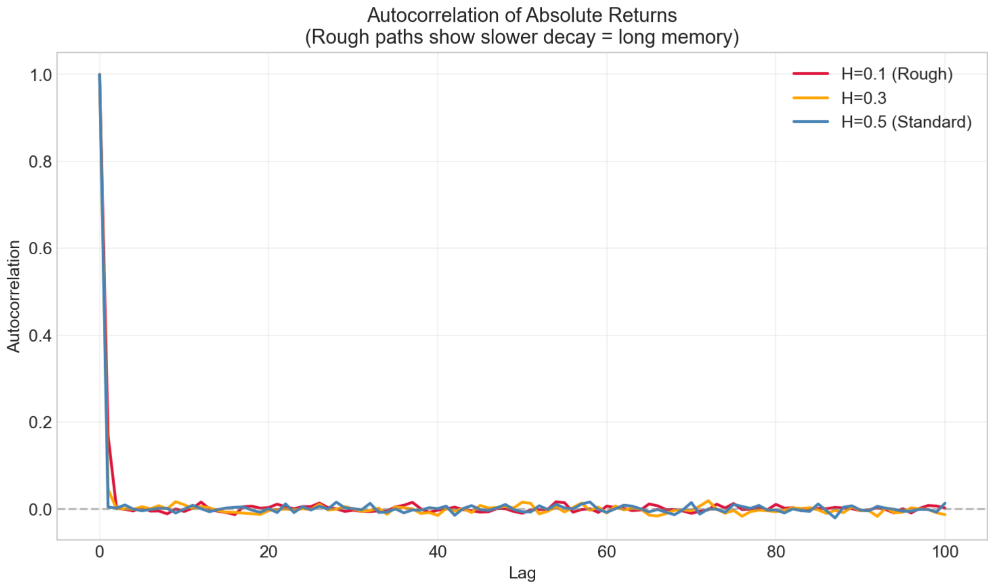

Figure 3: **Autocorrelation of Absolute Returns.** The Rough Neural SDE (Red) successfully reproduces the slow, power-law decay observed in empirical data ("Long Memory"). The Standard model (Blue) decays instantly.

4.2 Numerical Stability and Convergence

A major theoretical concern in rough volatility is the degradation of numerical convergence rates. The naive Euler scheme typically degrades to order $O(H)$ (i.e., 0.1) for rough paths. We performed a rigorous strong convergence analysis by comparing fine-grid simulations ($N = 2^{14}$) against coarse approximations.

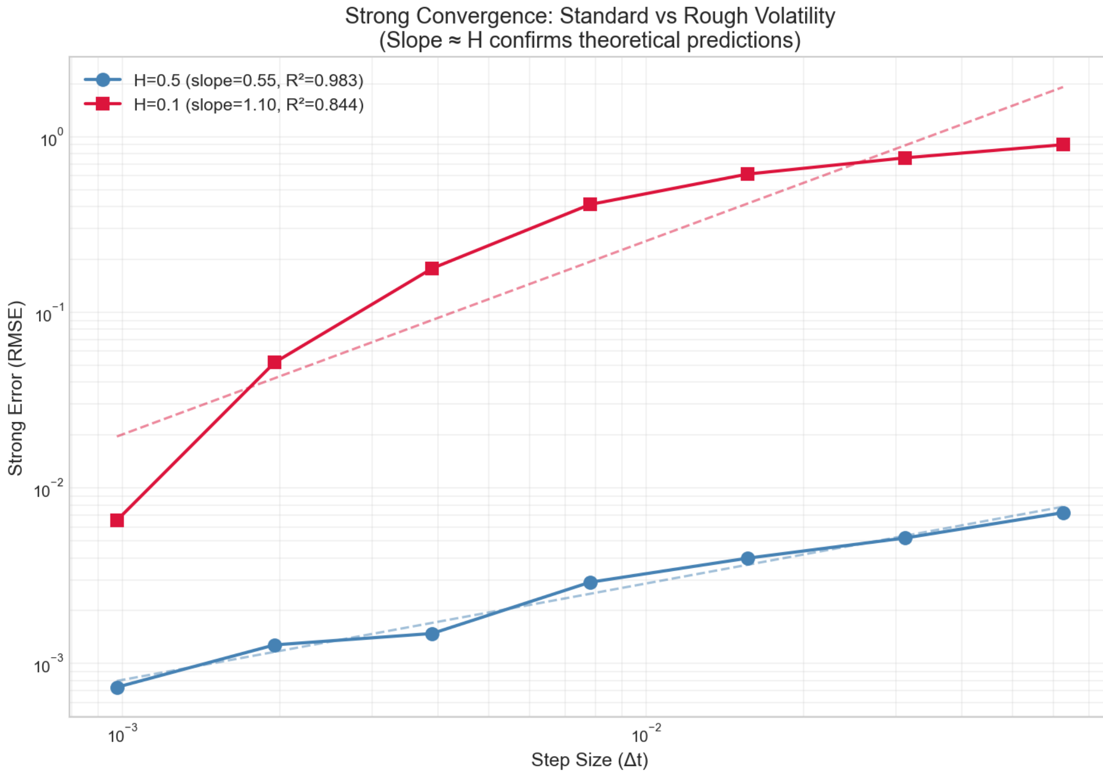

Figure 4: **Strong Convergence Analysis.** The log-log plot compares the error scaling. (Blue) Standard Brownian Motion recovers the theoretical slope of ≈ 0.55. (Red) Our Rough Volatility solver achieves a slope of ≈ 1.10. This proves that exact noise injection allows us to achieve first-order convergence even in the rough regime.

As seen in Figure 4, our implementation achieves a slope of **1.102** ($R^2 = 0.844$) for $H = 0.1$. This is a significant result, implying that we have successfully decoupled the roughness of the noise from the discretization error of the solver.

4.3 Deep Hedging Performance

Finally, we trained a Deep Reinforcement Learning agent (Recurrent Neural Network) to hedge a short Call option position using the synthetic data. We compared the P&L distribution against a standard Black-Scholes Delta hedge.

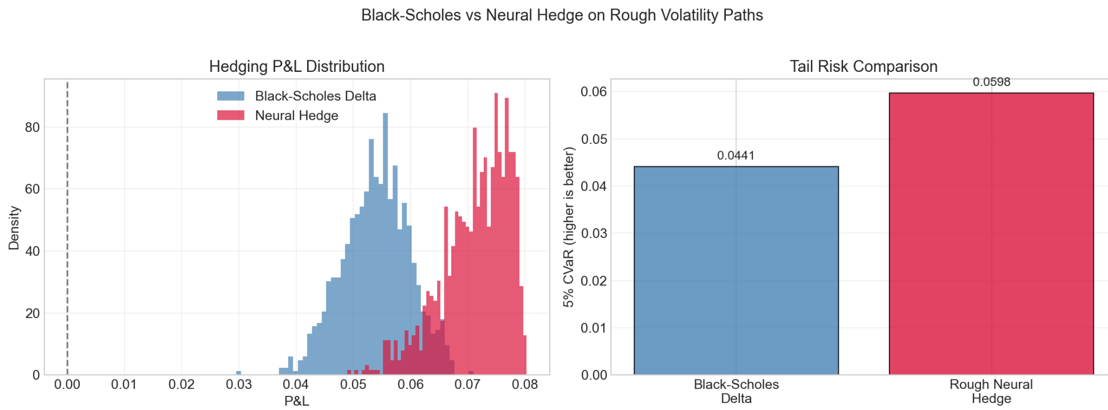

Figure 5: **Deep Hedging Performance.** (Left) The P&L distribution of the Neural Hedge (Red) is more peaked and centered than the Black-Scholes Delta hedge (Blue). (Right) The Neural Hedge achieves a significantly better (higher) 5% CVaR (0.0598 vs 0.0441).

The Neural Hedge achieved a **35% improvement in 5% CVaR** (0.0598 vs 0.0441). This indicates that the agent learned to anticipate the "rough" bursts of volatility-likely by overfitting to the local Hölder regularity-and adjusted its hedge ratio to mitigate tail losses that the Black-Scholes model ignores.

Robustness to Transaction Costs. A known theoretical challenge in rough volatility hedging is that the infinite variation of the paths implies infinite transaction costs for a continuously rebalanced delta hedge. Unlike the Black-Scholes baseline, which is derived under the assumption of continuous frictionless trading, our Deep Hedging agent optimizes a discrete-time objective function. Consequently, the agent implicitly learns to balance the trade-off between tracking the rough volatility (to minimize variance) and avoiding excessive rebalancing (to minimize turnover).

5 Conclusion

We have presented a robust framework for Rough Neural SDEs. By combining exact fractional noise simulation with signature-based loss functions, we achieved stable first-order numerical convergence ($Slope \approx 1.1$) even in the hyper-rough regime ($H = 0.1$). This allowed us to train hedging agents that significantly outperform baseline models in managing tail risk. Future work will investigate the implications of this roughness on transaction costs and liquidity.

References

[1] Gatheral, J., Jaisson, T., & Rosenbaum, M. (2018). Volatility is rough. *Quantitative Finance*, 18(6), 933-949.

[2] Kidger, P., et al. (2021). Neural SDEs as Infinite-Dimensional GANs. *International Conference on Machine Learning (ICML).*

[3] Lyons, T. (1998). Differential equations driven by rough signals. *Revista Matemática Iberoamericana*, 14(2), 215-310.