SMOOTHING THE PAYOFF FOR EFFICIENT COMPUTATION OF BASKET OPTION PRICING

CHRISTIAN BAYER, MARKUS SIEBENMORGEN, AND RAUL TEMPONE

Abstract. We consider the problem of pricing basket options in a multivariate Black Scholes or Variance Gamma model. From a numerical point of view, pricing such options corresponds to moderate and high dimensional numerical integration problems with non-smooth integrands. Due to this lack of regularity, higher order numerical integration techniques may not be directly available, requiring the use of methods like Monte Carlo specifically designed to work for non-regular problems. We propose to use the inherent smoothing property of the density of the underlying in the above models to mollify the payoff function by means of an exact conditional expectation. The resulting conditional expectation is unbiased and yields a smooth integrand, which is amenable to the efficient use of adaptive sparse grid cubature. Numerical examples indicate that the high-order method may perform orders of magnitude faster compared to Monte Carlo or Quasi Monte Carlo in dimensions up to 25.

1. Introduction

In quantitative finance, the price of an option on an underlying S can typically—disregarding discounting—be expressed as $E[f(S)]$ for some (payoff) function f on S and the expectation operator E induced by the appropriate pricing measure. Hence, option pricing is an integration problem. The integration problem is usually challenging due to a combination of two complications:

- S often takes values in a high-dimensional space. The reason for the high-dimensionality may be time-discretization of a stochastic differential equation, path dependence of the option (i.e., S is actually a path of an asset price, not the value at a specific time), a large number of underlying assets, or others.
- the payoff function f is typically not smooth.

In this work, we shall focus on the problem of pricing basket options in models, where the distribution of the underlying is explicitly given to us (more precisely, we consider multivariate Black-Scholes and Variance-Gamma models), i.e., no time-discretization is required. We consider a basket option on a d-dimensional underlying asset $S_T = \left(S_T^1, \dots, S_T^d\right)$ with payoff function

$$f(S_T) = \left(\sum_{i=1}^{d} w_i S_T^i - K \right)^+$$

2010 *Mathematics Subject Classification.* Primary: 91G60; Secondary: 65D30, 65C20.

Key words and phrases. Computational Finance, European Option Pricing, Multivariate approximation and integration, Sparse grids, Stochastic Collocation methods, Monte Carlo and Quasi Monte Carlo methods.

R. Tempone is a member of the KAUST Strategic Research Initiative, Center for Uncertainty Quantification in Computational Sciences and Engineering.

We are grateful to Juho Häppölä for pointing out the immediate applicability of our method to Variance-Gamma processes.

for some positive weights $w_1, \dots, w_d$, a maturity T and a strike price K. Observe in passing that one could also allow some weights being negative, an option type known as "spread option". Note in addition that (discrete) Asian options also fall under this framework.

Even in the standard Black-Scholes framework, closed-form expressions for basket option prices are not available, since sums of log-normal random variables are generally not log-normally distributed. Some explicit approximation formulas are based on approximate distributional identities of sums of lognormal random variables, see, for instance, [25, 11]. Also Laplace's method, possibly coupled with heat kernel expansions when the distribution of the factors S_T^i are given only as solutions of SDEs, has been shown to yield highly exact results even in high dimensions, see [4, 6, 5]. In this work, however, we aim at solving the problem at hand using generic numerical integration techniques, which remain available beyond the restrictions of the previous methods.

Efficient numerical integration algorithms are available even in high dimensions, but they usually require smoothness of the integrand. Hence, they are a-priori not applicable in many option pricing problems. We will specifically focus on (adaptive) sparse grid methods, see e.g. [7, 16].

Another efficient numerical integration technique is quasi-Monte Carlo. Formally, QMC methods also rely on smoothness of the integrand to retain first order convergence (up to multiplicative logarithmic terms), but it has been observed for some time that QMC typically works very well for integration problems in quantitative finance, even when the theoretically required regularity of the integrand is not satisfied, see, for instance, [26] for an overview. In a series of works, Griebel, Kou and Sloan [19, 20, 21] have analyzed the good performance of QMC methods for typical option pricing problems based on the ANOVA decomposition. In particular, they show that all terms of the ANOVA decomposition are smooth except for the last one. In the context of barrier options, Achtsis, Cools and Nuyens [1, 2] successfully applied QMC using a conditional sampling strategy to fulfill the barrier conditions. Moreover, they use a root finding procedure to determine the region where the payoff function of the option is positive. In other words, this root finding procedure, which has been discussed in e.g. [17, 24], locates the unsmooth part of the payoff function. Note that the boundary of the support of the payoff function may be quite complicated in terms of the coordinates for the integration problem, an issue that may limit the applicability of such approach.

From a numerical analysis point of view, the obvious solution to the problem seems to be to smoothen the integrand using standard mollifiers, and there is a prominent history of successful application of mollification in quantitative finance, see, for instance, [12] in the context of computing sensitivities of option prices. For many financial applications there seems a more attractive approach which avoids the balancing act between providing the smoothness needed for the numerical integration algorithm and introducing bias in the integrand. Indeed, we suggest to use the smoothing property of the distribution of the underlying itself for regularizing the integrand. This technique is quite standard in a time-stepping setting, and we indeed plan to explore its applicability in that context in the future.

In this work, however, the regularization will be achieved by integrating against one factor of the multivariate geometric Brownian motion first—conditioning on all the other factors. More specifically, we show in Section 3 below that we can always decompose

$$\sum_{i=1}^{d} w_i S_T^i \overset{\mathcal{L}}{=} H e^Y$$

for two independent random variables H and Y—for the precise, explicit construction see Lemma 3.3 together with Lemma 3.1. Here, the random variable Y is normally distributed. Therefore, by computing the conditional expectation given H, the basket option valuation problem is reduced to an integration problem in H (corresponding to an integration in $\mathbb{R}^{d-1}$) with a payoff function given in this case by the Black Scholes formula, a smooth function.

The idea of integrating out one factor first, thereby obtaining an "option" on the remaining factors with payoff function giving by the Black Scholes formula is not new in finance. For instance, Romano and Touzi [31] have applied this idea in a theoretical study of stochastic volatility models as a tool to show convexity of admissible prices. In this vein, see also the work [13]. The above mentioned decomposition (allowing the use of this trick in the basket option context), however, seems new. As conditional expectations always reduce the variance of a random variable, this trick can also be useful in a Monte Carlo setting as well.

Let us point out already here that the smoothing approach proposed in this work can be applied in more general situation, possibly in modified ways, including more complicated models, where the asset price process can only be simulated by a time-stepping procedure. In that case, we may no longer obtain an explicit, exact formula for the smoothed payoff through the conditional expectation step. Still, a properly constructed numerical quadrature to the conditional expectation will still inherit the fast convergence rates.

Outline. We start by describing the setting of the problem in more detail. In Section 2 we recall two popular efficient numerical integration techniques for high dimensions, namely (adaptive) sparse grids and quasi Monte Carlo. Then, in Section 3 we describe the smoothing of the payoff in the multivariate Black Scholes framework. Confirming the exploratory style of this work, we give two detailed numerical examples. In Section 4 we present numerical results for the multivariate Black Scholes model, and in Section 5 we consider a multivariate Variance Gamma model, indicating that the smoothing method proposed here is applicable beyond the standard Black Scholes regime. Afterwards, we present some concluding remarks including an outlook on future research.

Setting. We consider a European basket option in a Black-Scholes model. More specifically, assume that the interest rate $r = 0$ – i.e., we are working with forward prices. We consider $d \in \mathbb{N}$ assets with prices $S_t = \left(S_t^1, \dots, S_t^d\right)$, $t > 0$, with risk-neutral dynamics

$$dS_t^i = \sigma_i S_t^i dW_t^i, \quad i = 1, \dots, d, \tag{1}$$

for volatilities $\sigma_i > 0$, $i = 1, \dots, d$, driven by a correlated d-dimensional Q- Brownian motion W with

$$d\left\langle W^i, W^j \right\rangle_t = \rho_{i,j} dt, \quad i, j = 1, \dots, d.$$

Obviously, (1) has the explicit solution

$$S_t^i = S_0^i \exp\left(-\frac{1}{2}\sigma_i^2 t + \sigma_i W_t^i\right), \quad i = 1, \dots, d,\ t > 0. \tag{2}$$

We note that the components of the random vector S_t have log-normal distributions and are correlated.

A basket option is an option on such a collection of assets. We assume a standard call option with strike $K > 0$ and maturity $T > 0$ with price

$$C_{\mathcal{B}} \coloneqq E\left[\left(\sum_{i=1}^{d} c_i S_T^i - K\right)^+\right]. \tag{3}$$

Let us next transform the pricing problem (3) into a slightly more abstract form. As already observed, the random vector $\left(c_1 S_T^1, \dots, c_d S_T^d\right)$ can be represented as $\left(w_1 e^{X_1}, \dots, w_d e^{X_d}\right)$ for scalars $w_1, \dots, w_d$ and a zero-mean Gaussian vector $X = (X_1, \dots, X_d) \sim \mathcal{N}(0, \Sigma)$. Indeed, we may choose

$$w_i = c_i S_0^i e^{-\frac{1}{2}\sigma_i^2 T}, \quad i = 1, \dots, d,$$
$$\Sigma_{i,j} = \sigma_i \sigma_j \rho_{i,j} T, \quad i, j = 1, \dots, d.$$

Therefore, we are left with the problem of computing

$$E\left[\left(\sum_{i=1}^{d} w_i e^{X_i} - K\right)^+\right] \tag{4}$$

for $X \sim \mathcal{N}(0, \Sigma)$ and $d > 1$.

Remark 1.1. Note that the problem of computing the price of a (discretely-monitored) Asian option on a (one-dimensional) Black-Scholes asset is of the form (4) as well, but with different covariance matrix Σ.

In Section 5, we will, also consider a Variance Gamma model, see [29] for the univariate and [27] for the multivariate Variance Gamma model. We first recall the univariate case: Let

$$X_t := \theta t + \sigma W_{\gamma_t} \tag{5}$$

for a standard Brownian motion W and an independent Γ process γ_t with parameters 1 and ν, i.e., γ is a process with stationary, independent increments with $\gamma_{t+h} - \gamma_t$ Γ-distributed with mean h and variance νh, for any $h > 0$, $t > 0$. Additionally, we impose $\gamma_0 = 0$. Under the risk-neutral measure with $r = 0$ (for simplicity), we then consider the asset price process

$$S_t = S_0 \exp\left(\omega t + X_t\right), \quad \omega = \frac{\log(1 - \theta\nu - \sigma^2\nu/2)}{\nu}, \tag{6}$$

see [29, formula (22)]. Notice that the process X is a Lévy process and can alternatively be described as the difference of two independent Γ processes.

Economically, the time change γ is often interpreted as "business" or "trading" time. Hence, it makes sense to assume that different stocks are subject to a single time change. A reasonable multivariate generalization of the Variance Gamma model (also adopted in [27]) consists in defining log terms X_t^i as in (5) based on *correlated* Brownian motions W_t^i, parameters θ_i, σ_i, but a common Γ-process γ_t (hence, with a fixed parameter ν). The stock price components S_t^i, $i = 1, \dots, d$, are then defined according to (6) based on X_t^i, θ_i, σ_i, but the common parameter ν.

2. A reminder on efficient multi-dimensional numerical integration

In this section, we give a brief review on efficient multi-dimensional integration schemes, in particular the Monte Carlo quadrature, the quasi-Monte Carlo quadrature and the adaptive sparse grid quadrature. To this end, let us consider a function $f\colon \mathbb{R}^d \to \mathbb{R}$ and denote the d-dimensional standard Gaussian density function by $\phi_d\colon \mathbb{R}^d \to \mathbb{R}_+$, $x \mapsto (2\pi)^{-d/2} \prod_{k=1}^d \exp(-x_k^2/2)$. As we will see later on, the multi-dimensional integration problem that we are faced with is to find an approximation to the integral

$$\int_{\mathbb{R}^d} f(x)\phi_d(x)\,\mathrm{d}x. \tag{7}$$

2.1. **Monte Carlo and quasi-Monte Carlo quadrature.** The most widely used quadrature technique to tackle high-dimensional integration problems is the Monte Carlo quadrature, see e.g. [22]. This quadrature draws $N \in \mathbb{N}$ independently and identically distributed samples $\xi_i \in \mathbb{R}^d,\ i = 1, \dots, N$ with respect to the d-dimensional standard normal distribution. Then, the unbiased Monte Carlo estimator for the integral (7) is given by

$$\int_{\mathbb{R}^d} f(x)\phi_d(x)\,\mathrm{d}x \approx \frac{1}{N}\sum_{i=1}^{N} f(\xi_i). \tag{8}$$

The big advantage of this quadrature is that it converges with a rate which is independent of the dimensionality d, but the convergence rate $O(N^{-1/2})$ is rather low. Another advantage of this quadrature is that it works under low regularity requirements on the integrand. To be more precisely, the variance of the integrand is a multiplicative constant in the error estimate.

The quasi-Monte Carlo quadrature is of the same form (8) as the Monte Carlo quadrature, but the sample points x_i are constructed or taken from a predescribed sequence and not chosen randomly. There are several quasi-Monte Carlo sequences available in the literature, see e.g. [8, 30]. Nevertheless, almost all the quasi-Monte Carlo sequences refer to integration over the unit cube $[0, 1]^d$ with respect to the Lebesgue measure and, hence, these points have to be mapped to the domain of integration Γ by the inverse normal distribution. The aim of a quasi-Monte Carlo sequence is to mirror with the first N sample points the uniformly distribution on the unit cube as good as possible. A measure of the distance between the uniformly distribution and the first N sample points is then given by the discrepancy of these sample points, see [30]. The reason for this is that the quasi-Monte Carlo integration error for functions with bounded variation in the sense of Hardy and Krause can be estimated up to a constant by the discrepancy of the integration points. A quasi-Monte Carlo sequence is called a low-discrepancy sequence if the discrepancy of the first N points of this sequence is $O(N^{-1}\log(N)^d)$. Thus, for the quasi-Monte Carlo quadrature based on low-discrepancy sequences can improve the convergence of the Monte-Carlo quadrature. In our numerical examples, we will use the quasi-Monte Carlo quadrature based on the Sobol-sequence, cf. [33], which is a classical low-discrepancy sequence.

2.2. **Adaptive sparse grid quadrature.** The construction of a sparse grid quadrature is based on a sequence of one-dimensional quadrature rules, cf. [7, 32]. Hence, we define for a function $f\colon \mathbb{R} \to \mathbb{R}$ quadrature rules

$$\int_{\mathbb{R}} f(x)\phi_1(x)\,\mathrm{d}x \approx Q_j(f) = \sum_{i=1}^{N_j} w_i^{(j)} f\left(\eta_i^{(j)}\right), \quad N_j \in \mathbb{N}, \quad j = 0, 1, \dots \tag{9}$$

with suitable quadrature points and weights $\left\{\left(\eta_i^{(j)}, w_i^{(j)}\right)\right\}_{i=1}^{N_j} \subset \mathbb{R}\times\mathbb{R}$. Usually, the sequence of quadrature rules is increasing, i.e $N_0 < N_1 < \dots$, and the first quadrature uses only one quadrature point and weight, i.e. $N_0 = 1$. According to the sequence $\{Q_j\}_j$, we introduce the difference quadrature operator

$$\Delta_j := Q_j - Q_{j-1}, \quad \text{where} \quad Q_{-1} := 0. \tag{10}$$

Assume that the sequence $\{Q_j f\}_j$ converges, i.e

$$\int_{\mathbb{R}} f(x)\phi_1(x)\,\mathrm{d}x = \lim_{j\to\infty} Q_j f = \lim_{n\to\infty}\sum_{j=0}^{n} \Delta_j f.$$

This implies, that the sequence $\{|\Delta_j f|\}_j$ converges to zero and, hence, the importance of the difference quadrature operators decays in j. Unfortunately, this decay is not necessarily monotonic, but it builds the basic idea of adaptive sparse grid constructions.

With the difference quadrature operators Δ_j at hand, a generalized sparse grid quadrature for the integration problem (7) is defined by

$$\int_{\mathbb{R}^d} f(x)\phi_d(x)\,\mathrm{d}x \approx \sum_{\alpha\in\mathcal{I}} \Delta_\alpha f := \sum_{\alpha\in\mathcal{I}} \Delta_{\alpha_1}\otimes\Delta_{\alpha_2}\otimes\cdots\otimes\Delta_{\alpha_d} f \tag{11}$$

for an admissible index set $\mathcal{I}\subset\mathbb{N}_0^d$. Such an index set $\mathcal{I}$ is called admissible if it holds for $j=1,\dots,n$ and the unit multi-index e_j that

$$\alpha\in\mathcal{I} \Longrightarrow \alpha - e_j \in \mathcal{I} \quad \text{if } \alpha_j > 0.$$

As can be seen from (10) and (11), a generalized sparse grid quadrature is uniquely determined by a sequence of univariate quadrature rules $\{Q_j\}_j$ and an admissible index set $\mathcal{I}$. The index set $\mathcal{I}$ can be chosen a priori, for example as

$$\mathcal{I} = \left\{\alpha\in\mathbb{N}_0^d : \sum_{i=1}^n \alpha_i \le q\right\} \tag{12}$$

which corresponds to a total degree sparse grid on level q.

Another option is to adapt $\mathcal{I}$ a posteriori. In this case an initial index set is selected, most often $\mathcal{I}_0 = \{(0,\dots,0)\}$. Then, the integration error of the sparse grid quadrature with respect to $\mathcal{I}_0$ is estimated by a local error estimator and, afterwards, the indices with the largest local error estimator are successively added to $\mathcal{I}_0$ until a global error estimator η has reached a certain tolerance. We denote the local error estimator of an index $\alpha\in\mathcal{I}$ by g_α and use for our purpose simply the absolute value of the associated difference quadrature formula, i.e. $g_\alpha := |\Delta_\alpha f|$. Of course, we have to guarantee during the algorithm that the admissible condition of $\mathcal{I}$ is not violated. The detailed description of this method is provided in [16]. We recall here the algorithm from [16] and explain the most important steps.

In Algorithm 1, the index set $\mathcal{I}$ in (11) is partitioned into the *old index set* $\mathcal{O}$ and the *active index set* $\mathcal{A}$. The active index sets contains all indices α whose local error estimators g_α actually contribute to the global error estimator η. Then, the element α of $\mathcal{A}$ with the largest local error estimator is removed from the active to the old index set and the children of α, i.e. $\alpha+e_j$, are successively added to the active index set, as long as all their parents belong to the old index set. The last step is necessary in order to guarantee the admissibility condition. Then, the contribution of the new indices to the value of the integral as well as the local and global error estimator is updated and the procedure is repeated. The change in the current index sets during one step of the algorithm is visualized in Figure 1.

We will use as one-dimensional sequences Gauß-Hermite and Genz-Keister quadrature rules, cf. [15]. Gauß-Hermite quadrature rules have the highest degree of polynomial exactness for integrals as in (9) while Genz-Keister rules have the advantage that they are nested. More precisely, the Genz-Keister rules are extensions of Gauß-Hermite quadrature rules of relatively low degree. As the first extension of the one point Gauß-Hermite quadrature we use the three point Gauß-Hermite quadrature. Nevertheless, the further extensions do not coincide with any other Gauß-Hermite quadrature rule.

At the end of this section, we visualize on the right-hand side of Figure 2 the 2-dimensional adaptive sparse grid points which are used in our first numerical example. On the left-hand side of Figure 2, we show the associated adaptive index set.

Algorithm 1 Adaptive sparse grid quadrature for a function f

$\alpha \leftarrow (0, \ldots, 0)$
$\mathcal{O} \leftarrow \emptyset$
$\mathcal{A} \leftarrow \alpha$
$y \leftarrow \Delta_\alpha f$
$\eta \leftarrow g_\alpha$
while ($\eta >$ TOL) **do**
  select α from $\mathcal{A}$ with largest g_α
  $\mathcal{A} \leftarrow \mathcal{A} \setminus \alpha$
  $\mathcal{O} \leftarrow \mathcal{O} \cup \alpha$
  $\eta \leftarrow \eta - g_\alpha$
  for ($k = 1, \ldots, d$) **do**
    $\beta \leftarrow \alpha + e_k$
    if ($\beta - e_q \in \mathcal{O}$ for all q) **then**
      $\mathcal{A} \leftarrow \mathcal{A} \cup \beta$
      $x \leftarrow \Delta_\beta f$
      $y \leftarrow y + x$
      $\eta \leftarrow \eta + g_\beta$
    end if
  end for
end while
return y

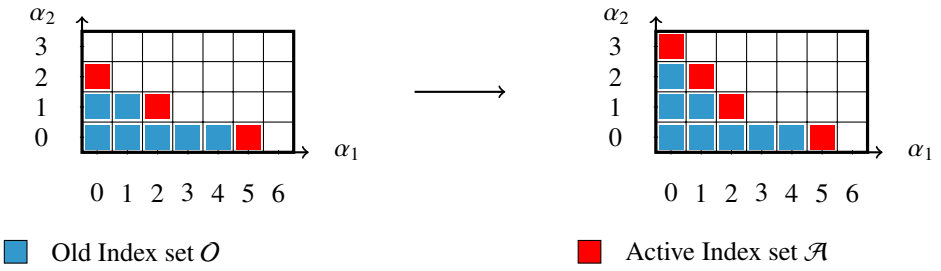

FIGURE 1. One step of the adaptive quadrature where $\alpha = (0, 2)$ is the index with largest $g(\alpha)$.

Remark 2.1. A further alternative could be to use multi-dimensional cubature formulas, see, for instance, [9]. In principle, high-order cubature formulas also require smoothness of the integrand, therefore we suspect that the approach presented here will also work well in the cubature context. Still, we do not further discuss these methods in the current paper.

3. SMOOTHING THE PAYOFF

In this section, we will describe a simple technique for smoothing the integrand in (4) which, at the same time,

- produces an analytic integrand;
- does not introduce a bias error;
- reduces the variance of the resulting integrand.

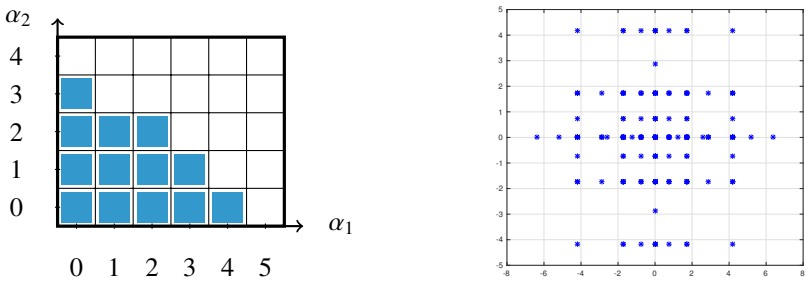

FIGURE 2. Index set $\mathcal{I}$ of the sparse grid on the left and associated sparse grid points, which are used in the first numerical example, on the right .

For the following, we assume that the covariance matrix Σ is invertible, i.e., a positive definite symmetric matrix.

The general idea is that we want to integrate out one Gaussian factor in (4), conditioning on the remaining $d-1$ factors. Clearly, the outcome of such a procedure is a smooth function of the remaining factors. However, generically there is no closed formula for this function. The reason for this is that there is no closed formula for the simple special case

$$E\left[\left(e^{\sigma_1 Z} + e^{\sigma_2 Z} - K\right)^+\right]$$

for $Z \sim \mathcal{N}(0,1)$ and $\sigma_1 \neq \sigma_2$. Indeed, $e^{\sigma_1 Z} + e^{\sigma_2 Z}$ has a log-normal distribution if and only if $\sigma_1 = \sigma_2$. In this case, the above expression is given in terms of the celebrated Black-Scholes formula, to be reviewed below.

It turns out, that a clever choice of factorization of the covariance matrix of the Gaussian factors allows us to factor out one common, independent log-normal term. This is a consequence of the

Lemma 3.1. *Let Σ be a symmetric, positive definite $d \times d$ matrix. Then there is a diagonal matrix $D = \operatorname{diag}\left(\lambda_1^2, \lambda_d^2, \ldots, \lambda_d^2\right)$ and an invertible matrix $V \in \mathbb{R}^{d\times d}$ with the property that $V_{i,1} \equiv 1$, $i = 1, \ldots, d$, such that*

$$\Sigma = VDV^\top.$$

Moreover, we may choose the remaining columns of V such that $\lambda_2^2 \geq \ldots \geq \lambda_d^2 \geq 0$.

Proof. From [3, p. 126], we know that for every $0 \neq s \in \mathbb{R}^n$ the rank-1 modification

$$\tilde{A} = A - \frac{(As)(As)^\top}{s^\top A s} \tag{13}$$

of a symmetric and positive definite matrix $A \in \mathbb{R}^{d\times d}$ yields a symmetric and positive semidefinite matrix $\tilde{A} \in \mathbb{R}^{d\times d}$ of rank $d-1$. Let us denote $\mathbf{1} = [1, \ldots, 1]^\top$ and choose $v = \Sigma^{-1}\mathbf{1}$. Then it follows from (13) that

$$\tilde{\Sigma} = \Sigma - \frac{\mathbf{1}\cdot\mathbf{1}^\top}{\mathbf{1}^\top v}$$

is a symmetric and positive semidefinite matrix of rank $d-1$. Denote by (λ_i^2, v_i) for $i = 2, \ldots, d$ the $d-1$ eigenpairs corresponding to the $d-1$ positive eigenvalues of $\tilde{\Sigma}$. Defining $V = [v_1, v_2, \ldots, v_d]$ with $v_1 = \mathbf{1}$ and $D = \operatorname{diag}(\lambda_1^2, \lambda_2^2, \ldots, \lambda_d^2)$ with $\lambda_1^2 = (\mathbf{1}^\top \mathbf{v})^{-1}$ leads to the desired result. □

Remark 3.2. Of course, the vector $\mathbf{1}$ is by no means special in Lemma 3.1 and can be replaced by any other fixed vector.

In the next step, we replace X by $Y := V^{-1}X \sim \mathcal{N}(0, D)$ and note that the components of Y are independent. Substituting the decomposition $X = VY$ into (4), we obtain

$$\begin{aligned} C_{\mathcal{B}} &= E\left[\left(\sum_{i=1}^{d} w_i e^{(VY)_i} - K\right)^+\right] \\ &= E\left[\left(\sum_{i=1}^{d} w_i \exp\left(Y_1 + \sum_{j=2}^{d} V_{i,j} Y_j\right) - K\right)^+\right] \\ &= E\left[\left(h(Y_2, \ldots, Y_d) e^{Y_1} - K\right)^+\right] \end{aligned} \tag{14}$$

with

$$h(\overline{y}) = h(y_2, \ldots, y_d) := \sum_{i=1}^{d} w_i \exp\left(\sum_{j=2}^{d} V_{i,j} y_j\right), \quad \overline{y} := (y_2, \ldots, y_d) \in \mathbb{R}^{d-1}. \tag{15}$$

Lemma 3.3 (Conditional Expectation formula)**.** *Let* $\overline{Y} = (Y_2, \ldots, Y_d) = ((V^{-1}X)_2, \ldots, (V^{-1}X)_d) \sim \mathcal{N}\left(0, \overline{D}\right)$, $\overline{D} := \operatorname{diag}(\lambda_2^2, \ldots, \lambda_d^2)$. *Then*

$$E\left[\left(\sum_{i=1}^{d} w_i e^{X_i} - K\right)^+ \middle| \overline{Y}\right] = C_{BS}\left(h(\overline{Y}) e^{\lambda_1^2/2}, K, \lambda_1\right),$$

where

$$\begin{aligned} C_{BS}(S_0, K, \sigma) &:= \Phi(d_1) S_0 - \Phi(d_2) K, \\ d_{1/2} &:= \frac{1}{\sigma}\left[\log\left(\frac{S_0}{K}\right) \pm \frac{\sigma^2}{2}\right], \end{aligned}$$

is the Black-Scholes formula for $r = 0$*, with maturity* $T = 1$.

Proof. As Y_1 and $\overline{Y}$ are independent and $Y_1 \sim \mathcal{N}(0, \lambda_1^2)$, we have

$$E\left[\left(h(Y_2, \ldots, Y_d) e^{Y_1} - K\right)^+ \middle| \overline{Y} = \overline{y}\right] = E\left[\left(h(\overline{y}) e^{\lambda_1 Z} - K\right)^+\right]$$

for some $Z \sim \mathcal{N}(0, 1)$. On the other hand, for $r = 0$ and maturity $T = 1$, the Black-Scholes formula is given by

$$C_{BS}(S_0, K, \sigma) = E\left[\left(S_0 e^{-\frac{1}{2}\sigma^2 + \sigma Z} - K\right)^+\right] = \Phi(d_1) S_0 - \Phi(d_2) K,$$

since $S_T = S_0 \exp\left(-\frac{1}{2}\sigma^2 T + \sigma B_T\right)$ for a Brownian motion B. Comparing these expressions, we see that we have to choose $K = K$, $\sigma = \lambda_1$ and $S_0 = h(\overline{y}) e^{\frac{1}{2}\lambda_1^2}$. □

Lemma 3.3 directly implies

Proposition 3.4. *The basket option price in the multi-variate Black-Scholes setting satisfies*

$$C_{\mathcal{B}} = E\left[C_{BS}\left(h\left(\sqrt{\overline{D}} Z\right) e^{\lambda_1^2/2}, K, \lambda_1\right)\right], \quad Z \sim \mathcal{N}(0, I_{d-1}), \ \sqrt{\overline{D}} = \operatorname{diag}(\lambda_2, \ldots, \lambda_d). \tag{16}$$

Remark 3.5. As remarked earlier, a similar closed form expression cannot be obtained when the first column $V_{\cdot,1}$ of the matrix V is a general d-dimensional vector. However, we may still get an explicit formula if $V_{\cdot,1}$ only takes values in $\{0,1\}$. For simplicity, let us assume that the first k entries of $V_{\cdot,1}$ are 1 and the remaining entries are 0—by Remark 3.2 this can always be achieved. The computation before Lemma 3.3 then gives

$$C_{\mathcal{B}} = E\left[\left(h_1(Y_2,\dots,Y_k)e^{Y_1} + h_2(Y_{k+1},\dots,Y_d) - K\right)^+\right],$$
$$h_1(y_2,\dots,y_k) \coloneqq \sum_{i=1}^{k} w_i \exp\left(\sum_{j=2}^{d} V_{i,j}y_j\right),$$
$$h_2(y_{k+1},\dots,y_d) \coloneqq \sum_{i=k+1}^{d} w_i \exp\left(\sum_{j=2}^{d} V_{i,j}y_j\right).$$

Conditioning again on $\overline{Y}$, we once again arrive at the Black-Scholes formula, this time requiring a shift in K, as well. In the end, we obtain

$$E\left[\left(\sum_{i=1}^{d} w_i e^{X_i} - K\right)^+ \middle| \overline{Y}\right] = C_{BS}\left(h_1(Y_2,\dots,Y_k)e^{\lambda_1^2/2}, K - h_2(Y_{k+1},\dots,Y_d), \lambda_1\right),$$

in the sense that

$$C_{BS}(S_0, K, \sigma) = S_0 - K \text{ for } K < 0.$$

In general, we therefore suggest to choose $V_{\cdot,1}$ such as to maximize the effective smoothing parameter λ_1.

Remark 3.6. Generally speaking, the decay of derivatives of the integrand in (16) depends – inter alia – on the size of $\lambda_1^2 = \left\langle \mathbf{1}, \Sigma^{-1}\mathbf{1}\right\rangle^{-1}$. If we normalize the variances of the individual components, then λ_1^2 mostly depends on the angle of $\mathbf{1}$ with the eigenspace corresponding to the largest eigenvalue of Σ.

Remark 3.7. It is worth observing that after the conditional expectation (16) one may also perform a change of measure on the resulting $d-1$ dimensions to enhance the convergence of all the quadratures discussed in this work. For instance, this is particularly important for out of the money options.

4. Numerical example 1: Multivariate Black Scholes setting

In our first numerical example, we consider the pricing problem (3) of a European basket option in a Black-Scholes model. This price depends on the strike price K, the weight vector c and the vector S_T containing the values of the different assets at the maturity T. Moreover, the distribution of S_T can be deduced from the initial values of the assets S_0, the vector of volatilities σ and the correlation matrix ρ which determine the Black-Scholes model in (1). The initial values in our examples are chosen randomly, i.e. independently and uniformly distributed from the interval $S_0^i \in [8, 20]$. The volatilities are chosen randomly as well from the interval $\sigma_i \in [0.3, 0.4]$. Following [10], the correlation matrix $\rho = \tau\tau^\top$ is given by a lower triangular matrix τ which is constructed from a random vector

$x \in \mathbb{R}^{d-1}$ with independently and uniformly distributed entries $x_i \in [0.8, 1]$ as follows:

$$\tau_1 = \begin{pmatrix} 1 \\ \mathrm{cp}(x) \end{pmatrix}, \quad \tau_2 = \sqrt{1 - x_1^2} \begin{pmatrix} 0 \\ 1 \\ \mathrm{cp}(x_{2:d-1}) \end{pmatrix}, \ \ldots, \ \tau_d = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \sqrt{1 - x_{d-1}^2} \end{pmatrix}.$$

Herein, we employed the MATLAB-inspired notation $x_{2:d-1} = [x_2, \ldots, x_{d-1}]^\top$. In addition, we denote by $\mathrm{cp}\colon \mathbb{R}^{d-1} \to \mathbb{R}^{d-1}$ the cummulative product given by

$$\mathrm{cp}(x) = [x_1, x_1 x_2, \ldots, x_1 x_2 \cdots x_{d-1}]^\top.$$

The weight vector c is chosen such that the basket is an average of the different assets, i.e. $c_i = 1/d$. Moreover, we choose three different settings for the strike price, $K = c^\top S_0$ ("at the money"), $K = 1.2 \cdot c^\top S_0$ ("out of the money") and $K = 0.8 \cdot c^\top S_0$ ("in the money").

Remark 4.1. We tested our experiments with different, randomly chosen weight vectors $c \in [0, 1]^d$ and obtained similar results. Hence, it seems that there is only a slight dependence between the weight vector in the basket and the performance of the different quadrature methods.

We compare several integration schemes applied to the original problem (4) and the smoothened problem (16). To be more precise, we consider the Monte-Carlo method, the quasi-Monte Carlo method based on Sobol points and the sparse grid method described in Section 2.

4.1. **Performance of the sparse grid methods.** In this subsection, we investigate the convergence behaviour of the adaptive sparse grid method for the smoothened problem (16) (SGBS). Therefore, we apply the (SGBS) to our model problem in dimension $d = 3$ in the "at the money case", $d = 8$ in the "out of the money case" and $d = 25$ in the "in the money case", respectively. As a reference solution, we use an adaptive sparse grid quadrature to determine (16) with a very small tolerance, i.e. $\varepsilon = 10^{-11}$ for $d = 3$, $\varepsilon = 10^{-9}$ for $d = 8$, and $\varepsilon = 10^{-7}$ for $d = 25$ respectively. As the sequence of univariate quadrature points, we use the listed Genz-Keister points from [23]. Unfortunately, there exists only 9 different Genz-Keister extensions and it might happen that a higher precision is needed in a particular direction. In this case, we use Gauß-Hermite points with a successively higher degree of precision. The one-dimensional Gauß-Hermite points and weights can be easily constructed for an arbitrary degree of precision by solving an associated eigenvalue problem, see e.g. [14] for the details.

To observe the convergence behaviour of the SGBS, we successively refine the tolerance, e.g. from 10^{-2} to 10^{-9} for $d = 3$, and compute the relative error between the corresponding approximation to (16) and the reference solution. In order to compare the results with other methods and also to validate the reference solution, we additionally apply a Monte-Carlo quadrature (MC), a quasi-Monte Carlo quadrature (QMC) and an adaptive sparse grid quadrature (SG) to the original problem (4) and compare the results with the reference solution as well. Herein, we increase the number of quadrature points for the (quasi-) Monte Carlo quadrature as $3 \cdot 6^q$ for $q = 1, \ldots, 8$. In addition, we use 20 runs of the Monte-Carlo estimator on each level q and plot the median of the relative errors to the reference solution of these 20 runs.

Remark 4.2. The convergence results are shown in terms of quadrature points. Of course, it is also interesting to compare the computational times to see the overhead of the adaptive

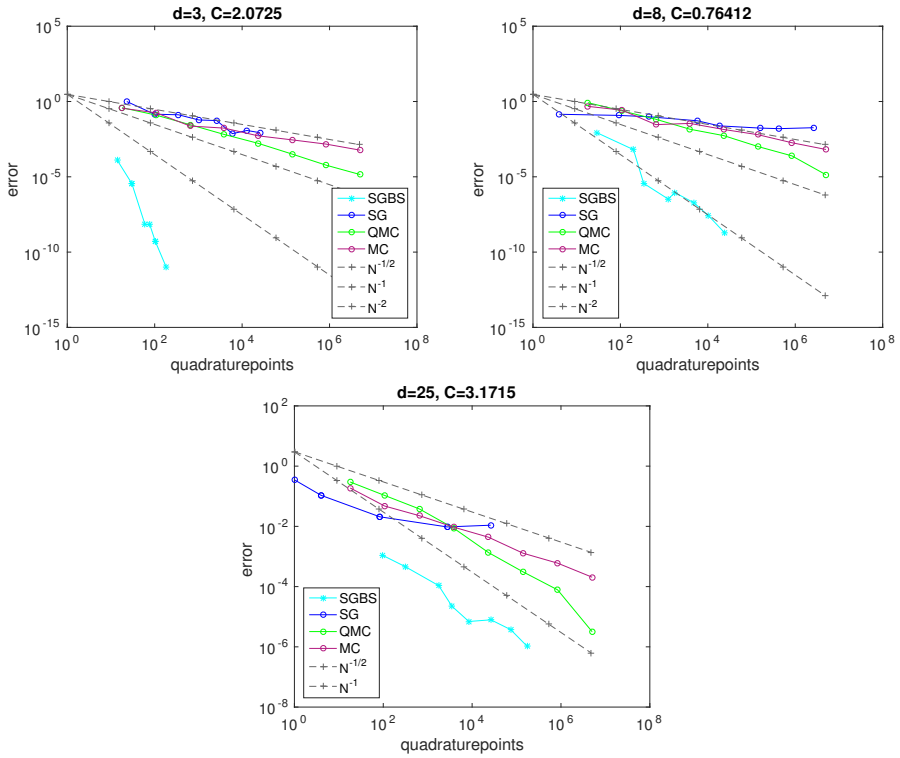

FIGURE 3. Errors for $d = 3$, $d = 8$ and $d = 25$ with volatilities selected randomly from the interval $[0.3, 0.4]$.

sparse grid construction. Therefore, we depict in Table 1 computational times in seconds and errors for the different quadrature methods at a comparable number of quadrature points for each dimension. As we can deduce from these times, there is indeed a huge overhead for the SGBS. In dimension $d = 25$, for example, the computation of the adaptive sparse grid method with around 25% more quadrature points requires around 23 times the computation time in comparison to QMC. Nevertheless, the error of the SGBS is aroud a factor 600 smaller in comparison to QMC. Note that all the computations are done in

	SGBS			QMC			MC		
	time	error	points	time	error	points	time	error	points
$d = 3$	0.0057	4.9 e-10	104	0.0016	1.25 e-1	108	0.0013	1.77 e-1	108
$d = 8$	0.3675	1.81 e-9	24622	0.0161	5.39 e-3	23328	0.0135	1.38 e-2	23328
$d = 25$	5.4283	1.04 e-6	174098	0.2409	6.18 e-4	139968	0.2188	1.29 e-3	139968

TABLE 1. Computation times for the different quadrature methods

MATLAB and that the evaluation of the integrand is completely vectorized in case of the (Q)MC. This is, of course, not possible for the adaptive sparse grid quadrature, since we adaptively add indices, which correspond to difference quadrature rules with a relatively low number of quadrature points, to the index set. Although, the evaluation of the integrand in each difference quadrature rule is vectorized, we need to do this several times during the

algorithm. Hence, a MATLAB implementation is not the most efficient one for adaptive sparse grid quadratures or adaptive methods in general and the overhead could be reduced drastically with an efficient implementation in e.g. C.

The results for $d = 3$, $d = 5$ and $d = 25$ are depicted in Figure 3. As expected, the Monte-Carlo quadrature converges in each dimension algebraically with a rate $1/2$ against the reference solution, while the rate of the quasi-Monte Carlo quadrature is close to 1. The convergence of the SG is comparable to that of the MC for $d = 3$ and becomes worse for $d = 8$ and $d = 25$. Hence, it is not very suitable to tackle the original problem (4). In contrast to that, the SGBS outperforms all the other considered methods, especially for $d = 3$ and $d = 8$, in both convergence rate and constant. For $d = 3$, the rate is rather exponential than algebraic and the observed algebraic rate for $d = 8$ is 2. In $d = 25$ dimensions, the rate deteriorates to 1 but the constant is still around a factor 100 less compared to the constant of the QMC.

Summarizing, we find that the adaptive sparse grid quadrature applied to (16) yields very good results to approximate the value of a basket option. In particular, it significantly improves the performance of the adaptive sparse grid quadrature applied to (4). This is due to the fact that the integrand in (16) is smooth while the integrand in (4) is not even differentiable.

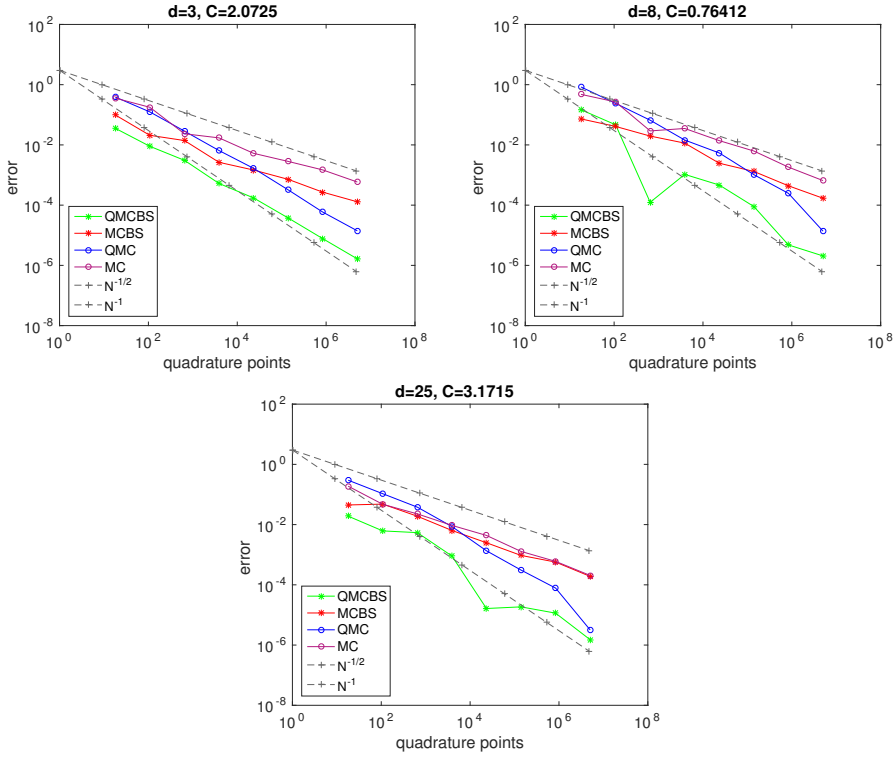

FIGURE 4. Smoothing effect for the (quasi)-Monte Carlo quadrature for $d = 3$, $d = 8$ and $d = 25$ with volatilities selected randomly from the interval $[0.3, 0.4]$.

4.2. **Smoothing effect for Monte Carlo and quasi-Monte Carlo quadrature.** In this subsection, we examine the smoothing effect on the (quasi-) Monte Carlo quadrature. To that end, we apply the (quasi-) Monte Carlo quadrature with the same number of quadrature points as before, i.e. $3 \cdot 6^q$ for $q = 1, \dots, 8$, to approximate the integral in (16) and compare the results with those of the (quasi-) Monte Carlo quadrature applied to (4). For the Monte Carlo quadrature, we expect that the smoothing effect is not as strong as for the sparse grid quadrature. Nevertheless, the convergence constant might be improved since we determined a conditional expectation to deduce (16) from (4) which should decrease the variance of the integrand. Figure 4 corroborrates that the smoothing has the expected effect for the Monte Carlo quadrature, but the effect seems to diminish in higher dimensions. In case of the quasi-Monte Carlo quadrature, the smoothing does not effect the convergence rate but improves the convergence constant as well. Moreover, the effect is even stronger as for the Monte Carlo quadrature. The convergence constant of the quasi-Monte Carlo quadrature relies on the variation of the integrand and, hence, we suspect a larger decrease in the variation of the integrand than in the variance. An explanation is that the variation of a function can be calculated from the first mixed derivatives. Thus, the variation strongly depends on the smoothness of the integrand, in particular this dependence is stronger than for the variance of the integrand.

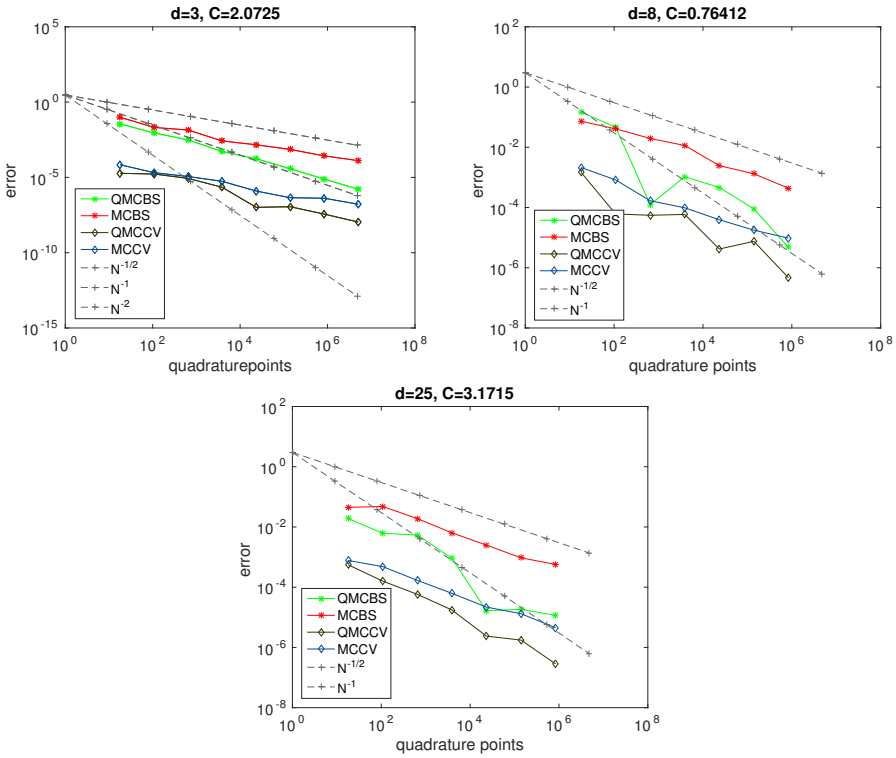

FIGURE 5. Acceleration of the (quasi-) Monte Carlo quadrature with a sparse grid control variate for $d = 3$, $d = 8$ and $d = 25$ with volatilities selected randomly from the interval $[0.3, 0.4]$.

4.3. **Acceleration by using a sparse grid interpolant as a control variate.** Another option to exploit the smoothness of the integrand is to combine a (quasi-) Monte Carlo quadrature with a sparse grid approximation. To that end, we construct a sparse grid interpolant on the integrand in (16), i.e. we use sparse grid quadrature nodes as interpolation points, and employ this interpolant as a control variate. To explain the concept of a control variate, let us consider the integration problem of a function $f\colon \mathbb{R}^d \to \mathbb{R}$ and an approximation $g\colon \mathbb{R}^d \to \mathbb{R}$ on f. We assume that it is easy to calculate $E(g) := \int_{\mathbb{R}^d} g(x)\,\mathrm{d}x$. Then, we rewrite the integral as

$$\int_{\mathbb{R}^d} f(x)\,\mathrm{d}x = \int_{\mathbb{R}^d} f(x) - g(x)\,\mathrm{d}x + E(g). \tag{17}$$

Instead of using a (quasi-) Monte Carlo estimate of the integral on the left-hand side of 17, we estimate the integral on the right-hand side. Then, a function $g\colon \mathbb{R}^d \to \mathbb{R}$ serves as a control variate, see e.g. [18] for a more detailed description. Of course, the quality of the control variate depends on how much the variance or the variation of $f - g$ is reduced compared with the variance or variation of f. Hence, it is closely connected to the approximation quality of g on f.

In our examples, we use as interpolation points classical sparse grid quadrature points. That means that we choose the index set a priori as in (12) on level $q = 2$. Moreover, as one-dimensional quadrature points, we use Gauß-Hermite points with $N_0 = 1$, $N_1 = 3$ and $N_2 = 5$, respectively.

Remark 4.3. The evaluation of this sparse grid interpolant at the (quasi-) Monte Carlo quadrature points becomes quite costly, especially in high dimensions. Most likely, more efficient control variates could be used, e.g. by including only the 5 most important dimensions in the sparse grid interpolant. Nevertheless, the aim here is to demonstrate that it is possible, due to the smoothing, to significantly improve the convergence behaviour of the (quasi-) Monte Carlo quadrature by a sparse grid control variate on a relatively low level but we do not incorporate an efficiency analysis in terms of computational times here.

The results of employing such a function as a control variate to improve the convergence of the (quasi-) Monte Carlo quadrature are visualized in Figure 5. The error reduction is quite impressive for both methods. Especially the error of the Monte-Carlo quadrature is reduced by around a factor 10^3 in $d = 3$ dimension and still by a factor 10^2 in $d = 8$ and $d = 25$ dimensions while the convergence rate is preserved. In case of the quasi-Monte Carlo quadrature, the constant is reduced by a similar factor as in the Monte Carlo case. Although, the convergence rate seems to be slightly worse in comparison with the quasi-Monte Carlo quadrature without control variate, the quasi-Monte Carlo quadrature with a sparse grid control variate achieves the best error behaviour of the four considered methods in Figure 5.

5. Numerical example 2: Multivariate Variance Gamma setting

In our second numerical example, we consider the pricing of a basket option in a multivariate variance gamma model as introduced in [28]. Therefore, we recall that the multivariate extension of the univariate asset price process (6) is described as follows, cf. [27] and Section 1 above,

$$S_t^i = S_0^i \exp\left((r + \omega_i)t + \theta_i \gamma_t + \sigma_i W_i(\gamma_t)\right) \tag{18}$$

with

$$\omega_i = \frac{1}{\nu}\log\left(1 - \frac{1}{2}\sigma_i^2\nu - \theta_i\nu\right).$$

We incorporate here also the deterministic interest rate r in order to compare our results with those from [27]. The correlated d-dimensional Brownian motion W in (18) is as in (2) given by its correlation matrix $\rho = \left(\rho_{i,j}\right)_{i,j=1}^{d}$ and its volatility vector $\sigma = [\sigma_1, \dots, \sigma_d]^\top$. The Gamma process γ_t is independent from W and described by the parameter ν via its density function

$$f_{\gamma_t}(y) = \frac{y^{1/\nu - 1}}{\nu^{t/\nu}\Gamma(t/\nu)} e^{-y/\nu}.$$

The calculation of a European basket call option at time T under the variance gamma model leads then to

$$C_{\mathcal{B}} \coloneqq \int_0^\infty e^{-rT} E\left[\left(\sum_{i=1}^{d} c_i S_T^i - K\right)^+ \middle| \gamma_T = y\right] f_{\gamma_T}(y)\,\mathrm{d}y. \tag{19}$$

Herein, the integrand is for every fixed $y \geq 0$ just the value of a basket call option according to (3). Let us define

$$\begin{aligned} w_i &= c_i S_0^i e^{(r+\omega_i)T}, \quad i = 1, \dots, d, \\ \Sigma_{i,j} &= \sigma_i \sigma_j \rho_{i,j} T, \quad i, j = 1, \dots, d. \end{aligned} \tag{20}$$

Then, we can as in (4) rewrite the integrand in terms of a d-dimensional Gaussian vector $X^y = (X_1^y, \dots, X_d^y) \sim \mathcal{N}(0, y \cdot \Sigma)$ to

$$E\left[\left(\sum_{i=1}^{d} c_i S_T^i - K\right)^+ \middle| \gamma_T = y\right] = E\left[\left(\sum_{i=1}^{d} e^{\theta_i y} w_i e^{X_i} - K\right)^+ \middle| \gamma_T = y\right].$$

Hence, we can apply the technique from Section 3 to equation (19). Therefore, we recall the decomposition of the matrix $\Sigma = VDV^\top$ according to Lemma 3.1. The first row of the matrix V is the vector $v = [1, \dots, 1]^\top$ and we denote the entries of the diagonal matrix by $D = \operatorname{diag}(\lambda_1^2, \dots, \lambda_d^2)$. Continuing in the same fashion as in Section 3, we end up with the equivalent integration problem, cf. (16),

$$\begin{aligned} C_{\mathcal{B}} &= \int_0^\infty e^{-rT} E\left[C_{BS}\left(h_y\left(\sqrt{y\overline{D}}Z\right) e^{y\lambda_1^2/2}, K, \sqrt{y}\lambda_1\right)\right] f_{\gamma_T}(y)\,\mathrm{d}y, \\ Z &\sim \mathcal{N}(0, I_{d-1}), \quad \sqrt{\overline{D}} = \operatorname{diag}(\lambda_2, \dots, \lambda_d). \end{aligned} \tag{21}$$

Herein, the function h_y is given similar as in (15) by

$$h_y(z_2, \dots, z_d) \coloneqq \sum_{i=1}^{d} e^{\theta_i y} w_i \exp\left(\sum_{j=2}^{d} V_{i,j} z_j\right), \quad \bar{z} = (z_2, \dots, z_d) \in \mathbb{R}^{d-1}.$$

Note that the integrand in (21) is very easy to calculate with respect to y since we only need to incorporate the factor $e^{\theta_i y}$ in front of each weight w_i and scale the matrix D by y. Thus, the decomposition of the correlation matrix in view of Lemma 3.1 has only to be computed once although the correlation matrix of the Gaussian vector X^y depends on the parameter y.

In Figure 6, we present two examples for basket option pricing under the variance gamma model. The first picture on the right-hand side depicts the error of the calculation of an at the money basket call, cf. (19). We choose the parameters $r = 0$ and $\nu = 0.3$

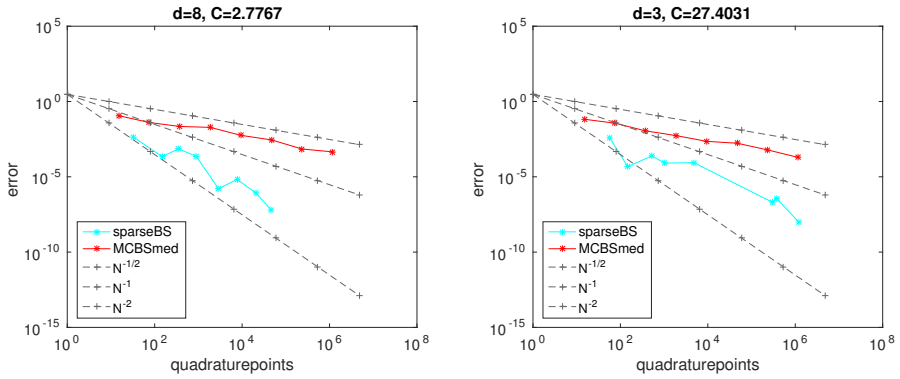

FIGURE 6. Errors of an at the money basket call under a variance gamma model with parameters $\nu = 0.3$ and $\theta_i \in [-0.1, 0.05]$ for $d = 8$ assets on the right and for an example from [27] with $d = 3$ assets on the left.

in (18) deterministically and randomly select $\theta_i \in [-0.1, 0.05]$. Moreover, the correlation matrix ρ, the volatilities σ_i and the initial values S_0^i are constructed as in Section 4. We compare the convergence of the Monte-Carlo quadrature and the adaptive sparse grid quadrature for the d-dimensional integral in (21). Note that the integration domain and the density function in (21) are given by

$$\Gamma = [0, \infty] \times \mathbb{R}^{d-1}, \qquad p(y, z_2, \dots, z_d) = f_{\gamma_T}(y) \cdot \frac{1}{(2\pi)^{d/2}} \exp\left(\frac{1}{2} \sum_{i=2}^{d} z_i^2\right).$$

Hence, we use as samples for the Monte-Carlo quadrature d-dimensional random vectors where the first component is distributed with respect to f_{γ_T} and independent to the remaining $d - 1$ variables which are normally distributed and independent as well. In case of the adaptive sparse grid quadrature, we apply tensor products of difference quadratures rules, cf. (11), where we use as quadrature sequence in the first variable differences of generalized Gauss-Laguerre quadrature rules. In the remaining variables, we set the univariate quadratures as in Section 4. Afterwards, we select the indices which are included in the sparse grid adaptively as described in Section 2.2. As expected, the Monte Carlo method converges exactly with a rate $N^{-1/2}$. Moreover, the result demonstrates that the adaptive quadrature outperforms the Monte Carlo method even in this variance gamma example with an observed rate of nearly N^{-2}.

The second numerical example is taken from the recent work [27] and stems originally from a parameter fitting of the variance gamma model in [29]. It describes a 3-dimensional model as in (18) where $\theta = [-0.1368, -0.056, -0.1984]^\top$, $\sigma = [0.1099, 0.1677, 0.0365]^\top$ and $S_0 = [100, 200, 300]^\top$. Additionally, the weight vector $c = [1/3, 1/6, 1/9]^\top$ and the correlation matrix

$$\rho = \begin{pmatrix} 1 & 0.6 & 0.9 \\ 0.6 & 1 & 0.8 \\ 0.9 & 0.8 & 1 \end{pmatrix}$$

are used. In [27], several different settings for the parameter ν and the strike price K are considered. We restrict ourselves to the setting $\nu = 0.5$ and $K = 75$ which corresponds to an "in the money" basket. On the left-hand side of picture 6 the convergence results for the Monte Carlo and the adaptive approach are shown. We observe that the Monte Carlo

quadrature converges as before. Although the convergence of the adaptive sparse grid quadrature is still better than that of the Monte Carlo method, an exponential rate as could be expected for such a low dimensional example cannot be obtained. This deterioration in the convergence rate does not depend on the variance gamma setting but, according to Remark 3.6, there is a connection of the smoothing to the entries of the diagonal matrix D from Lemma 3.1. For the considered example, the matrix D has the entries $\lambda_1^2 = 0.00023$, $\lambda_2^2 = 0.03432$ and $\lambda_3^2 = 0.00652$. In particular, the small value of λ_1^2 explains the relatively low smoothing effect. In view of Remark 3.5, the vector $\mathbf{1}$ in Lemma 3.1 can be replaced by any other vector $0 \neq v \in \{0,1\}^d$ in order to obtain a closed-form expression in Lemma 3.3. Therefore, we investigated also the convergence behaviour when we used a vector $v \neq \mathbf{1}$.

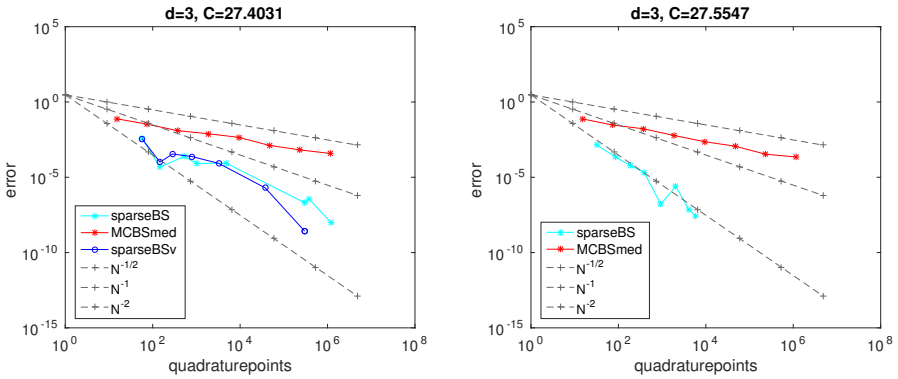

FIGURE 7. Errors for the example from [27] with $d = 3$ assets. On the left-hand side, we included the convergence when the vector $\mathbf{1}$ is replaced by $v = [1,1,0]^\top$. On the right-hand side, we used the modified volatility $\sigma_3 = 0.1365$.

On the left-hand side of Figure 7, the result is visualized for the vector $v = [1,1,0]^\top$, which is the best possible choice for this example. We observed an improved convergence with this choice of v, which comes in concordance with an increase in the size of $\lambda_{1,v}^2 = 0.00109$, i.e. ($\lambda_{1,v}^2$ is five times as high as λ_1^2. Nevertheless, $\lambda_{1,v}^2$ is still quite small compared with $\lambda_{2,v}^2 = 0.03294$ and, hence, the improvement in the convergence is not that extraordinary. This leads to the supposition that the considered example is not that well suited for our proposed method. In particular, the low value of $\sigma_3 = 0.0365$ compared with the other volatilities seems to have a negative effect on the smoothing. Hence, we tested this example also for the modified volatility $\sigma_3 = 0.1365$. The results for this case are depicted on the right-hand side of figure 7 and, indeed, we observe a drastically improved convergence. Furthermore, the entries of D are given by $\lambda_1^2 = 0.01034$, $\lambda_2^2 = 0.02255$ and $\lambda_3^2 = 0.00526$ which demonstrates the influence of the differences in the volatilities on the size of λ_1^2 and, thus, on the smoothing.

CONCLUSIONS

In the context of basket options, we show that the inherent smoothing property of a Gaussian component of the underlying can be used to mollify the integrand (payoff function) without introducing an additional bias. Having obtained a smooth integrand, we can now directly apply (adaptive) sparse grid methods. We observe that these methods are

highly efficient in low and moderately high dimensions. For instance, the error can be improved by two orders of magnitude in dimension 25 compared to (Q)MC methods. In dimension 3, we even obtain exponential convergence. We have also discussed improvements for MC and QMC methods by introducing the smoothed payoff. In the Monte Carlo case, we do not observe a significant improvement of the computational error, as the variance reduction seems rather negligible. For QMC methods—Sobol numbers, to be more precise—we do see considerable improvements in the constant. As expected, the rate stays the same.

We note that the method employed in this work is not restricted to basket options in a multivariate Black Scholes or Variance Gamma setting, but can be generalized considerably. For instance, each step of an Euler discretization of an SDE corresponds to a Gaussian mixture model. Hence, the conditional expectation of the final integrand given all the Brownian increments but the last one is of the form of a Gaussian integral of the payoff function w.r.t. to a normal distribution with possibly complicated mean vector and covariance matrix. If this integral can be computed explicitly, then we can directly obtain mollification of the payoff without introducing a bias.

Even if the integral cannot be computed in closed form, there may be use cases for employing numerical integration. For instance, in the basket option case, a fast and highly accurate numerical integration of the one-dimensional log-normal integral, coupled with regression/interpolation (to avoid re-computation of the one-dimensional integral for each new (sparse) gridpoint) could well turn out to be more efficient than a numerical integration technique applied to the full problem.

Finally, note that there are also clear limitations of the technique. For instance, consider a variety of the basket option studied in this work, namely a best of call option. Here, the payoff is given by

$$\left(\max_{i=1,\dots d} S_T^i - K\right)^+$$

for log-normally distributed, correlated variables S_T^i (in the Black-Scholes setting). Clearly, we can use Lemma 3.1 in order to construct a common normal factor Y and other factors $Y_1, \dots, Y_d$ (all jointly normal, Y independent of the rest), such that $S_T^i = e^Y e^{Y_i}$. Therefore, we obtain for the price of the best of call option

$$E\left[\left(\max_{i=1,\dots d} S_T^i - K\right)^+\right] = E\left[\left(e^Y \max_{i=1,\dots d} e^{Y_i} - K\right)^+\right].$$

Taking the conditional expectation, we obtain the Black-Scholes formula applied at $\max_{i=1,\dots d} e^{Y_i}$, which is still a non-smooth payoff. The mollification can only remove one source of irregularity in this case, not all of them. Indeed, as currently presented in this work, the conditional expectation step is most effective when the discontinuity surface of the option's payoff has codimension one.

Acknowledgements. R. Tempone is a member of the KAUST Strategic Research Initiative, Center for Uncertainty Quantification in Computational Sciences and Engineering. C. Bayer and M. Siebenmorgen received support for research visits related to this work from R. Tempone's KAUST baseline funds.

References

[1] Nico Achtsis, Ronald Cools, and Dirk Nuyens. *Conditional Sampling for Barrier Option Pricing Under the Heston Model*, pages 253–269. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[2] Nico Achtsis, Ronald Cools, and Dirk Nuyens. Conditional sampling for barrier option pricing under the LT method. *SIAM J. Financial Math.*, 4(1):327–352, 2013.

[3] W. Alt. *Nichtlineare Optimierung: Eine Einführung in Theorie, Verfahren und Anwendungen*. Vieweg+Teubner Verlag, 2002.
[4] Christian Bayer, Peter K. Friz, and Peter Laurence. On the probability density function of baskets. In *Large deviations and asymptotic methods in finance*, volume 110 of *Springer Proc. Math. Stat.*, pages 449–472. Springer, Cham, 2015.
[5] Christian Bayer and Peter Laurence. Asymptotics beats Monte Carlo: the case of correlated local vol baskets. *Comm. Pure Appl. Math.*, 67(10):1618–1657, 2014.
[6] Christian Bayer and Peter Laurence. Small-time asymptotics for the at-the-money implied volatility in a multi-dimensional local volatility model. In *Large deviations and asymptotic methods in finance*, volume 110 of *Springer Proc. Math. Stat.*, pages 213–237. Springer, Cham, 2015.
[7] H.-J. Bungartz and M. Griebel. Sparse grids. *Acta Numerica*, 13:147–269, 2004.
[8] R. Caflisch. Monte Carlo and quasi-Monte Carlo methods. *Acta Numerica*, 7:1–49, 1998.
[9] Ronald Cools. An encyclopaedia of cubature formulas. *Journal of Complexity*, 19(3):445 – 453, 2003. Oberwolfach Special Issue.
[10] P. Doust. Modelling discrete probabilities. Technical report, Quantitative Analysis Group, RBS, 2007.
[11] Daniel Dufresne. The log-normal approximation in financial and other computations. *Adv. in Appl. Probab.*, 36(3):747–773, 2004.
[12] Eric Fournié, Jean-Michel Lasry, Jérôme Lebuchoux, Pierre-Louis Lions, and Nizar Touzi. Applications of Malliavin calculus to Monte Carlo methods in finance. *Finance Stoch.*, 3(4):391–412, 1999.
[13] Jim Gatheral. *The Volatility Surface: A Practitioner's Guide*. Wiley, 2006.
[14] John H. Welsch Gene H. Golub. Calculation of gauss quadrature rules. *Mathematics of Computation*, 23(106):221–s10, 1969.
[15] Alan Genz and B.D. Keister. Fully symmetric interpolatory rules for multiple integrals over infinite regions with gaussian weight. *Journal of Computational and Applied Mathematics*, 71(2):299 – 309, 1996.
[16] T. Gerstner and M. Griebel. Dimension–adaptive tensor–product quadrature. *Computing*, 71(1):65–87, 2003.
[17] Thomas Gerstner. Sparse grid quadrature methods for computational finance. *Habilitation, University of Bonn*, 77, 2007.
[18] P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Applications of mathematics : stochastic modelling and applied probability. Springer, 2004.
[19] M. Griebel, F. Y. Kuo, and I. H. Sloan. The smoothing effect of integration in $\mathbb{R}^d$ and the ANOVA decomposition. *Math. Comp.*, 82:383–400, 2013. Also available as INS preprint No. 1007, 2010.
[20] M. Griebel, F. Y. Kuo, and I. H. Sloan. The ANOVA decomposition of a non-smooth function of infinitely many variables can have every term smooth. Submitted to Mathematics of Compuation. Also available as INS preprint No. 1403, 2014, 2014.
[21] M. Griebel, F. Y. Kuo, and I. H. Sloan. Note on "The smoothing effect of integration in $\mathbb{R}^d$ and the ANOVA decomposition". Submitted to Mathematics of Compuation. Also available as INS preprint No. 1513, 2015.
[22] J. M. Hammersley and D. C. Handscomb. *Monte Carlo methods*. Methuen, London, 1964.
[23] Florian Heiss and Viktor Winschel. Likelihood approximation by numerical integration on sparse grids. *Journal of Econometrics*, 144(1):62–80, 2008.
[24] Markus Holtz. *Sparse Grid Quadrature in High Dimensions with Applications in Finance and Insurance*. Springer Verlag, Berlin, Heidelberg, 2011.
[25] Martin Krekel, Johan de Kock, Ralf Korn, and Tin-Kwai Man. An analysis of some methods for pricing basket options. *Wilmott*, pages 82–89, 2004.
[26] Pierre L'Ecuyer. Quasi-Monte Carlo methods with applications in finance. *Finance and Stochastics*, 13(3):307–349, 2009.
[27] Danil Linders and Ben Stassen. The multivariate variance gamma model: basket option pricing and calibration. *Quantitative Finance*, 0(0):1–18, 0.
[28] Elisa Luciano and Wim Schoutens. A multivariate jump-driven financial asset model. *Quantitative Finance*, 6(5):385–402, 2006.
[29] Dilip B. Madan, Peter P. Carr, and Eric C. Chang. The variance gamma process and option pricing. *Eur. Finance Rev.*, 2(1):79–105, 1998.
[30] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
[31] Marc Romano and Nizar Touzi. Contingent claims and market completeness in a stochastic volatility model. *Math. Finance*, 7(4):399–412, 1997.
[32] S. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Doklady Akademii Nauk SSSR*, 4:240–243, 1963.

[33] I.M Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86 – 112, 1967.

WEIERSTRASS INSTITUTE, MOHRENSTRASSE 39, 10117 BERLIN, GERMANY
E-mail address: christian.bayer@wias-berlin.de

INSTITUTE FOR NUMERICAL SIMULATION, UNIVERSITY OF BONN, WEGELERSTR. 6, 53115 BONN, GERMANY
E-mail address: siebenmo@ins.uni-bonn.de

CEMSE, KING ABDULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY (KAUST), THUWAL 23955-6900, SAUDI ARABIA
E-mail address: raul.tempone@kaust.edu.sa