1 Problem setting

1.1 The rBergomi model

We consider the rBergomi model for the price process S_t as defined in [1], normalized to $r = 0$[1], which is defined by

$$\begin{aligned} dS_t &= \sqrt{v_t} S_t dZ_t, \\ v_t &= \xi_0(t) \exp\left(\eta \widetilde{W}_t^H - \frac{1}{2}\eta^2 t^{2H} \right), \end{aligned} \tag{1.1}$$

where the Hurst parameter $0 < H < 1$ and $\eta > 0$. We refer to v_t as the variance process, and $\xi_0(t) = \mathrm{E}[v_t]$ is the forward variance curve. Here, $\widetilde{W}^H$ is a certain Riemann-Liouville fBm process[2], defined by

$$\widetilde{W}_t^H = \int_0^t K^H(t,s) dW_s^1, \quad t \geq 0, \tag{1.2}$$

where the kernel $K^H : \mathbb{R}_+ \to \mathbb{R}_+$ is

$$K^H(t-s) = \sqrt{2H}(t-s)^{H-1/2}, \quad \forall\, 0 \leq s \leq t. \tag{1.3}$$

By construction, $\widetilde{W}^H$ is a centered, locally $(H-\epsilon)$- Hölder continuous, Gaussian process with $\mathrm{Var}\left[\widetilde{W}_t^H\right] = t^{2H}$, and a dependence structure defined by

$$\mathrm{E}\left[\widetilde{W}_u^H \widetilde{W}_v^H\right] = u^{2H} G\left(\frac{v}{u}\right), \quad v > u,$$

where for $x \geq 1$ and $\gamma = \frac{1}{2} - H$

$$G(x) = 2H \int_0^1 \frac{ds}{(1-s)^\gamma (x-s)^\gamma}. \tag{1.4}$$

In (1.1) and (1.2), W^1, Z denote two *correlated* standard Brownian motions with correlation $\rho \in]-1, 0]$, so that we can represent Z in terms of W^1 as

$$Z = \rho W^1 + \overline{\rho} W^\perp = \rho W^1 + \sqrt{1-\rho^2} W^\perp,$$

where $(W^1, W^\perp)$ are two independent standard Brownian motions. Therefore, the solution to (1.1), with $S(0) = S_0$, can be written as

$$\begin{aligned} S_t &= S_0 \exp\left(\int_0^t \sqrt{v(s)} dZ(s) - \frac{1}{2} \int_0^t v(s) ds \right), \quad S_0 > 0 \\ v_u &= \xi_0(u) \exp\left(\eta \widetilde{W}_u^H - \frac{\eta^2}{2} u^{2H} \right), \quad \xi_0 > 0. \end{aligned} \tag{1.5}$$

[1] r is the interest rate.

[2] The so-called Riemann-Liouville processes are deduced from the standard Brownian motion by applying Riemann-Liouville fractional operators, whereas the standard fBm requires a weighted fractional operator [?, ?].

The filtration $(\mathcal{F}_t)_{t\geq 0}$ can here be taken as the one generated by the two-dimensional Brownian motion $(W^1, W^{\perp})$ under the risk neutral measure $\mathbb{Q}$, resulting in a filtered probability space $(\Omega, \mathcal{F}, \mathcal{F}_t, \mathbb{Q})$. The stock price process S is clearly then a local $(\mathcal{F}_t)_{t\geq 0}$-martingale and a supermartingale. We shall henceforth use the notation $\mathrm{E}[.] = E^{\mathbb{Q}}[. \mid \mathcal{F}_0]$ unless we state otherwise.

Remark 1.1. The rBergomi model is non-Markovian in the instantaneous variance v_t, that is $E^{\mathbb{Q}}[v_u \mid \mathcal{F}_t] \neq= E^{\mathbb{Q}}[v_u \mid v_t]$. However, it is Markovian in the state vector by definition, that is $E^{\mathbb{Q}}[v_u \mid \mathcal{F}_t] = \xi_t(u)$.

1.2 Option pricing under the rBergomi model

We are interested in pricing European call options under the rBergomi model. Assuming $S_0 = 1$, and using the conditioning argument on the σ-algebra generated by W^1 (an argument first used by [4] in the context of Markovian stochastic volatility models), we can show that the call price is given by

$$\begin{aligned}
C_{\mathrm{RB}}(T,K) &= \mathrm{E}\left[(S_T-K)^+\right] \\
&= \mathrm{E}\left[\mathrm{E}\left[(S_T-K)^+ \mid \sigma(W^1(t), t\leq T)\right]\right] \\
&= \mathrm{E}\left[C_{\mathrm{BS}}\left(S_0 = \exp\left(\rho\int_0^T \sqrt{v_t}dW_t^1 - \frac{1}{2}\rho^2\int_0^T v_t dt\right),\ k=K,\ \sigma^2 = (1-\rho^2)\int_0^T v_t dt\right)\right], \qquad (1.6)
\end{aligned}$$

where $C_{\mathrm{BS}}(S_0, k, \sigma^2)$ denotes the Black-Scholes call price, for initial spot price S_0, strike price k and volatility σ^2.

We point out that the analytical smoothing, based on conditioning, performed in (1.6) enables us to uncover the available regularity, and hence get a smooth, analytic integrand inside the expectation. Therefore, applying a deterministic quadrature technique such as ASGQ or QMC becomes an adequate option for computing the call price as we will investigate later. A similar conditioning was used in [**?**] but for variance reduction purposes only.

2 Details of our hierarchical methods

We recall that our goal is to compute the expectation in (1.6). We need $2N$-dimensional Gaussian inputs for the used hybrid scheme (N is the number of time steps in the time grid), namely

- $\mathbf{W}^{(1)} = \{W_i^{(1)}\}_{i=1}^N$: The N Gaussian random variables that are defined in Section 1.1.
- $\mathbf{W}^{(2)} = \{W_j^{(2)}\}_{j=1}^N$: An artificially introduced N Gaussian random variables that are used for left-rule points in the hybrid scheme.

We can rewrite (1.6) as

$$\begin{aligned}
C_{\mathrm{RB}}(T,K) &= \mathrm{E}\left[C_{\mathrm{BS}}\left(S_0 = \exp\left(\rho\int_0^T \sqrt{v_t}dW_t^1 - \frac{1}{2}\rho^2\int_0^T v_t dt\right),\ k=K,\ \sigma^2 = (1-\rho^2)\int_0^T v_t dt\right)\right] \\
&\approx \int_{\mathbb{R}^{2N}} C_{BS}\left(G(\mathbf{w}^{(1)}, \mathbf{w}^{(2)})\right)\rho_N(\mathbf{w}^{(1)})\rho_N(\mathbf{w}^{(2)})d\mathbf{w}^{(1)}d\mathbf{w}^{(2)} \\
&:= C_{RB}^N, \qquad (2.1)
\end{aligned}$$

where G maps $2N$ independent standard Gaussian random inputs to the parameters fed to Black-Scholes formula, and ρ_N is the multivariate Gaussian density, given by

$$\rho_N(\mathbf{z}) = \frac{1}{(2\pi)^{N/2}} e^{-\frac{1}{2}\mathbf{z}^T\mathbf{z}}.$$

Therefore, the initial integration problem that we are solving lives in $2N$-dimensional space, which becomes very large as the number of time steps N, used in the hybrid scheme, increases.

If we denote by $\mathcal{E}_{\text{tot}}$ the total error of approximating the expectation in (1.6) using the ASGQ estimator, Q_N, then we have a natural error decomposition

$$\mathcal{E}_{\text{tot}} \leq \left|C_{\text{RB}} - C_{\text{RB}}^N\right| + \left|C_{\text{RB}}^N - Q_N\right| \leq \mathcal{E}_B(N) + \mathcal{E}_Q(\text{TOL}_{\text{ASGQ}}, N), \tag{2.2}$$

where $\mathcal{E}_Q$ is the quadrature error, $\mathcal{E}_B$ is the bias, TOL_{ASGQ} is a user selected tolerance for ASGQ method, and C_{RB}^N is the biased price computed with N time steps as given by (2.1).

On the other hand, the total error of approximating the expectation in (1.6) using the randomized QMC or MC estimator, $Q_N^{\text{MC(QMC)}}$ can be bounded by

$$\mathcal{E}_{\text{tot}} \leq \left|C_{\text{RB}} - C_{\text{RB}}^N\right| + \left|C_{\text{RB}}^N - Q_N^{\text{MC (QMC)}}\right| \leq \mathcal{E}_B(N) + \mathcal{E}_S(M, N), \tag{2.3}$$

where $\mathcal{E}_S$ is the statistical error[3], M is the number of samples used for MC or randomized QMC method.

2.1 Adaptive sparse grids quadrature (ASGQ)

We assume that we want to approximate the expected value $\text{E}[f(Y)]$ of an analytic function $f\colon \Gamma \to \mathbb{R}$ using a tensorization of quadrature formulas over Γ.

To introduce simplified notations, we start with the one-dimensional case. Let us denote by β a non-negative integer, referred to as a "stochastic discretization level", and by $m : \mathbb{N} \to \mathbb{N}$ a strictly increasing function with $m(0) = 0$ and $m(1) = 1$, that we call "level-to-nodes function". At level β, we consider a set of $m(\beta)$ distinct quadrature points in $\mathbb{R}$, $\mathcal{H}^{m(\beta)} = \{y_\beta^1, y_\beta^2, \ldots, y_\beta^{m(\beta)}\} \subset \mathbb{R}$, and a set of quadrature weights, $\boldsymbol{\omega}^{m(\beta)} = \{\omega_\beta^1, \omega_\beta^2, \ldots, \omega_\beta^{m(\beta)}\}$. We also let $C^0(\mathbb{R})$ be the set of real-valued continuous functions over $\mathbb{R}$. We then define the quadrature operator as

$$Q^{m(\beta)} : C^0(\mathbb{R}) \to \mathbb{R}, \quad Q^{m(\beta)}[f] = \sum_{j=1}^{m(\beta)} f(y_\beta^j)\omega_\beta^j.$$

In our case, we have in (2.1) a multi-variate integration problem with, $f = C_{\text{BS}} \circ G$, $\mathbf{Y} = (\mathbf{W}^{(1)}, \mathbf{W}^{(2)})$, and $\Gamma = \mathbb{R}^{2N}$, in the previous notations. Furthermore, since we are dealing with Gaussian densities, using Gauss-Hermite quadrature points is the appropriate choice.

We define for any multi-index $\boldsymbol{\beta} \in \mathbb{N}^{2N}$

$$Q^{m(\boldsymbol{\beta})} : C^0(\mathbb{R}^{2N}) \to \mathbb{R}, \quad Q^{m(\boldsymbol{\beta})} = \bigotimes_{n=1}^{2N} Q^{m(\beta_n)},$$

[3]The statistical error estimate of MC or randomized QMC is $C_\alpha \frac{\sigma_M}{\sqrt{M}}$, where M is the number of samples and $C_\alpha = 1.96$ for 95% confidence interval.

where the n-th quadrature operator is understood to act only on the n-th variable of f. Practically, we obtain the value of $Q^{m(\boldsymbol{\beta})}[f]$ by using the grid $\mathcal{T}^{m(\boldsymbol{\beta})} = \prod_{n=1}^{2N} \mathcal{H}^{m(\beta_n)}$, with cardinality $\#\mathcal{T}^{m(\boldsymbol{\beta})} = \prod_{n=1}^{2N} m(\beta_n)$, and computing

$$Q^{m(\boldsymbol{\beta})}[f] = \sum_{j=1}^{\#\mathcal{T}^{m(\boldsymbol{\beta})}} f(\widehat{y}_j)\overline{\omega}_j,$$

where $\widehat{y}_j \in \mathcal{T}^{m(\boldsymbol{\beta})}$ and $\overline{\omega}_j$ are products of weights of the univariate quadrature rules. To simplify notation, hereafter, we replace $Q^{m(\boldsymbol{\beta})}$ by $Q^{\boldsymbol{\beta}}$.

A direct approximation $\mathrm{E}\left[f[\mathbf{Y}]\right] \approx Q^{\boldsymbol{\beta}}[f]$ is not an appropriate option due to the well-known "curse of dimensionality". We use a hierarchical ASGQ[4] strategy, specifically using the same construction as in [3], and which uses stochastic discretizations and a classic sparsification approach to obtain an effective approximation scheme for $\mathrm{E}\left[f\right]$.

To be concrete, in our setting, we are left with a $2N$-dimensional Gaussian random input, which is chosen independently, resulting in $2N$ numerical parameters for ASGQ, which we use as the basis of the multi-index construction. For a multi-index $\boldsymbol{\beta} = (\beta_n)_{n=1}^{2N} \in \mathbb{N}^{2N}$, we denote by $Q_N^{\boldsymbol{\beta}}$, the result of approximating (2.1) with a number of quadrature points in the i-th dimension equal to $m(\beta_i)$. We further define the set of differences $\Delta Q_N^{\boldsymbol{\beta}}$ as follows: for a single index $1 \leq i \leq 2N$, let

$$\Delta_i Q_N^{\boldsymbol{\beta}} = \begin{cases} Q_N^{\boldsymbol{\beta}} - Q_N^{\boldsymbol{\beta}'}, \text{ with } \boldsymbol{\beta}' = \boldsymbol{\beta} - e_i, \text{ if } \boldsymbol{\beta}_i > 0, \\ Q_N^{\boldsymbol{\beta}}, \quad \text{otherwise,} \end{cases}$$

where e_i denotes the ith $2N$-dimensional unit vector. Then, $\Delta Q_N^{\boldsymbol{\beta}}$ is defined as

$$\Delta Q_N^{\boldsymbol{\beta}} = \left(\prod_{i=1}^{2N} \Delta_i\right) Q_N^{\boldsymbol{\beta}}.$$

For instance, when $N = 1$, then

$$\begin{aligned} \Delta Q_1^{\boldsymbol{\beta}} = \Delta_2 \Delta_1 Q_1^{(\beta_1,\beta_2)} = \Delta_2 \left(Q_1^{(\beta_1,\beta_2)} - Q_1^{(\beta_1-1,\beta_2)}\right) &= \Delta_2 Q_1^{(\beta_1,\beta_2)} - \Delta_2 Q_1^{(\beta_1-1,\beta_2)} \\ &= Q_1^{(\beta_1,\beta_2)} - Q_1^{(\beta_1,\beta_2-1)} - Q_1^{(\beta_1-1,\beta_2)} + Q_1^{(\beta_1-1,\beta_2-1)}. \end{aligned}$$

Given the definition of C_{RB}^N by (2.1), we have the telescoping property

$$C_{RB}^N = Q_N^{\infty} = \sum_{\beta_1=0}^{\infty} \cdots \sum_{\beta_{2N}=0}^{\infty} \Delta Q_N^{(\beta_1,\ldots,\beta_{2N})} = \sum_{\boldsymbol{\beta} \in \mathbb{N}^{2N}} \Delta Q_N^{\boldsymbol{\beta}}.$$

The ASGQ estimator used for approximating (2.1), and using a set of multi-indices $\mathcal{I} \subset \mathbb{N}^{2N}$ is given by

$$Q_N^{\mathcal{I}} = \sum_{\boldsymbol{\beta} \in \mathcal{I}} \Delta Q_N^{\boldsymbol{\beta}}. \tag{2.4}$$

[4] More details about sparse grids can be found in [2].

The quadrature error in this case is given by

$$\mathcal{E}_Q(\mathrm{TOL}_{\mathrm{ASGQ}}, N) = \left|Q_N^{\infty} - Q_N^{\mathcal{I}}\right| \leq \sum_{\boldsymbol{\beta} \in \mathbb{N}^{2N} \setminus \mathcal{I}} \left|\Delta Q_N^{\boldsymbol{\beta}}\right|. \tag{2.5}$$

We define the work contribution, $\Delta \mathcal{W}_{\boldsymbol{\beta}}$, to be the computational cost required to add $\Delta Q_N^{\boldsymbol{\beta}}$ to $Q_N^{\mathcal{I}}$, and the error contribution, $\Delta E_{\boldsymbol{\beta}}$, to be a measure of how much the quadrature error, defined in (2.5), would decrease once $\Delta Q_N^{\boldsymbol{\beta}}$ has been added to $Q_N^{\mathcal{I}}$, that is

$$\Delta \mathcal{W}_{\boldsymbol{\beta}} = \mathrm{Work}[Q_N^{\mathcal{I} \cup \{\boldsymbol{\beta}\}}] - \mathrm{Work}[Q_N^{\mathcal{I}}] \tag{2.6}$$

$$\Delta E_{\boldsymbol{\beta}} = \left|Q_N^{\mathcal{I} \cup \{\boldsymbol{\beta}\}} - Q_N^{\mathcal{I}}\right|. \tag{2.7}$$

The construction of the optimal $\mathcal{I}$ is done by profit thresholding, that is, for a certain threshold value $\overline{T}$, and a profit of a hierarchical surplus defined by

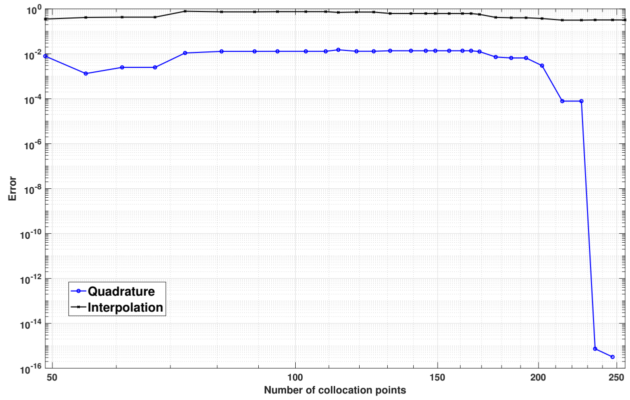
$$P_{\boldsymbol{\beta}} = \frac{|\Delta E_{\boldsymbol{\beta}}|}{\Delta \mathcal{W}_{\boldsymbol{\beta}}},$$

the optimal index set $\mathcal{I}$ for our ASGQ is given by $\mathcal{I} = \{\boldsymbol{\beta} : P_{\boldsymbol{\beta}} \geq \overline{T}\}$.

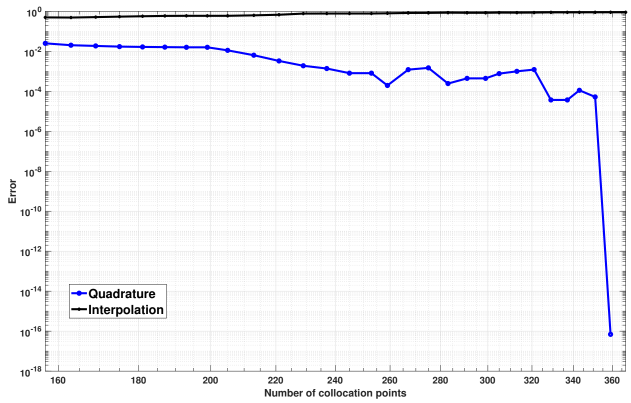
3 Numerical experiments of estimating quadrature error

In this section, I show the obtained quadrature estimates for one of my examples for $N = 2, 4, 8$ where N is the number of time steps. Looking at Figure 3.1, I have some questions that are not clear to me:

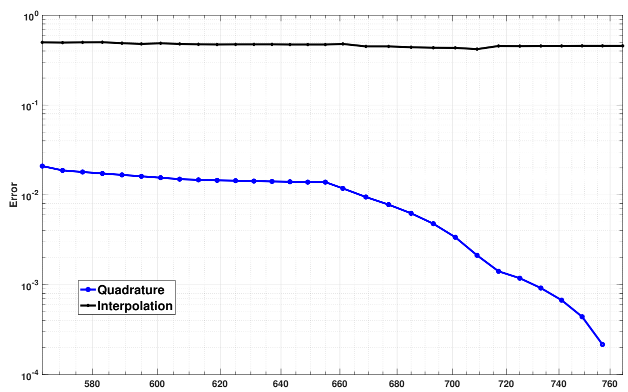
- I did not get why the quadrature error is kind of constant and then it decays so fast for very large number of collocation points. Is it normal to observe such behavior? Also, this behavior is less weird when taking $N = 8$.
- Why the interpolation error seems to be constant?
- Does it seem that using the interpolation error as a bound for the quadrature error is not that good?



(a)



(b)



(c)

Figure 3.1: Quadrature and interpolation error estimation for 3 different cases of number of time steps: a) $N = 2$, b) $N = 4$, c) $N = 8$

4 ASGQ error estimate

As discussed, potential ways of estimating the quadrature error are presented below.

4.1 First way

I want to check with you: does your code use a similar approach to the first way, right?

In our case, once we fix N, we define from (2.1)

$$F^N = C_{\mathrm{BS}}(G(\mathbf{W}^{(1)}, \mathbf{W}^{(2)})).$$

We introduce the set $C^0(\mathbb{R})$ of real-valued continuous functions over $\mathbb{R}$, and the subspace of polynomials of degree at most q over $\mathbb{R}$, $\mathbb{P}^q(\mathbb{R}) \subset C^0(\mathbb{R})$. Next, we consider a sequence of univariate Lagrangian interpolant operators in each dimension Y_n $(1 \leq n \leq 2N)$, that is, $\{U_n^{m(\beta_n)}\}_{\beta_n \in \mathbb{N}_+}$ (we refer to the value β_n as the interpolation level). Each interpolant is built over a set of $m(\beta_n)$ collocation points, $\mathcal{H}^{m(\beta_n)} = \{y_n^1, y_n^2, \dots, y_n^{m(\beta_n)}\} \subset \mathbb{R}$, thus, the interpolant yields a polynomial approximation,

$$U^{m(\beta_n)} : C^0(\mathbb{R}) \to \mathbb{P}^{m(\beta_n)-1}(\mathbb{R}), \quad U^{m(\beta_n)}[F^N](y_n) = \sum_{j=1}^{m(\beta_n)} \left(f(y_n^j) \prod_{k=1;k\neq j}^{m(\beta_n)} \frac{y_n - y_n^k}{y_n^j - y_n^k} \right).$$

The $2N$-variate Lagrangian interpolant can then be built by a tensorization of univariate interpolants: denote by $C^0(\mathbb{R}^{2N})$ the space of real-valued $2N$-variate continuous functions over $\mathbb{R}^{2N}$ and by $\mathbb{P}^{\mathbf{q}}(\mathbb{R}^{2N}) = \otimes_{n=1}^{2N} \mathbb{P}^{\mathbf{q}_n}(\mathbb{R})$ the subspace of polynomials of degree at most q_n over $\mathbb{R}$, with $\mathbf{q} = (q_1, \dots, q_{2N}) \in \mathbb{N}^{2N}$, and consider a multi-index $\boldsymbol{\beta} \in \mathbb{N}_+^{2N}$ assigning the interpolation level in each direction, y_n, then the multivariate interpolant can then be written as

$$U^{m(\boldsymbol{\beta})} : C^0(\mathbb{R}^{2N}) \to \mathbb{P}^{m(\boldsymbol{\beta})-1}(\mathbb{R}^{2N}), \quad U^{m(\boldsymbol{\beta})}[F^N](\mathbf{y}) = \bigotimes_{n=1}^{2N} U^{m(\beta_n)}[F^N](\mathbf{y}),$$

Given this construction, we can define the ASGQ interpolant for approximating F^N, using a set of multi indices $\mathcal{I} \in \mathbb{N}^{2N}$ as

$$I^{\mathcal{I}}[F^N] = \sum_{\boldsymbol{\beta} \in \mathcal{I}} \Delta U_N^{\boldsymbol{\beta}}, \tag{4.1}$$

where

$$\Delta_i U_N^{\boldsymbol{\beta}} = \begin{cases} U_N^{\boldsymbol{\beta}} - U_N^{\boldsymbol{\beta}'}, \text{ with } \boldsymbol{\beta}' = \boldsymbol{\beta} - e_i, \text{ if } \boldsymbol{\beta}_i > 0 \\ U_N^{\boldsymbol{\beta}}, \quad \text{otherwise,} \end{cases}$$

where e_i denotes the ith $2N$-dimensional unit vector. Then, $\Delta U_N^{\boldsymbol{\beta}}$ is defined as

$$\Delta U_N^{\boldsymbol{\beta}} = \left(\prod_{i=1}^{2N} \Delta_i \right) U_N^{\boldsymbol{\beta}}.$$

We define the interpolation error induced by ASGQ as

$$e_N = F^N - I^{\mathcal{I}}[F^N]. \tag{4.2}$$

One can have a bound on the interpolation error of ASGQ, e_N, by tensorizing one dimensional error estimates, and then simply integrate that bound to get the ASGQ error, $\mathcal{E}_Q(\mathrm{TOL}_{\mathrm{ASGQ}}, N)$, defined in (2.2). However, we think that this will not lead to a sharp error estimate for ASGQ. Another strategy for estimating the ASGQ error, is to estimate $\mathrm{E}[e_N]$ using MC by sampling directly e_N. Finally, one can learn the error curve as a way to reduce the extra burden that comes from estimating the ASGQ error.

If we define $Y = F^N + (Q_N^{\mathcal{I}} - I^{\mathcal{I}}[F^N])$ (where $Q_N^{\mathcal{I}}$ is the ASGQ estimator, defined in (2.4)), then we have

$$\begin{aligned} \mathrm{E}[Y] &= \mathrm{E}\left[F^N\right] \\ Var[Y] &= Var[e_N] < Var[\mathcal{A}_{\mathrm{MC}}], \end{aligned} \tag{4.3}$$

where $\mathcal{A}_{\mathrm{MC}}$ is the MC estimator for $\mathrm{E}\left[F^N\right]$.

(4.3) shows that ASGQ can be seen as a control variate for MC estimator and consequently as a powerful variance reduction tool.

This way of estimating the quadrature error comes with the disadvantage of exciting the strong error which has a poor behavior in our context resulting maybe to having a non-sharp error estimate. Therefore, we suggest to use a second option as detailed in the following.

4.2 Second way

To avoid exciting the strong error when estimating the quadrature error and just act on the weak error, we can use a second way that is inspired of randomized QMC. In fact, we suggest to use a randomized version of ASGQ where the randomization involves randomized rotation and scaling for quadrature rules since we deal with unbounded domains and Hermite quadrature rule. Although this comes with the advantage of just acting on the weak error, it has the issue of reducing anisotropy which is a main feature for a good performance of ASGQ.

References Cited

[1] Christian Bayer, Peter Friz, and Jim Gatheral. Pricing under rough volatility. *Quantitative Finance*, 16(6):887–904, 2016.

[2] Hans-Joachim Bungartz and Michael Griebel. Sparse grids. *Acta numerica*, 13:147–269, 2004.

[3] Abdul-Lateef Haji-Ali, Fabio Nobile, Lorenzo Tamellini, and Raul Tempone. Multi-index stochastic collocation for random pdes. *Computer Methods in Applied Mechanics and Engineering*, 306:95–122, 2016.

[4] Marc Romano and Nizar Touzi. Contingent claims and market completeness in a stochastic volatility model. *Mathematical Finance*, 7(4):399–412, 1997.