Weierstraß-Institut
für Angewandte Analysis und Stochastik

Leibniz-Institut im Forschungsverbund Berlin e. V.

Preprint ISSN 2198-5855

Numerical smoothing with hierarchical adaptive sparse grids and quasi-Monte Carlo methods for efficient option pricing

Christian Bayer[1], Chiheb Ben Hammouda[2], Raúl F. Tempone[2,3]

submitted: January 26, 2022

[1] Weierstrass Institute
Mohrenstr. 39
10117 Berlin
Germany
E-Mail: christian.bayer@wias-berlin.de

[2] RWTH Aachen
Departement for Mathematics
Kackertstraße 9
52072 Aachen
Germany
E-Mail: benhammouda@uq.rwth-aachen.de
tempone@uq.rwth-aachen.de

[3] KAUST
Computer, Electrical and Mathematical
Sciences & Engineering Division (CEMSE)
Thuwal $23955 - 6900$
Saudi Arabia
E-Mail: raul.tempone@kaust.edu.sa

No. 2917
Berlin 2022

2020 *Mathematics Subject Classification.* 65C05, 65D30, 65D32, 65Y20, 91G20, 91G60.

Key words and phrases. Adaptive sparse grid quadrature, quasi-Monte Carlo, numerical smoothing, Brownian bridge, Richardson extrapolation, option pricing, Monte Carlo.

C. Bayer gratefully acknowledges support from the German Research Foundation (DFG) via the Cluster of Excellence MATH+ (project AA4-2) and the individual grant BA5484/1. This publication is based on the work supported by the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research (OSR) under Award No. OSR-2019-CRG8-4033 and the Alexander von Humboldt Foundation.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Leibniz-Institut im Forschungsverbund Berlin e. V.
Mohrenstraße 39
10117 Berlin
Germany

Fax: +49 30 20372-303
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Numerical smoothing with hierarchical adaptive sparse grids and quasi-Monte Carlo methods for efficient option pricing

Christian Bayer, Chiheb Ben Hammouda, Raúl F. Tempone

Abstract

When approximating the expectation of a functional of a stochastic process, the efficiency and performance of deterministic quadrature methods, such as sparse grid quadrature and quasi-Monte Carlo (QMC) methods, may critically depend on the regularity of the integrand. To overcome this issue and reveal the available regularity, we consider cases in which analytic smoothing cannot be performed, and introduce a novel numerical smoothing approach by combining a root finding algorithm with one-dimensional integration with respect to a single well-selected variable. We prove that under appropriate conditions, the resulting function of the remaining variables is a highly smooth function, potentially affording the improved efficiency of adaptive sparse grid quadrature (ASGQ) and QMC methods, particularly when combined with hierarchical transformations (*i.e.*, Brownian bridge and Richardson extrapolation on the weak error). This approach facilitates the effective treatment of high dimensionality. Our study is motivated by option pricing problems, and our focus is on dynamics where the discretization of the asset price is necessary. Based on our analysis and numerical experiments, we show the advantages of combining numerical smoothing with the ASGQ and QMC methods over ASGQ and QMC methods without smoothing and the Monte Carlo approach.

1 Introduction

In many applications in quantitative finance, one is usually interested in efficiently computing the expectation of a functional, g, of a solution of a stochastic differential equation (SDE), X:

$$\mathrm{E}\left[g(X)\right]. \tag{1.1}$$

Approximating (1.1) is usually challenging because of a combination of two complications:

1. An input space shows high dimensionality because of many reasons including (i) the time discretization of a SDE that describes the dynamics or (ii) numerous underlying assets.
2. The payoff function, g, exhibits low regularity with respect to (w.r.t.) the input parameters.

The first class of methods for approximating (1.1) relies on Monte Carlo (MC) methods. Although the convergence rate of the standard MC is insensitive to both the input space dimensionality and the regularity of the observable g, the convergence may be very slow. Moreover, it may not exploit the available regularity structure that could help achieve better convergence rates. Another class of methods relies on deterministic quadrature methods (e.g., sparse grid quadrature [25, 12, 5, 10], adaptive sparse grid quadrature (ASGQ) [6, 8], and quasi-MC (QMC) [22, 7]). In this work, we introduce a numerical smoothing technique for improving the performance of these approaches, by revealing the available regularity.

The high dimensionality of the input space and existence of discontinuities[1] in the integrand considerably degrade the performance of deterministic quadrature methods. Some studies [14, 15, 16, 7, 27] have addressed cases involving integrands with discontinuities; however, the emphasis was on the QMC method. In particular, [14, 15, 16] focused on the theoretical aspects of employing the QMC method in such a setting. Alternatively, in the present study, we focus on specific practical problems, where we include the adaptivity paradigm. Moreover,

[1] We consider discontinuities either in the gradients (kinks) or in the function (jumps).

DOI 10.20347/WIAS.PREPRINT.2917 Berlin 2022

the low regularity of the integrand was addressed by performing analytic smoothing using conditional expectation approaches [7, 27, 6]. An adaptive version of the QMC method combined with geometric random splitting was employed for pricing multidimensional vanilla options for the Black-Scholes model [11].

In this work, we consider cases in which analytic smoothing cannot be performed. We introduce a novel numerical smoothing technique based on (i) the identification of the discontinuities locations using root finding algorithms, (ii) employing suitable transformations of the integration domain, and (iii) a preintegration step w.r.t. the dimension containing the discontinuities. We prove that under appropriate conditions, the resulting function of the remaining variables is a highly smooth function. This potentially affords improved efficiency of the ASGQ and QMC methods, particularly when they are combined with hierarchical transformations to treat the high dimensionality effectively. Given that ASGQ and QMC methods benefit from anisotropy, the first technique involves employing a hierarchical path generation method based on the Brownian bridge construction for reducing the effective dimension. The second technique involves employing the Richardson extrapolation for reducing the bias (weak error), which subsequently reduces the number of time steps needed at the coarsest level to achieve a certain error tolerance and consequently decreases the total number of dimensions needed for the integration problem. Our analysis and numerical experiments show the advantage of our approach, which substantially outperforms the ASGQ and QMC methods without smoothing and the MC approach, for high-dimensional examples and for dynamics where discretization is needed such as the Heston model.

In Section 2, we explain the technique of numerical smoothing, the selection of the optimal smoothing direction, and the different building blocks that constitute our hierarchical quadrature methods. In Section 3.1, we present the smoothness analysis of the resulting integrand after numerical smoothing. In Section 3.2, an error and work discussion is presented for the ASGQ method with numerical smoothing. Finally, in Section 4, we report the results of different numerical experiments conducted using the ASGQ, QMC, and MC methods. These results verify the considerable computational gains achieved using the ASGQ and QMC methods (both combined with numerical smoothing) over the MC method and the standard (without smoothing) ASGQ and QMC methods.

2 Problem Setting and Approach Formulation

In this work, we mainly consider two possible structures of the payoff function g:

$$\text{(2.1)} \qquad \text{(i) } g(\mathbf{x}) = \max(\phi(\mathbf{x}), 0); \text{ (ii) } g(\mathbf{x}) = \mathbf{1}_{(\phi(\mathbf{x})\geq 0)},$$

where the function ϕ is supposed to be smooth.

We introduce the notation $\mathbf{x}_{-j}$ to denote a vector with length $d-1$ that represents all the variables other than x_j in $\mathbf{x}$. Abusing notation, we define $\phi(\mathbf{x}) = \phi(x_j, \mathbf{x}_{-j})$, and for ease of presentation, we assume that for fixed $\mathbf{x}_{-j}$, the function $\phi(x_j, \mathbf{x}_{-j})$ either has a simple root or is positive for all $x_j \in \mathbb{R}$. This is guaranteed by the monotonicity condition (2.2) and infinite growth condition (2.3), which are assumed for some $j \in \{1, \dots, d\}$.

$$\text{(2.2)} \qquad \frac{\partial \phi}{\partial x_j}(\mathbf{x}) > 0, \ \forall \mathbf{x} \in \mathbb{R}^d \ \textbf{(Monotonicity condition)}^2$$

$$\text{(2.3)} \qquad \lim_{x_j \to +\infty} \phi(\mathbf{x}) = \lim_{x_j \to +\infty} \phi(x_j, \mathbf{x}_{-j}) = +\infty, \ \forall \mathbf{x}_{-j} \in \mathbb{R}^{d-1} \text{ or } \frac{\partial^2 \phi}{\partial x_j^2}(\mathbf{x}) \geq 0, \ \forall \mathbf{x} \in \mathbb{R}^d \ \textbf{(Growth condition)}.$$

Note that our approach can be easily extended to the case of finetely many roots without accumulation. We explain this extension in Remark 2.4.

2.1 Continuous-time formulation and optimal smoothing direction

In this section, we characterize the optimal smoothing direction using the continuous-time formulation. The purpose of this work is to approximate $\mathrm{E}\left[g(\mathbf{X}_T)\right]$ at final time T, where g is a low-regular payoff function and

[2] Without loss of generality, we show the monotonicity condition for an increasing function. However, the assumption still holds for a decreasing function, which may be the case when considering a spread option.

$\mathbf{X} := (X^{(1)}, \ldots, X^{(d)})$ is described using the following SDE[3]

$$dX_t^{(i)} = a_i(\mathbf{X}_t)dt + \sum_{j=1}^{d} b_{ij}(\mathbf{X}_t)dW_t^{(j)}. \tag{2.4}$$

First, we hierarchically represent $\mathbf{W} := (W^{(1)}, \ldots, W^{(d)})$

$$W^{(j)}(t) = \frac{t}{T}W^{(j)}(T) + B^{(j)}(t) = \frac{t}{\sqrt{T}}Z_j + B^{(j)}(t),\ 1 \leq j \leq d, \tag{2.5}$$

where $\{Z_j\}_{j=1}^d$ are independent and identically distributed (i.i.d.) standard Gaussian random variables (rdvs), and $\{B^{(j)}\}_{j=1}^d$ are independent Brownian bridges.

We can hierarchically represent $\mathbf{Z} := (Z_1, \ldots, Z_d)$ as

$$\mathbf{Z} = \underbrace{P_0\mathbf{Z}}_{\text{One dimensional projection}} + \underbrace{P_\perp\mathbf{Z}}_{\text{Projection on the complementary}},$$

where[4] $P_0\mathbf{Z} := (\mathbf{Z}, \mathbf{v})\mathbf{v}$, with $||\mathbf{v}|| = 1$, and $Z_v := (\mathbf{Z}, \mathbf{v})$ is a standard Gaussian rdv.

Furthermore, defining $\mathbf{w} := \mathbf{Z} - Z_v\mathbf{v}$ yields

$$Z_j = Z_v v_j + (P_\perp \mathbf{Z})_j = Z_v v_j + w_j,\ 1 \leq j \leq d. \tag{2.6}$$

Using (2.5) and (2.6) in (2.4) implies

$$dX_t^{(i)} = \left(a_i(\mathbf{X}_t) + \sum_{j=1}^{d} b_{ij}(\mathbf{X}_t)\frac{Z_v v_j}{\sqrt{T}}\right)dt + \left(\sum_{j=1}^{d} b_{ij}(\mathbf{X}_t)\frac{w_j}{\sqrt{T}}\right)dt + \sum_{j=1}^{d} b_{ij}(\mathbf{X}_t)dB_t^{(j)}. \tag{2.7}$$

If we define $H_{\mathbf{v}}(Z_v, \mathbf{w}) := g(\mathbf{X}(T))$, then (2.6) and (2.7) can be used to yield

$$\begin{aligned} \mathrm{E}[g(\mathbf{X}(T))] &= \mathrm{E}[\mathrm{E}[H_{\mathbf{v}}(Z_v, \mathbf{w}) \mid \mathbf{w}]] \\ \mathrm{Var}[g(\mathbf{X}(T))] &= \mathrm{E}[\mathrm{Var}[H_{\mathbf{v}}(Z_v, \mathbf{w}) \mid \mathbf{w}]] + \mathrm{Var}[\mathrm{E}[H_{\mathbf{v}}(Z_v, \mathbf{w}) \mid \mathbf{w}]]. \end{aligned} \tag{2.8}$$

Using (2.8), the optimal smoothing direction is characterized as the one that maximizes the smoothing effect at T. The smoothing effect essentially refers to the variance of the component orthogonal to the discontinuity. This implies that $\mathbf{v}$ solves the following equivalent optimization problem:

$$\max_{\substack{\mathbf{v}\in\mathbb{R}^d \\ ||\mathbf{v}||=1}} \mathrm{E}[\mathrm{Var}[H_{\mathbf{v}}(Z_v, \mathbf{w}) \mid \mathbf{w}]] \iff \min_{\substack{\mathbf{v}\in\mathbb{R}^d \\ ||\mathbf{v}||=1}} \mathrm{Var}[\mathrm{E}[H_{\mathbf{v}}(Z_v, \mathbf{w}) \mid \mathbf{w}]]. \tag{2.9}$$

Solving (2.9) is difficult, and $\mathbf{v}$ is dependent on the problem. In this work, we aim to heuristically determine $\mathbf{v}$ by considering the structure of the problem. In the following section, we provide more insights on selecting $\mathbf{v}$ and performing numerical smoothing in the time-stepping setting.

2.2 Motivation and idea of numerical smoothing

Let us consider $\mathbf{X}$ to be the solution of the SDE (2.4). To illustrate our numerical smoothing idea, we consider, for ease of presentation, the discretized d-dimensional geometric Brownian motion (GBM) model given by[5]

$$dX_t^{(j)} = \sigma^{(j)}X_t^{(j)}dW_t^{(j)}, \quad 1 \leq j \leq d, \tag{2.10}$$

where $\{W^{(1)}, \ldots, W^{(d)}\}$ are correlated Brownian motions with correlations ρ_{ij}, and $\{\sigma^{(j)}\}_{j=1}^d$ denote the volatilities of the different assets.

[3] We assume that $\{W^{(j)}\}_{j=1}^d$ are uncorrelated and the correlation terms are included in the diffusion terms b_{ij}.

[4] We use $(.,.)$ to denote the scalar product operator.

[5] For ease of presentation, we set the drift term in (2.10) to zero.

We denote by $(Z_1^{(j)}, \dots, Z_N^{(j)})$ the N standard Gaussian independent rdvs that will be used to construct the approximate path of the j-th asset $\overline{X}^{(j)}$, where N represents the number of time steps used in the discretization ($\Delta t = \frac{T}{N}$). $\psi^{(j)} : (Z_1^{(j)}, \dots, Z_N^{(j)}) \to (B_1^{(j)}, \dots, B_N^{(j)})$ denotes the mapping of the Brownian bridge construction, and $\Phi : (\Delta t, \mathbf{B}) \to \left(\overline{X}_T^{(1)}, \dots, \overline{X}_T^{(d)}\right)$ denotes the mapping of the time-stepping scheme, where $\mathbf{B} := \left(B_1^{(1)}, \dots, B_N^{(1)}, \dots, B_1^{(d)}, \dots, B_N^{(d)}\right)$ is the noncorrelated Brownian bridge[6]. Then, the option price can be expressed as

$$
\begin{aligned}
\mathrm{E}\left[g(\mathbf{X}(T))\right] &\approx \mathrm{E}\left[g\left(\overline{X}_T^{(1)}, \dots, \overline{X}_T^{(d)}\right)\right] = \mathrm{E}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right] \\
&= \mathrm{E}\left[g \circ \Phi\left(B_1^{(1)}, \dots, B_N^{(1)}, \dots, B_1^{(d)}, \dots, B_N^{(d)}\right)\right] \\
&= \mathrm{E}\left[g \circ \Phi\left(\psi^{(1)}(Z_1^{(1)}, \dots, Z_N^{(1)}), \dots, \psi^{(d)}(Z_1^{(d)}, \dots, Z_N^{(d)})\right)\right] \\
&= \int_{\mathbb{R}^{d\times N}} G(z_1^{(1)}, \dots, z_N^{(1)}, \dots, z_1^{(d)}, \dots, z_N^{(d)}))\rho_{d\times N}(\mathbf{z}) dz_1^{(1)} \dots dz_N^{(1)} \dots z_1^{(d)} \dots dz_N^{(d)}, \qquad (2.11)
\end{aligned}
$$

where $G := g \circ \Phi \circ \left(\psi^{(1)}, \dots, \psi^{(d)}\right)$ and $\rho_{d\times N}$ represents the $d \times N$ multivariate Gaussian density.

Moreover, the numerical approximation of $X^{(j)}(T)$, using the forward Euler scheme, satisfies

$$
\overline{X}^{(j)}(T) = X_0^{(j)} \prod_{n=0}^{N-1} \underbrace{\left[1 + \frac{\sigma^{(j)}}{\sqrt{T}} Z_1^{(j)} \Delta t + \sigma^{(j)} \Delta B_n^{(j)}\right]}_{:= f_n^{(j)}(Z_1^{(j)})}, \quad 1 \le j \le d. \qquad (2.12)
$$

Remark 2.1. Note that (2.12) holds even for stochastic volatility models, where $\sigma^{(j)}$ is a nonconstant and changes at each time step.

2.2.1 Step 1 of numerical smoothing: Root finding for the discontinuity location

In this step, the discontinuity location is determined by solving the corresponding root finding problem in one dimension after adopting suboptimal linear mapping for the coarsest factors of the Brownian increments $\mathbf{Z}_1 := (Z_1^{(1)}, \dots, Z_1^{(d)})$

$$
\mathbf{Y} = \mathcal{A}\mathbf{Z}_1, \qquad (2.13)
$$

where $\mathcal{A}$ is a $d \times d$ matrix that represents the linear mapping.

To make connection with Section 2.1, the smoothing direction $\mathbf{v}$ is expressed using the first row of $\mathcal{A}$. $\mathcal{A}$ is generally selected from a family of rotations. For instance, if we consider an arithmetic basket call option, a sufficiently good selection of $\mathcal{A}$ will be a rotation matrix, with the first row leading to $Y_1 = \sum_{i=1}^d Z_1^{(i)}$ up to rescaling without any constraint for the remaining rows. In practice, we construct $\mathcal{A}$ by fixing the first row as[7] $\frac{1}{\sqrt{d}}\mathbf{1}_{1\times d}$ and the remaining rows are obtained using the Gram-Schmidt procedure.

From (2.12) and using (2.13), we obtain

$$
\overline{X}^{(j)}(T) = X_0^{(j)} \prod_{n=0}^{N-1} f_n^{(j)}\left((\mathcal{A}^{-1}\mathbf{Y})_j\right) = X_0^{(j)} \prod_{n=0}^{N-1} F_n^{(j)}(Y_1, \mathbf{Y}_{-1}), \quad 1 \le j \le d,
$$

where by defining $\mathcal{A}^{\mathsf{inv}} := \mathcal{A}^{-1}$, we have

$$
F_n^{(j)}(Y_1, \mathbf{Y}_{-1}) = \left[1 + \frac{\sigma^{(j)}\Delta t}{\sqrt{T}} A_{j1}^{\mathsf{inv}} Y_1 + \frac{\sigma^{(j)}}{\sqrt{T}} \left(\sum_{i=2}^d A_{ji}^{\mathsf{inv}} Y_i\right) \Delta t + \sigma^{(j)} \Delta B_n^{(j)}\right].
$$

[6] Without loss of generality, the correlated Brownian bridge can be obtained via simple matrix multiplication.

[7] Note that $\mathbf{1}_{1\times d}$ denotes the row vector with dimension d, where all its coordinates are one.

Considering that the irregularity is located at $\phi(\overline{\mathbf{X}}) = 0$ (see (2.1))[8] then to determine the discontinuity location y_1^*, we must find, for fixed $\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}$, the roots of $P(y_1^*)$:

$$\phi(\overline{\mathbf{X}}) = \phi\left(X_0^{(1)} \prod_{n=0}^{N-1} F_n^{(1)}(y_1^*, \mathbf{y}_{-1}), \ldots, X_0^{(d)} \prod_{n=0}^{N-1} F_n^{(d)}(y_1^*, \mathbf{y}_{-1})\right) := P(y_1^*) = 0. \tag{2.14}$$

We use the Newton iteration method to determine the approximated discontinuity location $\overline{y}_1^*$.

Remark 2.2. Since the coarsest factors are often the most important ones, we use the Brownian bridge construction. Furthermore, the selection of $\mathcal{A}$ creates a new hierarchy in terms of smoothness.

2.2.2 Step 2 of numerical smoothing: Numerical preintegration

In this stage, we perform the numerical preintegrating step w.r.t. the direction considered for finding the root to determine y_1^*. In fact, using the Fubini's theorem and from (2.11), we obtain

$$\begin{aligned} \mathrm{E}\left[g(\mathbf{X}(T))\right] \approx \mathrm{E}\left[g\left(\overline{X}_T^{(1)}, \ldots, \overline{X}_T^{(d)}\right)\right] &:= \mathrm{E}\left[I\left(\mathbf{Y}_{-1}, \mathbf{Z}_{-1}^{(1)}, \ldots, \mathbf{Z}_{-1}^{(d)}\right)\right] \\ &\approx \mathrm{E}\left[\overline{I}\left(\mathbf{Y}_{-1}, \mathbf{Z}_{-1}^{(1)}, \ldots, \mathbf{Z}_{-1}^{(d)}\right)\right], \end{aligned} \tag{2.15}$$

where

$$\begin{aligned} I\left(\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) &= \int_{\mathbb{R}} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 \\ &= \int_{-\infty}^{y_1^*} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 + \int_{y_1^*}^{+\infty} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1, \end{aligned} \tag{2.16}$$

and $\overline{I}$ is the approximation of I obtained using the Newton iteration and a two-sided Laguerre quadrature rule. It is expressed as

$$\overline{I}(\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}) \coloneqq \sum_{k=0}^{M_{\mathsf{Lag}}} \eta_k \, G\left(\zeta_k\left(\overline{y}_1^*\right), \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right), \tag{2.17}$$

where $\overline{y}_1^*$ denotes the approximated discontinuity location and M_{Lag} represents the number of Laguerre quadrature points $\zeta_k \in \mathbb{R}$ with $\zeta_0 = \overline{y}_1^*$ and corresponding weights η_k [9].

The numerical smoothing treatment enables us to obtain a highly smooth integrand $\overline{I}$ (refer to Section 3.1 for the smoothness analysis).

Remark 2.3 (Extending the numerical smoothing idea to other payoffs and dynamics)**.** Although we consider the case of the multivariate GBM model to illustrate our numerical smoothing approach, we believe that this concept is generic and can be extended in a straightforward manner to several types of payoffs functions and dynamics because of our formulation in Sections 2.2.1 and 2.2.2 (we refer to Section 4 for the different tested examples).

Remark 2.4 (Extending the numerical smoothing approach to the case of multiple roots)**.** The aforementioned preintegration step can be generalized to the case in which there are finetely many discontinuities without accumulation, which occurs either because of the payoff structure or because of the use of the Richardson extrapolation. If we have R different multiple roots, e.g., $\{y_i^*\}_{i=1}^R$ with the following order $y_1^* < y_2^* < \cdots < y_R^*$,

[8] The locations may differ depending on the considered payoff function, for instance, many payoffs in quantitative finance have kinks located at the strike price.

[9] Of course, the points ζ_k must be selected in a systematic manner depending on $\overline{y}_1^*$.

the smoothed integrand in (2.15) is expressed as

$$
\begin{aligned}
I\left(\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) &= \int_{\mathbb{R}} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 \\
&= \int_{-\infty}^{y_1^*} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 + \sum_{i=1}^{R-1} \int_{y_i^*}^{y_{i+1}^*} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 \\
&\quad + \int_{y_R^*}^{+\infty} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1, \qquad (2.18)
\end{aligned}
$$

and its approximation $\overline{I}$ is given by

$$
\begin{aligned}
\overline{I}(\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}) \coloneqq & \sum_{k=0}^{M_{\mathsf{Lag},1}} \eta_k^{\mathsf{Lag}} \, G\left(\zeta_{k,1}^{\mathsf{Lag}}\left(\overline{y}_1^*\right), \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \\
& + \sum_{i=1}^{R-1} \left(\sum_{k=0}^{M_{\mathsf{Leg},i}} \eta_k^{\mathsf{Leg}} \, G\left(\zeta_{k,i}^{\mathsf{Leg}}\left(\overline{y}_i^*, \overline{y}_{i+1}^*\right), \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \right) \\
& + \sum_{k=0}^{M_{\mathsf{Lag},R}} \eta_k^{\mathsf{Lag}} \, G\left(\zeta_{k,R}^{\mathsf{Lag}}\left(\overline{y}_R^*\right), \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right),
\end{aligned}
$$

where $\{\overline{y}_i^*\}_{i=1}^R$ are the approximated discontinuities locations, $M_{\mathsf{Lag},1}$ and $M_{\mathsf{Lag},R}$ are the number of Laguerre quadrature points $\zeta_{\cdot,\cdot}^{\mathsf{Lag}} \in \mathbb{R}$ with corresponding weights $\eta_{\cdot}^{\mathsf{Lag}}$, and $\{M_{\mathsf{Leg},i}\}_{i=1}^{R-1}$ are the number of Legendre quadrature points $\zeta_{\cdot,\cdot}^{\mathsf{Leg}}$ with corresponding weights $\eta_{\cdot}^{\mathsf{Leg}}$[10]. We note that $\overline{I}$ can be further approximated by $\overline{\overline{I}}$ depending on the decay of $G \times \rho_1$ in the semi-infinite domains in (2.18) and depending on how close the roots are to each other. This enables to deal with a countable number of discontinuities by keeping them towards infinity and then truncating the domain.

2.3 Hierarchical quadrature methods combined with numerical smoothing

After performing the numerical smoothing step, we end up with an integration problem (2.15) of a highly regular integrand $\overline{I}$ in a $(dN-1)$-dimensional space. The dimension may become very large because of either the use of (i) numerous time steps N in the discretization scheme or (ii) a large number of assets d. The second stage of our approach involves approximating (2.15) efficiently. To this end, we use two methods, namely, the ASGQ method, using the same construction as in [4, 17, 6], and the randomized QMC method based on lattice rules, as described in [24, 23, 6]. For a clear description of the ASGQ and QMC methods that we employ, we refer to Section 4 in [6].

To overcome the high-dimensionality issue, we use a similar idea to that introduced in [6] and combine the ASGQ and QMC methods with two hierarchical transformations. Thus, we first employ a hierarchical path generation method based on the Brownian bridge construction to reduce the effective dimension and then use the Richardson extrapolation to reduce the bias. Thus, we considerably reduce the dimension of the integration problem. More details on the application of these two hierarchical representations are available in [6]

3 Smoothness Analysis and Error Discussion

3.1 Smoothness analysis

To achieve the optimal performance of the ASGQ and QMC methods, the integrand should be highly smooth. Here, we perform a smoothness analysis of the integrand of interest after employing our numerical smoothing approach. We first introduce some notations and then state our smoothness theorem.

[10] Of course, the points $\zeta_{\cdot,\cdot}^{\mathsf{Lag}}$ and $\zeta_{\cdot,\cdot}^{\mathsf{Leg}}$ must be selected in a systematic manner depending on $\{\overline{y}_i^*\}_{i=1}^R$.

For simplicity, we assume that we work on a fixed time interval $[0,T]$, with $T = 1$. Using the Haar mother wavelet

$$\psi(t) \coloneqq \begin{cases} 1, & 0 \le t < \frac{1}{2}, \\ -1, & \frac{1}{2} \le t < 1, \\ 0, & \text{else}, \end{cases}$$

we construct the Haar basis functions of $L^2([0,1])$ by setting

$$\psi_{-1}(t) \coloneqq \mathbf{1}_{[0,1]}(t); \quad \psi_{n,k}(t) \coloneqq 2^{n/2}\psi\left(2^n t - k\right), \quad n \in \mathbb{N}_0,\ k = 0, \dots, 2^n - 1.$$

Note that the support of $\psi_{n,k}$ is $[2^{-n}k, 2^{-n}(k+1)]$. Moreover, we define a grid $\mathcal{D}^n \coloneqq \left\{ t_\ell^n \mid \ell = 0, \dots, 2^{n+1} \right\}$ by $t_\ell^n \coloneqq \frac{\ell}{2^{n+1}}T$. The Haar basis functions up to level n are piece-wise constants with points of discontinuity given by $\mathcal{D}^n$. Next, we define the antiderivatives of the Haar basis functions:

$$\Psi_{-1}(t) \coloneqq \int_0^t \psi_{-1}(s)ds; \quad \Psi_{n,k}(t) \coloneqq \int_0^t \psi_{n,k}(s)ds.$$

For an i.i.d. set of standard normal rdvs (*coefficients*) Z_{-1}, $Z_{n,k}$, $n \in \mathbb{N}_0$, $k = 0, \dots, 2^n - 1$, we can define the standard Brownian motion

$$W_t \coloneqq Z_{-1}\Psi_{-1}(t) + \sum_{n=0}^{\infty}\sum_{k=0}^{2^n-1} Z_{n,k}\Psi_{n,k}(t),$$

and the truncated version

$$W_t^N \coloneqq Z_{-1}\Psi_{-1}(t) + \sum_{n=0}^{N}\sum_{k=0}^{2^n-1} Z_{n,k}\Psi_{n,k}(t).$$

Note that W^N already coincides with W along the grid $\mathcal{D}^N$. We define the corresponding increments for any function or process F as follows:

$$\Delta_\ell^N F \coloneqq F(t_{\ell+1}^N) - F(t_\ell^N).$$

For simplicity, we consider a one-dimensional SDE for the process X as follows:

$$dX_t = b(X_t)dW_t, \quad X_0 = x \in \mathbb{R}. \tag{3.1}$$

We assume that b and its derivatives of all orders are bounded. Recall that we want to compute, for $g : \mathbb{R} \to \mathbb{R}$ which is not necessarily smooth, $\mathrm{E}\left[g\left(X_T\right)\right]$. Furthermore, we define the solution of the Euler scheme along $\mathcal{D}^N$ by $X_0^N \coloneqq X_0 = x$; for convenience, we also define $X_T^N \coloneqq X_{2^N}^N$.

$$X_{\ell+1}^N \coloneqq X_\ell^N + b\left(X_\ell^N\right)\Delta_\ell^N W, \quad \ell = 0, \dots, 2^N - 1. \tag{3.2}$$

Clearly, the rdv X_ℓ^N is a deterministic function of the rdvs Z_{-1} and $\mathbf{Z}^N \coloneqq \left(Z_{n,k}\right)_{n=0,\dots,N,\ k=0,\dots 2^n-1}$. Using this notation, we write

$$X_\ell^N = X_\ell^N\left(Z_{-1}, \mathbf{Z}^N\right), \tag{3.3}$$

for the appropriate (now deterministic) map $X_\ell^N : \mathbb{R} \times \mathbb{R}^{2^{N+1}-1} \to \mathbb{R}$. We shall write $y \coloneqq z_{-1}$ and $\mathbf{z}^N$ for the (deterministic) arguments of the function X_ℓ^N.[11]

We define the deterministic function $H^N : \mathbb{R}^{2^{N+1}-1} \to \mathbb{R}$, expressed as

$$H^N(\mathbf{z}^N) \coloneqq E\left[g\left(X_T^N\left(Z_{-1}, \mathbf{z}^N\right)\right)\right]. \tag{3.4}$$

Then, H^N satisfies Theorem 3.1. We refer to Appendix A for the proof of Theorem 3.1.

[11] We offer a note of caution regarding convergence as $N \to \infty$: while the sequence of random processes X^N converges to the solution of (3.1) (under the usual assumptions on b), this is not true in any sense for the deterministic functions.

Theorem 3.1. *Assume that X_T^N, defined by* (3.2) *and* (3.3)*, satisfies Assumptions A.1 and A.3. Then, for any $p \in \mathbb{N}$ and indices $n_1, \dots, n_p$ and $k_1, \dots, k_p$ (satisfying $0 \leq k_j < 2^{n_j}$), the function H^N defined in* (3.4) *satisfies the following (with constants independent of n_j, k_j)*

$$\frac{\partial^p H^N}{\partial z_{n_1,k_1} \cdots \partial z_{n_p,k_p}}(\boldsymbol{z}^N) = \mathcal{O}\left(2^{-\sum_{j=1}^p n_j/2}\right).$$

In particular, H^N is of class C^∞.

Remark 3.2 (About the analyticity of H^N)**.** We expect that H^N is analytic; however, a formal proof is subtle. In particular, note that our proof in Appendix A relies on successively applying the technique of dividing by $\frac{\partial X_T^N}{\partial y}$ and then integrating by parts. This means that the constant in $\mathcal{O}\left(2^{-\sum_{j=1}^p n_j/2}\right)$ term will depend on p and increase in p. In other words, Theorem 3.1 should be interpreted as an assertion of the anisotropy in the variables $z_{n,k}$ rather than a statement on the behavior of higher derivatives of H^N. In fact, our proof shows that the number of summands increases as $p!$. Therefore, the statement of the theorem does not already imply analyticity. Note that this problem is an artifact of our construction, and there is no reason to assume such a behavior in general. Finally, we expect the analyticity region to shrink as $N \to \infty$, which motivates the use of the Richardson extrapolation to keep N as small as possible while achieving the desired accuracy.

Remark 3.3. The analysis of the smoothness direction and sufficient conditions for Theorem 3.1 to be valid at high dimensions is an open problem and is beyond the scope of this study.

3.2 Error and work discussion for ASGQ combined with numerical smoothing

In this section, we discuss and analyze the different errors in our approach when using the ASGQ method combined with numerical smoothing. The error analysis of the QMC method combined with numerical smoothing is almost similar, as explained in Remark 3.4.

Let us denote by Q^{ASGQ} the ASGQ estimator[12] used to approximate $\mathrm{E}\left[g(X(T))\right]$, then following the notation given in Section 2.2, we obtain the following error decomposition

$$\begin{aligned}\mathrm{E}\left[g(X(T)\right] - Q^{\text{ASGQ}} = &\underbrace{\mathrm{E}\left[g(X(T))\right] - \mathrm{E}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right]}_{\text{Error I: bias or weak error}} \\ &+ \underbrace{\mathrm{E}\left[I\left(\mathbf{Y}_{-1}, \mathbf{Z}_{-1}^{(1)}, \dots, \mathbf{Z}_{-1}^{(d)}\right)\right] - \mathrm{E}\left[\overline{I}\left(\mathbf{Y}_{-1}, \mathbf{Z}_{-1}^{(1)}, \dots, \mathbf{Z}_{-1}^{(d)}\right)\right]}_{\text{Error II: numerical smoothing error}} \\ &+ \underbrace{\mathrm{E}\left[\overline{I}\left(\mathbf{Y}_{-1}, \mathbf{Z}_{-1}^{(1)}, \dots, \mathbf{Z}_{-1}^{(d)}\right)\right] - Q^{\text{ASGQ}}}_{\text{Error III: ASGQ error}},\end{aligned} \tag{3.5}$$

Because we use schemes based on the forward Euler method to simulate asset dynamics, we achieve

$$\text{Error I} = \mathcal{O}\left(\Delta t\right). \tag{3.6}$$

Let us denote by $\text{TOL}_{\text{Newton}}$ the tolerance of the Newton method used to approximate the discontinuity location by finding the roots of $P(y_1^*)$ defined in (2.14). Thus $|P(\overline{y}_1^*)| \leq \text{TOL}_{\text{Newton}}$, and using the Taylor expansion, we

[12] We refer to Section 4 in [6] for a clear description of the ASGQ estimator used in this work.

obtain that $(y_1^* - \overline{y}_1^*) = \mathcal{O}\left(\mathsf{TOL}_{\mathsf{Newton}}\right)$. Therefore, Error II in (3.5) is expressed as

$$
\begin{aligned}
\text{Error II} &:= \mathrm{E}\left[I\left(\mathbf{Y}_{-1}, \mathbf{Z}_{-1}^{(1)}, \ldots, \mathbf{Z}_{-1}^{(d)}\right)\right] - \mathrm{E}\left[\overline{I}\left(\mathbf{Y}_{-1}, \mathbf{Z}_{-1}^{(1)}, \ldots, \mathbf{Z}_{-1}^{(d)}\right)\right] \\
&\leq \sup_{\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}} \left| I\left(\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) - \overline{I}\left(\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \right| \\
&= \sup_{\mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}} \left| \int_{-\infty}^{y_1^*} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 + \int_{y_1^*}^{+\infty} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 \right. \\
&\qquad - \left(\int_{-\infty}^{\overline{y}_1^*} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 + \int_{\overline{y}_1^*}^{+\infty} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 \right) \\
&\qquad + \left(\int_{-\infty}^{\overline{y}_1^*} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 + \int_{\overline{y}_1^*}^{+\infty} G\left(y_1, \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \rho_1(y_1) dy_1 \right) \\
&\qquad \left. - \sum_{k=0}^{M_{\mathsf{Lag}}} \eta_k \, G\left(\zeta_k\left(\overline{y}_1^*\right), \mathbf{y}_{-1}, \mathbf{z}_{-1}^{(1)}, \ldots, \mathbf{z}_{-1}^{(d)}\right) \right| \\
&= \mathcal{O}\left(M_{\mathsf{Lag}}^{-s}\right) + \mathcal{O}\left(\left|y_1^* - \overline{y}_1^*\right|^{\eta+1}\right) = \mathcal{O}\left(M_{\mathsf{Lag}}^{-s}\right) + \mathcal{O}\left(\mathsf{TOL}_{\mathsf{Newton}}^{\eta+1}\right),
\end{aligned}
\tag{3.7}
$$

where $\eta \geq 0$[13] and $s > 0$ is related to the degree of regularity of the integrand, G, w.r.t. y_1.[14]

The first contribution to the error in (3.7) originates from the one-dimensional preintegration step using the Laguerre quadrature, as explained in Section 2.2.2. Considering that G is a smooth function in parts of the integration domain separated by the discontinuity location, we achieve a spectral convergence of the quadrature [25, 26, 12], which justifies the term M_{Lag}^{-s}. The second contribution to error in (3.7) originates from the gap created by integrating G over domains separated by the approximated discontinuity location $\overline{y}_1^*$ instead of y_1^*, which is the exact location.

Finally, considering M_{ASGQ} quadrature points used in the ASGQ method, we achieve

$$
\text{Error III} = \mathcal{O}\left(M_{\mathsf{ASGQ}}^{-p}\right). \tag{3.8}
$$

Note that $p := p\left(N, d\right) > 0$ is related to the degree of regularity of $\overline{I}$, as defined in (2.15) and (2.16), in the $(dN-1)$-dimensional space[15]. Remember that when using sparse grids (not adaptive), then error III will be $\mathcal{O}\left(M_{\mathsf{SG}}^{-p}\left(\log\left(M_{\mathsf{SG}}\right)\right)^{(d-1)(p-1)}\right)$ [25, 26, 12, 5] (where d is the dimension of the integration domain, and for functions with bounded mixed derivatives up to order p). Moreover it was observed in [13] that the convergence is even spectral for analytic functions. In our case, our smoothness analysis (Section 3.1) implies that $p \gg 1$, which justifies (3.8), under the assumption that $\overline{I}$ converges to I for sufficiently large M_{Lag} and small $\mathsf{TOL}_{\mathsf{Newton}}$. On the other hand, the optimal performance for ASGQ can be deteriorated i) if p and s are not large enough, or ii) due to the adverse effect of the high dimension that may affect the rates badly. Finally, although we work in the preasymptotic regime (small number of time steps, N), we should emphasize that the regularity parameter p may be deteriorated when increasing the dimension of the integration problem by increasing N, which justifies the use of Richardson extrapolation.

Considering (3.5), (3.6), (3.7) and (3.8), the total error estimate of our approach is

$$
\mathcal{E}_{\text{total, ASGQ}} := \mathrm{E}\left[g(X(T)) - Q^{\mathsf{ASGQ}}\right] = \mathcal{O}\left(\Delta t\right) + \mathcal{O}\left(M_{\mathsf{ASGQ}}^{-p}\right) + \mathcal{O}\left(M_{\mathsf{Lag}}^{-s}\right) + \mathcal{O}\left(\mathsf{TOL}_{\mathsf{Newton}}^{\eta+1}\right). \tag{3.9}
$$

[13] The value of η depends on the payoff: for instance, $\eta = 0$ for a digital option and $\eta = 1$ for call/put payoffs. In general, the contribution of the root finding to the numerical smoothing error may have different estimates but for cases where G is a polynomial of degree η then it holds.

[14] In this case, the derivatives of G w.r.t. y_1 are bounded up to order s.

[15] In this case, the weighted mixed derivatives of I are bounded up to order p.

To achieve an optimal performance, we need to optimize the parameters in (3.9) to satisfy a certain error tolerance, TOL, with the least amount of work. This can be achieved by solving (3.10)

$$\begin{cases} \min\limits_{(M_{\text{ASGQ}}, M_{\text{Lag}}, \text{TOL}_{\text{Newton}})} \text{Work}_{\text{ASGQ}} \propto M_{\text{ASGQ}} \times M_{\text{Lag}} \times \Delta t^{-1} \\ s.t.\ \mathcal{E}_{\text{total,ASGQ}} = \text{TOL}. \end{cases} \tag{3.10}$$

Although we do not solve (3.10) in our experiments in Section 4 (we select the parameters heuristically to achieve a suboptimal performance), we show in Appendix B that, for a given error tolerance TOL, and under certain conditions of the regularity parameters s and p ($p, s \gg 1$), a lower bound on the computational work of the ASGQ method is of order $\text{Work}_{\text{ASGQ}} = \mathcal{O}\left(\text{TOL}^{-1}\right)$. This is significantly better than the computational work of order $\mathcal{O}\left(\text{TOL}^{-3}\right)$ achieved by the MC method.

Remark 3.4 (About the error of the QMC method combined with numerical smoothing)**.** Let Q^{rQMC} denotes the randomized QMC (rQMC) estimator used to approximate $\mathrm{E}\left[g(X(T))\right]$ with M_{rQMC} samples. Then, we achieve an error decomposition similar to that shown in (3.5), with Error III being the rQMC statistical error in this case [22], and expressed as

$$\text{Error III (rQMC error)} = \mathcal{O}\left(M_{\text{rQMC}}^{-\frac{1}{2}-\delta} \left(\log M_{\text{rQMC}}\right)^{d\times N-1}\right),$$

where $0 \leq \delta \leq \frac{1}{2}$ is related to the degree of regularity of $\overline{I}$, defined in (2.15) and (2.17).

Remark 3.5. Although we do not use the Richardson extrapolation in our previous analysis, note that using this hierarchical representation improves the complexity rate of the ASGQ method (as observed in our numerical experiments in Section 4).

4 Numerical Experiments

We conduct experiments using three different examples of payoffs: a single digital option, a single call option, and a four-dimensional arithmetic basket call option[16]. These examples are tested under two dynamics for the asset price: the discretized GBM model (a didactic example) and the Heston model, which is a relevant application of our approach (discretization is required). Table 4.1 lists the specifications of each example. Further details of the models and discretization schemes are described in Section 4.1. In Sections 4.2 and 4.3, we illustrate the advantage of combining numerical smoothing with the ASGQ and rQMC methods over the ASGQ and rQMC without smoothing. In Section 4.4, we study the effect of the numerical smoothing parameters on the numerical smoothing error and consequently on the quadrature error of the ASGQ method. Finally, Section 4.5 presents a comparison of the MC and ASGQ methods in terms of errors and computational times. Our ASGQ implementation was based on `https://sites.google.com/view/sparse-grids-kit`.

4.1 Experiments setting

Regarding the numerical experiments under the GBM model, the assets dynamics follow (2.10), and are simulated using the forward Euler scheme. Moreover, we test options under the Heston model [18, 9, 20, 2], dynamics of which is given as follows

$$\begin{aligned} dS_t &= \mu S_t dt + \sqrt{v_t} S_t dW_t^S = \mu S_t dt + \rho\sqrt{v_t} S_t dW_t^v + \sqrt{1-\rho^2}\sqrt{v_t} S_t dW_t \\ dv_t &= \kappa(\theta - v_t)dt + \xi\sqrt{v_t} dW_t^v, \end{aligned} \tag{4.1}$$

where S_t denotes the asset price; v_t represents the instantaneous variance; $\left(W_t^S, W_t^v\right)$ are the correlated Wiener processes with correlation ρ; μ represents the asset's rate of return; θ is the mean variance; κ is the rate at which v_t reverts to θ; and ξ denotes the volatility of the volatility.

[16]The payoff g is expressed by $g(\mathbf{x}) = \max\left(\sum_{j=1}^{d} c_j x^{(j)} - K, 0\right)$, where $\{c_j\}_{j=1}^{d}$ denote the weights of the basket.

Example	Parameters	Reference solution
Single digital option under GBM	$\sigma = 0.4, r = 0, T = 1, S_0 = K = 100$	0.42074
Single digital option under Heston	$v_0 = 0.04, \mu = 0, \rho = -0.9, \kappa = 1, \xi = 0.1,$ $\theta = 0.0025, S_0 = K = 100$	0.5146 $(2.0e{-}05)$
Single call option under GBM	$\sigma = 0.4, r = 0, T = 1, S_0 = K = 100$	15.8519
Single call option under Heston	$v_0 = 0.04, \mu = 0, \rho = -0.9, \kappa = 1, \xi = 0.1,$ $\theta = 0.0025, S_0 = K = 100$	6.33254
4-dimensional basket call option under GBM	$\sigma_{1,2,3,4} = 0.4, \rho = 0.3, r = 0, T = 1,$ $S_0^{1,2,3,4} = K = 100, c_{1,2,3,4} = 1/4$	11.04 $(1.0e{-}03)$

Table 4.1: Models and options parameters of the tested examples along with their reference solution. The reference solution for the call option under the Heston model is computed using Premia software with the method in [18]. The numbers between parentheses correspond to the statistical error estimates when the reference solution is estimated using the MC estimator.

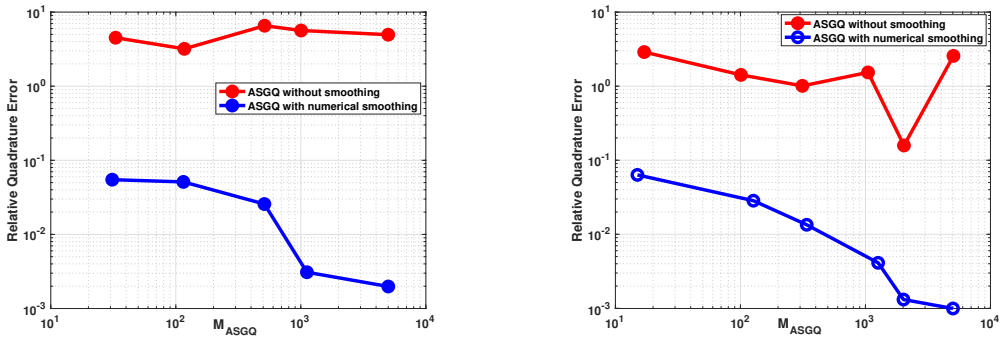
Many simulation schemes of (4.1) have been proposed in the literature. In Appendix C, we provide an overview of the most popular methods in this context. These methods mainly differ in the way in which they simulate the volatility process to ensure its positivity.

We recall that the ASGQ and rQMC methods are extremely sensitive to the smoothness of the integrand. We numerically found (Appendix C.4) that the use of a nonsmooth transformation to ensure the positivity of the volatility process deteriorates the performance of the ASGQ method. To overcome this undesirable feature, we propose the use of an alternative scheme, namely, the Heston OU-based scheme, in which the volatility is simulated as the sum of the Ornstein-Uhlenbeck (OU) or Bessel processes (Appendix C.3). In fact, in the literature [2, 21, 1], the focus was on designing schemes that ensure the positivity of the volatility process and show a good weak error behavior. In our setting, an optimal scheme is determined based on two criteria: (i) the behavior of the rates of mixed differences, which is an important feature for ensuring the optimal performance of the ASGQ method (see Appendix C.4.2 for more details), and (ii) the weak error behavior to apply the Richardson extrapolation when necessary. A comparison of the different schemes (Appendices C.4.1 and C.4.2) suggests that the Heston OU-based scheme yields the best results based on our criteria. Therefore, in our numerical experiments, we use this scheme with the ASGQ and rQMC methods. For the MC method, we use the full truncation scheme (explained in Appendix C.1).

4.2 Comparison of the ASGQ method with and without numerical smoothing

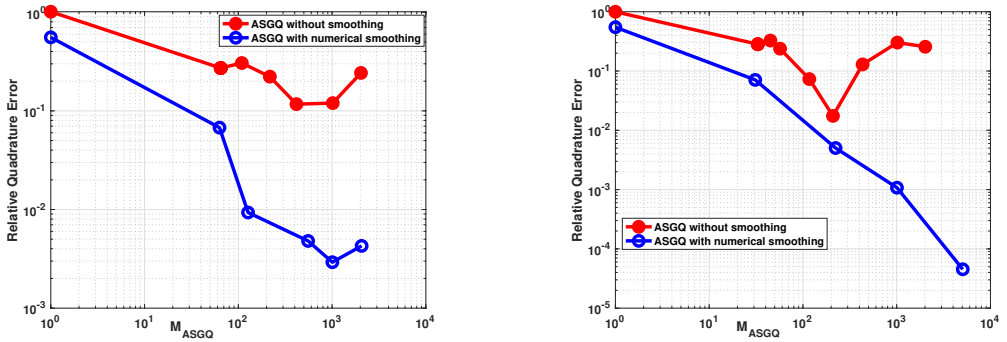
In this section, we illustrate the advantage of combining numerical smoothing with the ASGQ method over the ASGQ method without numerical smoothing. Figures 4.1 and 4.2 show comparisons of the relative quadrature error convergence for the different examples under the Heston model listed in Table 4.1, for cases with and without the Richardson Extrapolation, and for some number of time steps N[17]. We can clearly see that the numerical smoothing significantly improves the quadrature error convergence for all cases; this result is in agreement with Theorem 3.1. For instance, for the call option under the Heston model (left plot in Figure 4.2), the ASGQ method without smoothing cannot achieve a relative quadrature error below 10%, even in the case of more than 10^3 quadrature points. Alternatively, for the same number of quadrature points, the ASGQ method with numerical smoothing achieves a relative quadrature error below 1%. The gains are more evident for the digital option case (Figure 4.1). Further, using the Richardson extrapolation, the ASGQ method with numerical smoothing yields a smaller quadrature error.

[17] Note that the dimension of the integration problem is N for the GBM examples and $2N$ for the Heston examples.



(a) (b)

Figure 4.1: Digital option under the Heston model: Comparison of the relative quadrature error convergence for the ASGQ method with and without numerical smoothing. (a) Without Richardson extrapolation ($N = 8$), (b) with Richardson extrapolation ($N_{\text{fine level}} = 8$).



(a) (b)

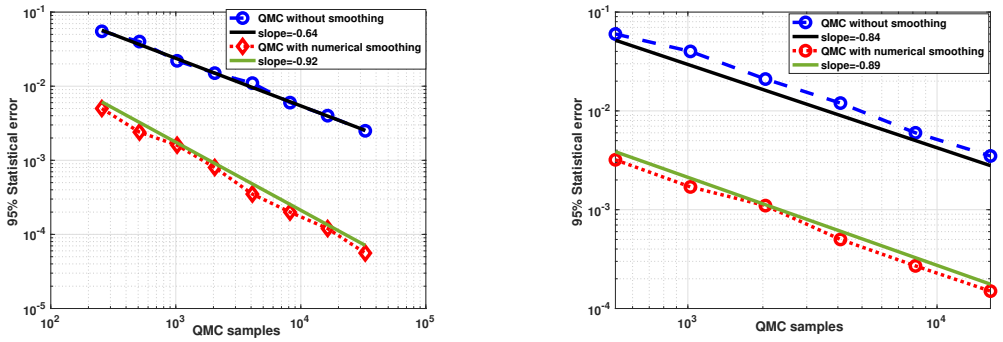
Figure 4.2: Call option under the Heston model: Comparison of the relative quadrature error convergence for the ASGQ method with and without numerical smoothing. (a) Without Richardson extrapolation ($N = 16$), (b) with Richardson extrapolation ($N_{\text{fine level}} = 8$).

4.3 Comparison of the RQMC method with and without numerical smoothing

In this section, we show the advantage of combining numerical smoothing with the rQMC method over the rQMC method without smoothing. Figures 4.3 and 4.4 show comparisons of the statistical error convergence for the examples listed in Table 4.1, and for some number of time steps N. Because regularity was regained using numerical smoothing, we observe an improvement in the statistical error convergence of the rQMC method, most noticeably in the case of the digital option based on the GBM and Heston models (left plots in Figures 4.3 and 4.4).

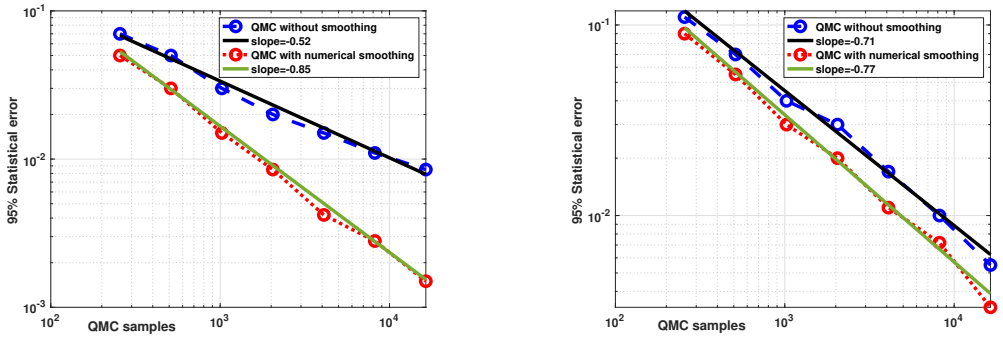
4.4 Study of the numerical smoothing parameters

In this section, we study the effect of the numerical smoothing parameters on the relative numerical smoothing error for sufficiently large ASGQ points $M_{\text{ASGQ}} = 10^3$. These parameters are (i) the number of Laguerre points used in the preintegration step, M_{Lag}, and (ii) the Newton tolerance used in the root-finding step, $\text{TOL}_{\text{Newton}}$. Figures 4.5 and 4.6 show the results for the digital and call options under the GBM model as listed in Table 4.1, for the case without Richardson extrapolation and for $N = 4$. These plots provide a numerical verficiation of our error estimates in (3.7) for the numerical smoothing error, where a faster convergence of the root finding



(a) (b)

Figure 4.3: Comparison of the 95% statistical error convergence for rQMC with and without numerical smoothing with $N = 8$. (a) Digital option under GBM, (b) call option under GBM.



(a) (b)

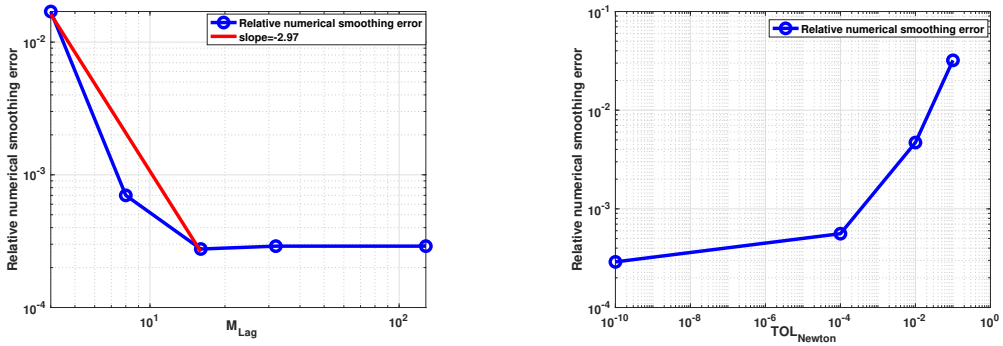
Figure 4.4: Comparison of the 95% statistical error convergence for rQMC with and without numerical smoothing with $N = 4$. (a) Digital option under Heston, (b) call option under Heston.

and quadrature errors can be achieved for the call option compared to the digital option. Moreover, we observe that the numerical smoothing procedure is very cheap since few Laguerre quadrature points and large values of Newton tolerance are required to achieve a certain accuracy. Note that similar observations are obtained for other tested examples.

4.5 ASGQ method with numerical smoothing versus MC method

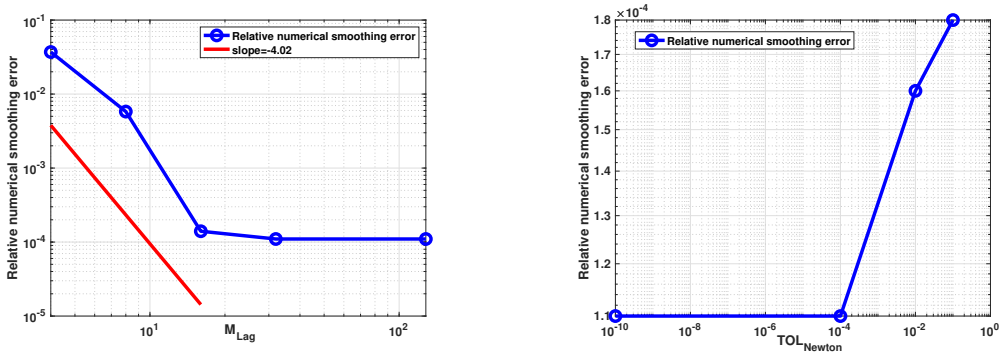
For a sufficiently fixed small error tolerance in the price estimates, we compare the computational time needed for the MC method and the ASGQ method with numerical smoothing to meet the desired error tolerance. The reported errors are relative errors normalized using the reference solutions. Furthermore, we conduct our numerical experiments for two different scenarios: without Richardson extrapolation, and with level-1 Richardson extrapolation. Note that the actual work (runtime) is obtained using a 3,2 GHz 8-Core Intel Xeon W architecture.

The numerical findings are summarized in Table 4.2. The reported results highlight the computational gains achieved using the ASGQ method with numerical smoothing compared to the MC method to meet a relative error below 1%. These results correspond to the best configuration with the Richardson extrapolation for each method. More details for each case are provided in Figures 4.7, 4.8 and 4.9, which show comparisons of the numerical complexity of each method under the two scenarios of Richardson extrapolation. These figures illustrate that to achieve a relative error of less than 1%, the optimal configuration is level-1 Richardson extrapolation for both the MC and ASGQ methods, except for the four-dimensional basket call option under the GBM model.



(a) (b)

Figure 4.5: Digital option under GBM with $N = 4$: The relative numerical smoothing error for a fixed number of ASGQ points $M_{\text{ASGQ}} = 10^3$ plotted against (a) different values of N_{Lag} with a fixed Newton tolerance $\text{TOL}_{\text{Newton}} = 10^{-10}$, (b) different values of $\text{TOL}_{\text{Newton}}$ with a fixed number of Laguerre quadrature points $M_{\text{Lag}} = 128$.



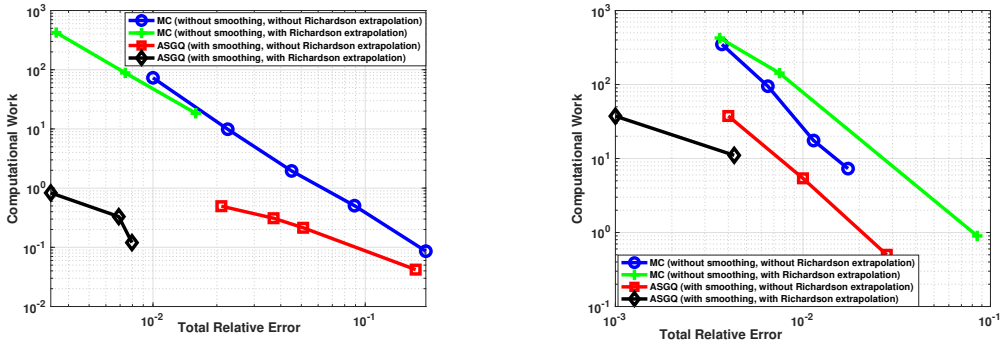
(a) (b)

Figure 4.6: Call option under GBM with $N = 4$: The relative numerical smoothing error for a fixed number of ASGQ points $M_{\text{ASGQ}} = 10^3$ plotted against (a) different values of N_{Lag} with a fixed Newton tolerance $\text{TOL}_{\text{Newton}} = 10^{-10}$, (b) different values of $\text{TOL}_{\text{Newton}}$ with a fixed number of Laguerre quadrature points $M_{\text{Lag}} = 128$.

Example	Total relative error	CPU time (ASGQ/MC) in %
Single digital option (GBM)	0.4%	0.2%
Single call option (GBM)	0.5%	0.3%
Single digital option (Heston)	0.4%	3.2%
Single call option (Heston)	0.5%	0.4%
4-dimensional basket call option (GBM)	0.8%	7.4%

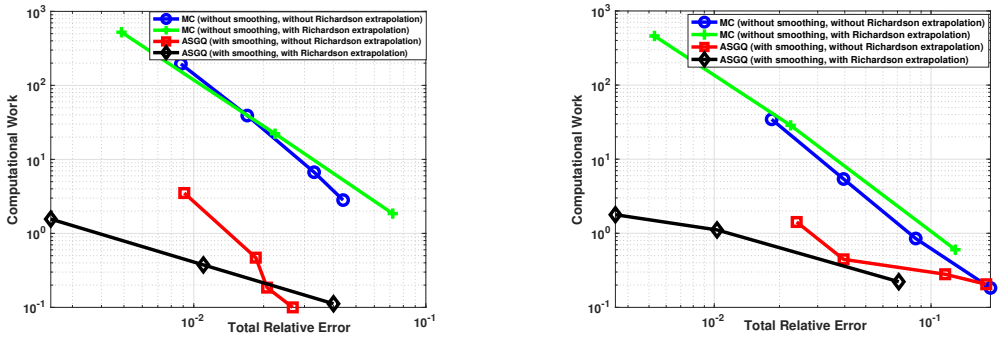
Table 4.2: Summary of the relative errors and computational gains achieved using ASGQ with numerical smoothing compared to the MC method, to realize a certain error tolerance. The CPU time ratios are computed for the best configuration with Richardson extrapolation for each method.

Remark 4.1 (About rQMC with numerical smoothing)**.** We also combine numerical smoothing with the rQMC method, and observe an improvement in the performance compared to the case without smoothing (Section 4.3). Moroever, the rQMC method with numerical smoothing consistently outperforms the MC method to achieve a relative error below 1%. However, we consistently observe that the ASGQ method outperforms the rQMC method in all our numerical examples, when both are combined with numerical smoothing. In particular, as an



(a) (b)

Figure 4.7: Computational work comparison for the different methods with the different configurations in terms of the level of Richardson extrapolation. To achieve a relative error below 1%, ASGQ combined with numerical smoothing and level-1 Richardson extrapolation significantly outperforms the other methods. (a) Digital option under GBM, (b) digital option under Heston.



(a) (b)

Figure 4.8: Computational work comparison for the different methods with the different configurations in terms of the level of Richardson extrapolation. To achieve a relative error below 1%, ASGQ combined with numerical smoothing and level-1 Richardson extrapolation significantly outperforms the other methods. (a) Call option under GBM, (b) call option under Heston.

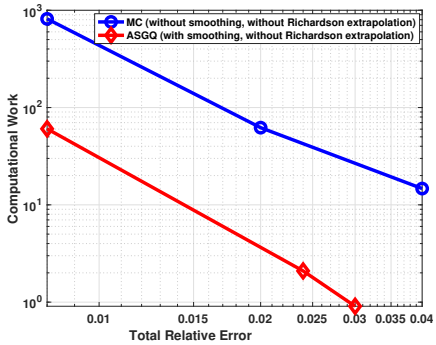


Figure 4.9: Four-dimensional basket call option under GBM: Computational work comparison for the different methods. To achieve a relative error below 1%, ASGQ combined with numerical smoothing significantly outperforms the MC method.

illustration, Figure 4.10 shows the comparison for the example of the digital option under the GBM model.

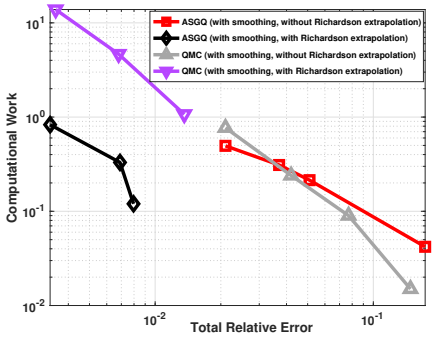


Figure 4.10: Digital option under GBM: Computational work comparison for rQMC and ASGQ, both combined with numerical smoothing, with the different configurations in terms of the level of Richardson extrapolation. To achieve a relative error below 1%, ASGQ combined with numerical smoothing and level-1 Richardson extrapolation significantly outperforms the other methods.

References

[1] Aurélien Alfonsi. High order discretization schemes for the CIR process: application to affine term structure and Heston models. *Mathematics of Computation*, 79(269):209–237, 2010.

[2] Leif Andersen. Efficient simulation of the Heston stochastic volatility model. *Available at SSRN 946405*, 2007.

[3] Leif Andersen and Rupert Brotherton-Ratcliffe. Extended libor market models with stochastic volatility. *Available at SSRN 294853*, 2001.

[4] J. Bäck, F. Nobile, L. Tamellini, and R. Tempone. Stochastic spectral Galerkin and collocation methods for PDEs with random coefficients: a numerical comparison. In J.S. Hesthaven and E.M. Ronquist, editors, *Spectral and High Order Methods for Partial Differential Equations*, volume 76 of *Lecture Notes in Computational Science and Engineering*, pages 43–62. Springer, 2011. Selected papers from the ICOSAHOM '09 conference, June 22-26, Trondheim, Norway.

[5] Volker Barthelmann, Erich Novak, and Klaus Ritter. High dimensional polynomial interpolation on sparse grids. *Advances in Computational Mathematics*, 12(4):273–288, 2000.

[6] Christian Bayer, Chiheb Ben Hammouda, and Raúl Tempone. Hierarchical adaptive sparse grids and quasi-Monte Carlo for option pricing under the rough Bergomi model. *Quantitative Finance*, 20(9):1457–1473, 2020.

[7] Christian Bayer, Markus Siebenmorgen, and Raúl Tempone. Smoothing the payoff for efficient computation of basket option pricing. *Quantitative Finance*, 18(3):491–505, 2018.

[8] Chiheb Ben Hammouda. *Hierarchical Approximation Methods for Option Pricing and Stochastic Reaction Networks*. PhD thesis, 2020.

[9] Mark Broadie and Özgür Kaya. Exact simulation of stochastic volatility and other affine jump diffusion processes. *Operations research*, 54(2):217–231, 2006.

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

[11] Christophe De Luigi, Jérôme Lelong, and Sylvain Maire. Robust adaptive numerical integration of irregular functions with applications to basket and other multi-dimensional exotic options. *Applied Numerical Mathematics*, 100:14–30, 2016.

[12] Thomas Gerstner and Michael Griebel. Numerical integration using sparse grids. *Numerical algorithms*, 18(3):209–232, 1998.

[13] Thomas Gerstner and Michael Griebel. Dimension–adaptive tensor–product quadrature. *Computing*, 71:65–87, 2003.

[14] Michael Griebel, Frances Kuo, and Ian Sloan. The smoothing effect of integration in $\mathbb{R}^d$ and the ANOVA decomposition. *Mathematics of Computation*, 82(281):383–400, 2013.

[15] Michael Griebel, Frances Kuo, and Ian Sloan. Note on "the smoothing effect of integration in $\mathbb{R}^d$ and the ANOVA decomposition". *Mathematics of Computation*, 86(306):1847–1854, 2017.

[16] Andreas Griewank, Frances Y Kuo, Hernan Leövey, and Ian H Sloan. High dimensional integration of kinks and jumps-smoothing by preintegration. *Journal of Computational and Applied Mathematics*, 344:259–274, 2018.

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

[18] Steven L Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The review of financial studies*, 6(2):327–343, 1993.

[19] Monique Jeanblanc, Marc Yor, and Marc Chesney. *Mathematical methods for financial markets*. Springer Science & Business Media, 2009.

[20] Christian Kahl and Peter Jäckel. Fast strong approximation Monte Carlo schemes for stochastic volatility models. *Quantitative Finance*, 6(6):513–536, 2006.

[21] Roger Lord, Remmert Koekkoek, and Dick Van Dijk. A comparison of biased simulation schemes for stochastic volatility models. *Quantitative Finance*, 10(2):177–194, 2010.

[22] Harald Niederreiter. *Random number generation and quasi-Monte Carlo methods*, volume 63. Siam, 1992.

[23] Dirk Nuyens. The construction of good lattice rules and polynomial lattice rules. In *Uniform distribution and quasi-Monte Carlo methods*, pages 223–256. De Gruyter, 2014.

[24] Ian H. Sloan. Lattice methods for multiple integration. *Journal of Computational and Applied Mathematics*, 12-13:131–143, 1985.

[25] Sergei Abramovich Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Doklady Akademii Nauk*, volume 148, pages 1042–1045. Russian Academy of Sciences, 1963.

[26] G.W. Wasilkowski and H. Wozniakowski. Explicit cost bounds of algorithms for multivariate tensor product problems. *Journal of Complexity*, 11(1):1–56, 1995.

[27] Ye Xiao and Xiaoqun Wang. Conditional quasi-Monte Carlo methods and dimension reduction for option pricing and hedging with discontinuous functions. *Journal of Computational and Applied Mathematics*, 343:289–308, 2018.

A Details for the Proof of Theorem 3.1 in Section 3.1

Here, we use same notation as in Section 3.1. Moreover, we introduce the following notation: for sequences of rdvs F_N and G_N we assume that $F_N = \mathcal{O}(G_N)$ if there is a rdv C with finite moments of all orders such that for all N, we have $|F_N| \leq C\,|G_N|$ a.s.

We consider a mollified version g_δ of g and the corresponding function H^N_δ (defined by replacing g with g_δ in (3.4)). Tacitly, assuming that we can interchange the integration and differentiation (refer to Lemma A.4 for justification), we achieve

$$\frac{\partial H^N_\delta(\mathbf{z}^N)}{\partial z_{n,k}} = E\left[g'_\delta\left(X^N_T\left(Z_{-1},\mathbf{z}^N\right)\right)\frac{\partial X^N_T(Z_{-1},\mathbf{z}^N)}{\partial z_{n,k}}\right].$$

Multiplying and dividing by $\frac{\partial X^N_T(Z_{-1},\mathbf{z}^N)}{\partial y}$ and replacing the expectation by an integral w.r.t. the standard normal density, we obtain

$$\frac{\partial H^N_\delta(\mathbf{z}^N)}{\partial z_{n,k}} = \int_{\mathbb{R}} \frac{\partial g_\delta\left(X^N_T(y,\mathbf{z}^N)\right)}{\partial y}\left(\frac{\partial X^N_T}{\partial y}(y,\mathbf{z}^N)\right)^{-1}\frac{\partial X^N_T}{\partial z_{n,k}}(y,\mathbf{z}^N)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}dy. \tag{A.1}$$

If integration by parts is possible, we can discard the mollified version and obtain the smoothness of H^N because

$$\frac{\partial H^N(\mathbf{z}^N)}{\partial z_{n,k}} = -\int_{\mathbb{R}} g\left(X^N_T(y,\mathbf{z}^N)\right)\frac{\partial}{\partial y}\left[\left(\frac{\partial X^N_T}{\partial y}(y,\mathbf{z}^N)\right)^{-1}\frac{\partial X^N_T}{\partial z_{n,k}}(y,\mathbf{z}^N)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\right]dy.$$

However, there are situations in which there may be a potential problem looming in the inverse of the derivative w.r.t. y[18]. This observation motivates the introduction of Assumptions A.1 and A.3.

Assumption A.1. There are positive rdvs C_p with finite moments of all orders such that

$$\forall N\in\mathbb{N},\ \forall \ell_1,\dots,\ell_p\in\{0,\dots,2^N-1\}:\ \left|\frac{\partial^p X^N_T}{\partial X^N_{\ell_1}\cdots\partial X^N_{\ell_p}}\right| \le C_p \text{ a.s.}$$

In terms of the aforementioned notation, this means that $\frac{\partial^p X^N_T}{\partial X^N_{\ell_1}\cdots\partial X^N_{\ell_p}} = \mathcal{O}(1)$.

Remark A.2. It is probably difficult to argue that a deterministic constant C may exist in Assumption A.1 .

Assumption A.1 is natural because it is fulfilled if the diffusion coefficient $b(.)$ is smooth; this situation is valid for many option pricing models. However, now we need to make a much more serious assumption that may be difficult to verify in practice for some models. In Appendix A.1, we motivate cases with sufficient conditions where this assumption is valid and discuss its limitations.

Assumption A.3. For any $p\in\mathbb{N}$ we obtain

$$\left(\frac{\partial X^N_T}{\partial y}\left(Z_{-1},\mathbf{Z}^N\right)\right)^{-p} = \mathcal{O}(1).$$

Lemma A.4. *We have*

$$\frac{\partial X^N_T}{\partial z_{n,k}}(Z_{-1},\boldsymbol{Z}^N) = 2^{-n/2+1}\mathcal{O}(1)$$

in the sense that the $\mathcal{O}(1)$ term does not depend on n or k.

Proof. First, note that Assumption A.1 implies that $\frac{\partial X^N_T}{\partial \Delta^N_\ell W} = \mathcal{O}(1)$. Indeed, we have

$$\frac{\partial X^N_T}{\partial \Delta^N_\ell W} = \frac{\partial X^N_T}{\partial X^N_{\ell+1}}\frac{\partial X^N_{\ell+1}}{\partial \Delta^N_\ell W} = \mathcal{O}(1)b(X^N_\ell) = \mathcal{O}(1).$$

Next we need to identify the increments Δ^N_ℓ that depend on $Z_{n,k}$. This is the case iff the support of $\psi_{n,k}$ has a nonempty intersection with $]t^N_\ell, t^N_{\ell+1}[$. Explicitly, this means that

$$\ell 2^{-(N-n+1)} - 1 < k < (\ell+1)2^{-(N-n+1)}.$$

[18] As an example, let us assume that $X^N_T(y,\mathbf{z}^N) = \cos(y) + z_{n,k}$. Then (A.1) is generally not integrable.

If we fix N, k, and n, this means that the derivative of $\Delta_\ell^N W$ w.r.t. $Z_{n,k}$ does not vanish iff

$$2^{N-n+1}k \le \ell < 2^{N-n+1}(k+1).$$

Because

$$\left|\frac{\partial \Delta_\ell^N W}{\partial Z_{n,k}}\right| = \left|\Delta_\ell^N \Psi_{n,k}\right| \le 2^{-(N-n/2)}, \tag{A.2}$$

we thus obtain

$$\frac{\partial X_T^N}{\partial z_{n,k}}(Z_{-1}, \mathbf{Z}^N) = \sum_{\ell=2^{N-n+1}k}^{2^{N-n+1}(k+1)-1} \frac{\partial X_T^N}{\partial \Delta_\ell^N W}\frac{\partial \Delta_\ell^N W}{\partial Z_{n,k}} = 2^{N-n+1}2^{-(N-n/2)}\mathcal{O}(1) = 2^{-n/2+1}\mathcal{O}(1). \tag{A.3}$$ □

Lemma A.5. *Similar to Lemma A.4, we have*

$$\frac{\partial^2 X_T^N}{\partial y \partial z_{n,k}}(Z_{-1}, \boldsymbol{Z}^N) = 2^{-n/2+1}\mathcal{O}(1).$$

Proof. $\Delta_\ell^N W$ is a linear function in Z_{-1} and $\mathbf{Z}^N$, implying that all mixed derivatives $\frac{\partial^2 \Delta_\ell^N W}{\partial Z_{n,k}\partial Z_{-1}}$ vanish. From equation (A.3) we hence obtain

$$\frac{\partial^2 X_T^N}{\partial z_{n,k}\partial y}(Z_{-1}, \mathbf{Z}^N) = \sum_{\ell=2^{N-n+1}k}^{2^{N-n+1}(k+1)-1} \frac{\partial^2 X_T^N}{\partial \Delta_\ell^N W \partial Z_{-1}}\frac{\partial \Delta_\ell^N W}{\partial Z_{n,k}}.$$

Further,

$$\frac{\partial^2 X_T^N}{\partial \Delta_\ell^N W \partial Z_{-1}} = \sum_{j=0}^{2^{N+1}-1} \frac{\partial^2 X_T^N}{\partial \Delta_\ell^N W \partial \Delta_j^N W}\frac{\partial \Delta_j^N W}{\partial Z_{-1}}.$$

Note that

$$\frac{\partial^2 X_T^N}{\partial \Delta_\ell^N W \partial \Delta_j^N W} = \frac{\partial^2 X_T^N}{\partial X_{\ell+1}^N \partial X_{j+1}^N} b(X_\ell^N) b(X_j^N) + \mathbb{1}_{j<\ell}\frac{\partial X_T^N}{\partial X_\ell^N} b'(X_\ell^N)\frac{\partial X_\ell^N}{\partial X_{j+1}^N} b(X_j^N) = \mathcal{O}(1) \tag{A.4}$$

using Assumption A.1. We also have $\frac{\partial \Delta_j^N W}{\partial Z_{-1}} = \mathcal{O}(2^{-N})$, implying the statement of the lemma. □

Remark A.6. Lemmas A.4 and A.5 also hold (mutatis mutandis) for $z_{n,k} = y$ (with $n = 0$).

Proposition A.7. *We have* $\frac{\partial H^N(\boldsymbol{z}^N)}{\partial z_{n,k}} = \mathcal{O}(2^{-n/2})$ *such that the constant in front of* $2^{-n/2}$ *does not depend on* n *or* k.

Proof. We have

$$\begin{aligned}
\frac{\partial H^N(\mathbf{z}^N)}{\partial z_{n,k}} &= -\int_{\mathbb{R}} g\left(X_T^N(y,\mathbf{z}^N)\right)\frac{\partial}{\partial y}\left[\left(\frac{\partial X_T^N}{\partial y}(y,\mathbf{z}^N)\right)^{-1}\frac{\partial X_T^N}{\partial z_{n,k}}(y,\mathbf{z}^N)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\right]dy \\
&= -\int_{\mathbb{R}} g\left(X_T^N(y,\mathbf{z}^N)\right)\left[-\left(\frac{\partial X_T^N}{\partial y}(y,\mathbf{z}^N)\right)^{-2}\frac{\partial^2 X_T^N}{\partial y^2}(y,\mathbf{z}^N)\frac{\partial X_T^N}{\partial z_{n,k}}(y,\mathbf{z}^N)+\right. \\
&\quad \left.+\left(\frac{\partial X_T^N}{\partial y}(y,\mathbf{z}^N)\right)^{-1}\frac{\partial^2 X_T^N}{\partial z_{n,k}\partial y}(y,\mathbf{z}^N) - y\left(\frac{\partial X_T^N}{\partial y}(y,\mathbf{z}^N)\right)^{-1}\frac{\partial X_T^N}{\partial z_{n,k}}(y,\mathbf{z}^N)\right]\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}dy.
\end{aligned}$$

Hence, Lemmas A.4 and Lemma A.5 together with Assumption A.3 (for $p = 2$) imply that

$$\frac{\partial H^N(\mathbf{z}^N)}{\partial z_{n,k}} = \mathcal{O}(2^{-n/2}),$$

with constants independent of n and k.[19] □

[19] Notice that when $F^N(Z_{-1}, \mathbf{Z}^N) = \mathcal{O}(c)$ for some deterministic constant c, this property is retained when integrating out one of the rdvs, i.e., we still achieve $\int_{\mathbb{R}} F^N(y, \mathbf{Z}^N)\frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}dy = \mathcal{O}(c)$.

For the general case we need the following Lemma.

Lemma A.8. *For any $p \in \mathbb{N}$ and indices $n_1, \dots, n_p$ and $k_1, \dots, k_p$ (satisfying $0 \leq k_j < 2^{n_j}$) we have (with constants independent of n_j, k_j)*

$$\frac{\partial^p X_T^N}{\partial z_{n_1,k_1} \cdots \partial z_{n_p,k_p}}(Z_{-1}, \boldsymbol{Z}^N) = \mathcal{O}\left(2^{-\sum_{j=1}^p n_j/2}\right).$$

The result also holds (mutatis mutandis) if one or several z_{n_j,k_j} are replaced by $y = z_{-1}$ (with n_j set to 0).

Proof. We start noting that each $\Delta_\ell^N W$ is a linear function of $(Z_{-1}, \mathbf{Z}^N)$ implying that all higher derivatives of $\Delta_\ell^N W$ w.r.t. $(Z_{-1}, \mathbf{Z}^N)$ vanish. Hence,

$$\frac{\partial^p X_T^N}{\partial Z_{n_1,k_1} \cdots \partial Z_{n_p,k_p}} = \sum_{\ell_1=2^{N-n_1+1}k_1}^{2^{N-n_1+1}(k_1+1)-1} \cdots \sum_{\ell_p=2^{N-n_p+1}k_p}^{2^{N-n_p+1}(k_p+1)-1} \frac{\partial^p X_T^N}{\partial \Delta_{\ell_1}^N \cdots \partial \Delta_{\ell_p}^N W} \frac{\partial \Delta_{\ell_1}^N W}{\partial Z_{n_1,k_1}} \cdots \frac{\partial \Delta_{\ell_p}^N W}{\partial Z_{n_p,k_p}}.$$

By an argument similar to that made for Assumption A.4, we obtain

$$\frac{\partial^p X_T^N}{\partial \Delta_{\ell_1}^N \cdots \partial \Delta_{\ell_p}^N W} = \mathcal{O}(1).$$

By (A.2) we see that each summand in the aforementioned sum is of order $\prod_{j=1}^p 2^{-(N-n_j/2)}$. The number of summands in total is $\prod_{j=1}^p 2^{N-n_j+1}$. Therefore, we obtain the desired result. □

Sketch of a proof of Theorem 3.1. We apply integration by parts p times, as performed in the proof of Proposition A.7, which shows that we can again replace the mollified payoff function g_δ by the true, nonsmooth function g. Moreover, using this procedure, we obtain a formula of the form

$$\frac{\partial^p H^N}{\partial z_{n_1,k_1} \cdots \partial z_{n_p,k_p}}(z^N) = \int_{\mathbb{R}} g\left(X_T^N(y, z^N)\right) \blacksquare \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy,$$

where $\blacksquare$ represents a long sum of products of various terms. However, note that when the derivatives w.r.t. y are ignored, each summand contains all derivatives w.r.t. $z_{n_1,k_1}, \dots, z_{n_p,k_p}$ exactly once. (Generally, each summand will be a product of the derivatives of X_T^N w.r.t. some z_{n_j,k_j}s, possibly including other terms such as polynomials in y and derivatives w.r.t. y.) As all other terms are assumed to be of order $\mathcal{O}(1)$ based on Assumptions A.1 and A.3, the result suggested by Lemma A.8 is implied, and ends up the proof of Theorem 3.1. □

A.1 Discussion of Assumption A.3

Here, we motivate sufficient conditions for Assumption A.3 to be valid in the one-dimensional setting. Moreover, we discuss its limitation and some multivariate cases in which this assumption hold.

We want to examine the term given by $\left(\frac{\partial X_T^N}{\partial y}\left(Z_{-1}, \mathbf{Z}^N\right)\right)^{-p}$ for $p \in \mathbb{N}$. For this, we consider the one-dimensional SDE

$$dX_t = a(X_t)dt + b(X_t)dW_t.$$

For ease of presentation, we set the drift term $a(.)$ to zero. Moreover, using the Brownian bridge construction, we achieve

$$dX_t = b(X_t)\left(\frac{y}{\sqrt{T}}dt + dB_t\right), \tag{A.5}$$

where y is a standard Gaussian rdv and B is the Brownian bridge.

The solution of (A.5), at the final time $T > 0$ is

$$X_T = x_0 + \frac{y}{\sqrt{T}}\int_0^T b(X_s)ds + \int_0^T b(X_s)dB_s,$$

and consequently,

$$\frac{\partial X_T}{\partial y} = \frac{y}{\sqrt{T}} \int_0^T b'(X_s) \frac{\partial X_s}{\partial y} ds + \frac{1}{\sqrt{T}} \int_0^T b(X_s) ds + \int_0^T b'(X_s) \frac{\partial X_s}{\partial y} dB_s.$$

This implies that $\frac{\partial X_T}{\partial y}$ solves

$$\begin{cases} d\left(\frac{\partial X_T}{\partial y}\right) & = \frac{b(X_t)}{\sqrt{T}} dt + b'(X_t) \frac{\partial X_t}{\partial y} dW_t, \\ \frac{\partial X_T}{\partial y} \mid_{t=0} & = 0. \end{cases}$$

Using Duhamel's principle, we obtain

$$\frac{\partial X_T}{\partial y} = \int_0^T \frac{b(X_s)}{\sqrt{T}} \exp\left(\left(\int_s^T b'(X_u) dW_u\right) - \frac{1}{2} \int_s^T (b')^2(X_u) du\right) ds.$$

If there exists $b_0 \in \mathbb{R}$ such that

$$b^2(x) \geq b_0^2, \quad \forall x \in \mathbb{R}, \tag{A.6}$$

then

$$\begin{aligned} \left|\frac{\partial X_T}{\partial y}\right| &\geq \frac{|b_0|}{\sqrt{T}} \int_0^T \exp\left(\left(\int_s^T b'(X_u) dW_u\right) - \frac{1}{2} \int_s^T (b')^2(X_u) du\right) ds, \\ &\geq \frac{|b_0|}{\sqrt{T}} \exp\left(\left(\int_0^T b'(X_u) dW_u\right) - \frac{1}{2} \int_0^T (b')^2(X_u) du\right) ds, \end{aligned}$$

and consequently, for any $p \in \mathbb{N}$, we obtain

$$\left(\left|\frac{\partial X_T}{\partial y}\right|\right)^{-p} \leq \left(\frac{|b_0|}{\sqrt{T}}\right)^{-p} \exp\left(-p\left(\left(\int_0^T b'(X_u) dW_u\right) - \frac{1}{2} \int_0^T (b')^2(X_u) du\right)\right) ds,$$

and the sufficient condition for Assumption A.3 to be valid is that for any $p \in \mathbb{N}$, there exists a real deterministic constant $D_p > 0$ such that

$$\mathrm{E}\left[\exp\left(-p\left(\left(\int_0^T b'(X_u) dW_u\right) - \frac{1}{2} \int_0^T (b')^2(X_u) du\right)\right)\right] \leq D_p. \tag{A.7}$$

Observe that for the particular one-dimensional GBM model, condition (A.7) is clearly satisfied. Moreover, both (i) one-dimensional models with a linear or constant diffusion and (ii) multivariate models with a linear drift and constant diffusion satisfy Assumption A.3. Interestingly, the multivariate lognormal model can be observed in case (ii) (refer to [7] for further details). On the other hand, we stress that there may be some cases in which Assumption A.3 is not fulfilled, e.g., $X_T = W_T^2$, which corresponds to a system of SDEs where the diffusion coefficient does not satisfy condition (A.6). However, our method works well in such cases since, using notation of Section 2.2, we have $g(X_T) = G(y_1^2)$, and then we can apply our numerical smoothing technique to obtain a highly smooth integrand. Finally, we emphasize that additional investigation on the sufficient conditions for our smoothness Theorem 3.1 to be valid in high dimensions is an open problem and is not in the scope of this work.

B More Details on the Error and Work Discussion of ASGQ Method

Here, we show that under certain conditions of the regularity parameters p and s, we can achieve $\mathrm{Work}_{\mathsf{ASGQ}} = \mathcal{O}\left(\mathrm{TOL}^{-1}\right)$ under the best scenario ($p, s \gg 1$). In fact, using the method of Lagrange multipliers, we obtain

$$M_{ASGQ} \propto \Delta t^{\frac{p+s-ps}{p(ps+p+s)}}, \quad \text{and} \quad M_{\mathsf{lag}} \propto \Delta t^{\frac{p+s-ps}{s(ps+p+s)}},$$

and using the constraint in (3.10), we can easily show that for an error tolerance TOL, we achieve $\Delta t = \mathcal{O}\left(\text{TOL}^{\frac{ps+p+s}{ps-p-s}}\right)$. Therefore, the optimal work $\text{Work}_{\text{ASGQ}}$ solution of (3.10) satisfies

$$\begin{aligned}\text{Work}_{\text{ASGQ}} \propto M_{\text{ASGQ}} \times M_{\text{Lag}} \times \Delta t^{-1} &\propto \Delta t^{-1}\Delta t^{\frac{p+s-ps}{s(ps+p+s)}}\Delta t^{\frac{p+s-ps}{p(ps+p+s)}} \\ &\propto \text{TOL}^{-1-\frac{2(p+s)}{ps-p-s}-\frac{1}{p}-\frac{1}{s}} \\ &= \mathcal{O}\left(\text{TOL}^{-1}\right), \quad \text{since } p, s \gg 1.\end{aligned}$$

C Simulation Schemes for the Heston Dynamics

C.1 Modified Euler scheme

The forward Euler scheme can be used to simulate the Heston model. Many solutions to avoid the problems arising from the use of negative values of the volatility process v_t in (4.1) have been reported in the literature [21]. In Table C.1, we introduce f_1, f_2, and f_3, which imply different schemes when different choices are adopted. The use of the forward Euler scheme to discretize (4.1) yields

$$\begin{aligned}\widehat{S}_{t+\Delta t} &= \widehat{S}_t + \mu\widehat{S}_t\Delta t + \sqrt{\widehat{V}_t\Delta t}\widehat{S}_t Z_s \\ \widehat{V}_{t+\Delta t} &= f_1(\widehat{V}_t) + \kappa(\theta - f_2(\widehat{V}_t))\Delta t + \xi\sqrt{f_3(\widehat{V}_t)\Delta t}Z_V \\ \widehat{V}_{t+\Delta t} &= f_3(\widehat{V}_{t+\Delta t}),\end{aligned}$$

where Z_s and Z_V are two correlated standard normal rdvs with correlation ρ.

Scheme	f_1	f_2	f_3
Full truncation scheme	$\widehat{V}_t$	$\widehat{V}_t^+$	$\widehat{V}_t^+$
Partial truncation scheme	$\widehat{V}_t$	$\widehat{V}_t$	$\widehat{V}_t^+$
Reflection scheme	$\lvert\widehat{V}_t\rvert$	$\lvert\widehat{V}_t\rvert$	$\lvert\widehat{V}_t\rvert$

Table C.1: Different variants for the forward Euler scheme for the Heston model. $\widehat{V}_t^+ = \max(0, \widehat{V}_t)$.

[21] suggests that the full truncation scheme is the optimal option in terms of the weak error convergence. Therefore, we use this variant of the forward Euler scheme.

C.2 Moment-matching scheme

We consider the moment-matching scheme suggested by Andersen and Brotherton-Ratcliffe [3] (herein, called the ABR scheme). This scheme assumes that the variance v_t is locally lognormal, and the parameters are determined such that the first two moments of the discretization coincide with the theoretical moments:

$$\widehat{V}(t+\Delta t) = \left(e^{-\kappa\Delta t}\widehat{V}(t) + \left(1-e^{-\kappa\Delta t}\right)\theta\right)e^{-\frac{1}{2}\Gamma(t)^2\Delta t + \Gamma(t)\Delta W_v(t)}$$

$$\Gamma^2(t) = \Delta t^{-1}\log\left(1 + \frac{\frac{1}{2}\xi^2\kappa^{-1}\widehat{V}(t)(1-e^{-2\kappa\Delta t})}{\left(e^{-\kappa\Delta t}\widehat{V}(t) + (1-e^{-\kappa\Delta t})\theta\right)^2}\right).$$

As reported in [21], the scheme, being very easy to implement, is more effective than many of the Euler variants presented in Section C.1; however, it was observed that this scheme shows a nonrobust weak-error behavior w.r.t. the model parameters.

C.3 Heston OU-based scheme

It is well known that any OU process is normally distributed. Thus, the sum of n squared OU processes is chi-squared distributed with n degrees of freedom, where $n \in \mathbb{N}_+$. Let us define $\mathbf{X}$ as a n-dimensional vector-valued OU process with

$$\mathrm{d}X_t^i = \alpha X_t^i \mathrm{d}t + \beta \mathrm{d}W_t^i, \tag{C.1}$$

where $\mathbf{W}$ is a n-dimensional vector of independent Brownian motions.

We also define the process Y_t as

$$Y_t = \sum_{i=1}^{n} \left(X_t^i\right)^2.$$

Then, using the fact that

$$\mathrm{d}\left(X_t^i\right)^2 = 2X_t^i \mathrm{d}X_t^i + 2\mathrm{d}\langle X^i\rangle_t = \left(2\alpha\left(X_t^i\right)^2 + \beta^2\right)\mathrm{d}t + 2\beta X_t^i \mathrm{d}W_t^i,$$

we can write, using the independence of the Brownian motions:

$$\mathrm{d}Y_t = \mathrm{d}\left(\sum_{i=1}^{n}\left(X_t^i\right)^2\right) = \sum_{i=1}^{n}\mathrm{d}\left(X_t^i\right)^2 = \left(2\alpha Y_t + n\beta^2\right)\mathrm{d}t + 2\beta\sum_{i=1}^{n} X_t^i \mathrm{d}W_t^i. \tag{C.2}$$

Furthermore, the process $Z_t = \int_0^t \sum_{i=1}^n X_u^i \mathrm{d}W_u^i$ is a martingale with quadratic variations

$$\langle Z\rangle_t = \int_0^t \sum_{i=1}^{n}\left(X_u^i\right)^2 \mathrm{d}u = \int_0^t Y_u \mathrm{d}u.$$

Consequently, using the Lévy's characterization theorem, the process $\widetilde{W}_t = \int_0^t \frac{1}{\sqrt{Y_u}}\sum_{i=1}^n X_u^i \mathrm{d}W_u^i$ is a Brownian motion.

Finally, we obtain

$$\begin{aligned} \mathrm{d}Y_t &= \left(2\alpha Y_t + n\beta^2\right)\mathrm{d}t + 2\beta\sqrt{Y_t}\mathrm{d}\widetilde{W}_t \\ &= \kappa\left(\theta - Y_t\right)\mathrm{d}t + \xi\sqrt{Y_t}\mathrm{d}W_t, \end{aligned} \tag{C.3}$$

where $\kappa = -2\alpha$, $\theta = -n\beta^2/2\alpha$ and $\xi = 2\beta$.

Equations (C.1), (C.2), and (C.3) show that to simulate the process Y_t given by (C.3), we can simulate the OU process $\mathbf{X}$ with dynamics (C.1) such that its parameters (α, β) are expressed in terms of those of the process Y_t:

$$\alpha = -\frac{\kappa}{2}, \quad \beta = \frac{\xi}{2}, \quad n = \frac{-2\theta\alpha}{\beta^2} = \frac{4\theta\kappa}{\xi^2}.$$

Consequently, we can simulate the volatility of the Heston model using a sum of OU processes.

Remark C.1. The previous derivation can be generalized to cases where n^* is not an integer by considering a time-change of a squared Bessel process (refer to Chapter 6 in [19] for details). An alternative way to generalize the scheme for any noninteger n^* is to consider $n^* = n + p,\ p \in (0,1)$, and then compute $\mathrm{E}\left[g(X_{n^*})\right]$ for any observable g as follows

$$\mathrm{E}\left[g(X_{n^*})\right] \approx (1-p)\mathrm{E}\left[g(X_n)\right] + p\mathrm{E}\left[g(X_{n+1})\right].$$

C.4 On the choice of the simulation scheme of the Heston model

We determine the optimal scheme for simulating the Heston model defined in (4.1). In our setting, an optimal scheme is characterized by two properties: (i) the behavior of mixed rate convergence (Section C.4.1), which is

an important requirement for the optimal performance of ASGQ and (ii) the weak-error behavior (Section C.4.2) to apply the Richardson extrapolation when necessary.

Although we tested many parameter sets and obtained consistent numerical observations; for illustration, we only show the results for the single call option based on the Heston model with parameters listed in Table 4.1. This set corresponds to $n = 1$, where n represents the number of OU processes used in the Heston OU-based scheme (Section C.3). Furthermore, this set does not satisfy the Feller condition, *i.e.*, $4\kappa\theta > \xi^2$.

C.4.1 Comparison of different schemes in terms of mixed difference rates

Before comparing different schemes in terms of the convergence of the mixed difference rates, we recall some important notations from [6] for the ASGQ estimator. Using the same notations and construction as those used in Section 4.1 in [6], the ASGQ estimator used for approximating (2.15), and using a set of multi-indices $\mathcal{I} \subset \mathbb{N}^{dN-1}$ is given by

$$Q_{\mathcal{I}}^{\text{ASGQ}} = \sum_{\boldsymbol{\beta}\in\mathcal{I}} \Delta Q^{\boldsymbol{\beta}}, \quad \text{with} \quad \Delta Q^{\boldsymbol{\beta}} = \left(\prod_{i=1}^{dN-1} \Delta_i \right) Q^{\boldsymbol{\beta}},$$

and

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

where $Q^{\boldsymbol{\beta}}$ is the Cartesian quadrature operator with $m(\beta_i)$ points along the ith dimension.

As emphasized in [17, 6], one important requirement to achieve the optimal performance of the ASGQ is to check the error convergence of the first and mixed difference operators, as expressed by the error contribution (C.4). This is a measure of how much the quadrature error would decrease after the addition of $\Delta Q^{\boldsymbol{\beta}}$ to the ASGQ estimator $Q_{\mathcal{I}}^{\text{ASGQ}}$

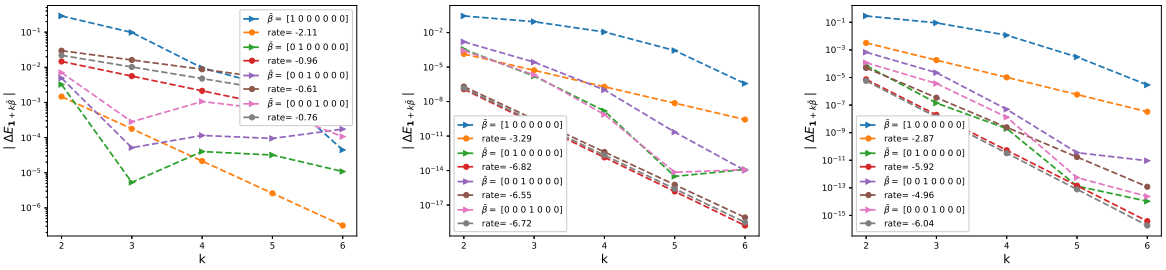
$$\Delta E_{\boldsymbol{\beta}} = \left| Q_{\mathcal{I}\cup\{\boldsymbol{\beta}\}}^{\text{ASGQ}} - Q_{\mathcal{I}}^{\text{ASGQ}} \right|. \tag{C.4}$$

The ASGQ method exhibits optimal behavior if (i) $\Delta \text{E}_{\boldsymbol{\beta}}$ decreases exponentially fast w.r.t. β_i and (ii) $\Delta \text{E}_{\boldsymbol{\beta}}$ has a product structure so that a faster error decay is observed for second differences compared to the corresponding first difference operators.

In this section, we compare the three approaches of simulating Heston dynamics: (i) the full truncation scheme (Section C.1), (ii) the ABR scheme (Section C.2), and (iii) the Heston OU-based scheme (Section C.3). The mixed difference convergences of these approaches are compared. In our numerical experiments, we only observe the differences in the mixed difference rates related to the volatility coordinates because we use schemes that only differ in the way in which they simulate the volatility process. Figure C.1 shows a comparison of the first differences rates related to the volatility coordinates for different schemes. From this figure, we see that the full truncation scheme is the worst scheme and the Heston OU-based and the ABR schemes show very good performance in terms of the speed of mixed rate convergence.

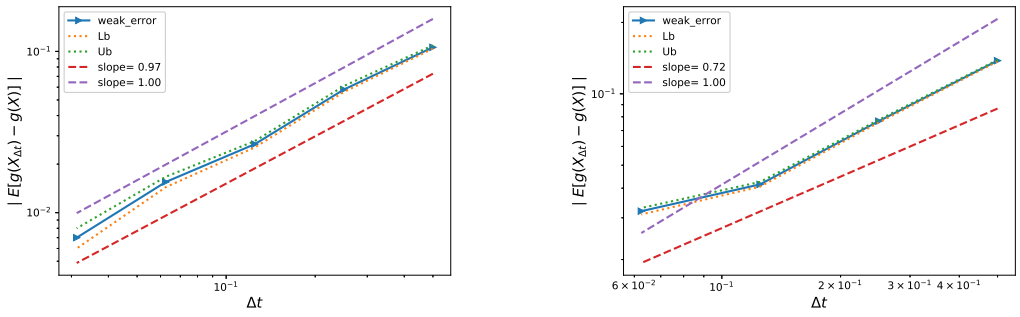
C.4.2 Comparison in terms of the weak error behavior

Herein, we compare the ABR scheme discussed in Section C.2 and the Heston OU-based scheme discussed in Section C.3 in terms of the weak-error convergence. We select the scheme with weak error rate of order one ($\mathcal{O}(\Delta t)$) in the preasymptotic regime to efficiently employ the Richardson extrapolation in our proposed methods. Figure C.2 shows a comparison of the weak error rates for the different schemes. From this figure, the Heston OU-based scheme exhibits a better weak convergence rate that is closer to 1 than the ABR scheme, which shows a weak error rate of 0.7.



(a) (b) (c)

Figure C.1: Rate of error convergence of the first-order differences $|\Delta \mathrm{E}_{\boldsymbol{\beta}}|$, defined in (C.4), ($\boldsymbol{\beta} = \mathbf{1} + k\overline{\boldsymbol{\beta}}$) for the single call option under the Heston model. The parameters are given in Set 1 in Table 4.1, and the number of time steps $N = 4$. We only show the first four dimensions that are used for the volatility noise (mainly dW_v in (4.1)). (a) Full truncation scheme, (b) ABR scheme, and (c) Heston OU-based scheme.



(a) (b)

Figure C.2: Weak error convergence for the single call option under the Heston model for the parameters listed in Table 4.1. The upper and lower bounds are 95% confidence intervals. (a) Heston OU-based scheme, (b) ABR scheme.