SEMI-CLOSED FORM CUBATURE AND APPLICATIONS TO FINANCIAL DIFFUSION MODELS

CHRISTIAN BAYER, PETER FRIZ, RONNIE LOEFFEN

Abstract. Cubature methods, a powerful alternative to Monte Carlo due to Kusuoka [Adv. Math. Econ. 6, 69–83, 2004] and Lyons–Victoir [Proc. R. Soc. Lond. Ser. A 460, 169–198, 2004], involve the solution to numerous auxiliary ordinary differential equations. With focus on the Ninomiya-Victoir algorithm [Appl. Math. Fin. 15, 107–121, 2008], which corresponds to a concrete level 5 cubature method, we study some parametric diffusion models motivated from financial applications, and exhibit structural conditions under which all involved ODEs can be solved explicitly and efficiently. We then enlarge the class of models for which this technique applies, by introducing a (model-dependent) variation of the Ninomiya-Victoir method. Our method remains easy to implement; numerical examples illustrate the savings in computation time.

1. Introduction

We deal with the common problem in quantitative finance to compute, as fast and accurately as possible,

$$\mathbb{E}\left[f\left(X_T\right)\right]. \tag{1}$$

Here, $f : \mathbb{R}^N \to \mathbb{R}$ denotes a typical payoff function and $(X_t)_{0\le t\le T}$ is an N-dimensional diffusion process, given in terms of a stochastic differential equation (SDE) in Stratonovich form

$$\begin{pmatrix} X_1(t,x) \\ \vdots \\ X_N(t,x) \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix} + \begin{pmatrix} \int_0^t V_0^1(X(s,x))\mathrm{d}s \\ \vdots \\ \int_0^t V_0^N(X(s,x))\mathrm{d}s \end{pmatrix} + \begin{pmatrix} \sum_{j=1}^d \int_0^t V_j^1(X(s,x)) \circ \mathrm{d}B_s^j \\ \vdots \\ \sum_{j=1}^d \int_0^t V_j^N(X(s,x)) \circ \mathrm{d}B_s^j \end{pmatrix}.$$

where $x = (x_1, \dots, x_N) \in \mathbb{R}^N$ and $B = (B^1, \dots, B^d)$ is a d-dimensional standard Brownian motion. Whenever convenient, we shall use the compact notation

$$X(t,x) = x + \int_0^t V_0(X(s,x))\mathrm{d}s + \sum_{j=1}^d \int_0^t V_j(X(s,x)) \circ \mathrm{d}B_s^j, \tag{2}$$

or, in Itô form,

$$X(t,x) = x + \int_0^t \widetilde{V}_0(X(s,x))\mathrm{d}s + \sum_{j=1}^d \int_0^t V_j(X(s,x))\mathrm{d}B_s^j,$$

where $\tilde{V}_0^i(x) = V_0^i(x) + \frac{1}{2}\sum_{j=1}^d \sum_{k=1}^N V_j^k \partial_k V_j^i(x)$.

Key words and phrases. Ninomiya–Victoir method, cubature method, Monte Carlo simulation.

As is common in the analysis of higher-order, weak approximation methods for such SDEs (cf. the classics Kloeden and Platen [10], Glasserman [7] as well as Kusuoka [11], Lyons and Victoir [15] and Ninomiya and Victoir [18] for cubature type methods) we shall assume that the payoff function f and all vector fields $V_0, V_1, \dots, V_d$ are smooth, with bounded derivatives of any order. The standing remark in this subject, *implicit in all of the aforementioned references*, is that any scheme obtained from such an analysis can and will be applied to typical financial diffusion models (such as Heston, SABR and their – possibly higher-dimensional – generalizations) even if they do not satisfy the technical assumptions initially used in the analysis; numerical experiments (which are necessary for every numerical scheme in any case!) serve as a posteriori justification.[1]

We do not wish to impose any special structure on (2); in particular the vector fields are not supposed to commute (cf. Kloeden and Platen [10, page 348] for the advantages in such a case in the particular case of the Milstein scheme), no affine structure (as in the Heston model) is assumed, nor do we want to rely on heat-kernel based expansions of (1) (such as the SABR formula). In this generality, one has essentially two approaches. The *PDE method*, based on the Feynman-Kac formula, consists in solving the Cauchy problem for the partial differential equation

$$\partial_t u\left(t,x\right) + Lu(t,x) = 0, \quad u\left(T,x\right) = f(x)$$

where the 2nd order differential operator L is given in Hörmander form $L = V_0 + \frac{1}{2}\sum_{i=1}^{d} V_i^2$ where vector-fields are identified with first order differential operators. As is well known, that PDE approach is prohibitively slow in higher dimension; there are also stability issues when L is not elliptic. The other approach is *the probabilistic "simulation" method* which requires two steps. In **step 1** one discretizes $X\left(t,x\right)$ in order to obtain an approximation $\overline{X}^{K}\left(t,x\right)$; typically, K corresponds to the number of partitions of $[0,T]$; examples include the Euler-Maruyama (EM) scheme

$$\begin{aligned}
\overline{X}^{(EM),K}\left(0,x\right) &= x \in \mathbb{R}^N \\
\overline{X}^{(EM),K}\left(\frac{k+1}{K},x\right) &= \overline{X}^{(EM),K}\left(\frac{k}{K},x\right) + \widetilde{V}_0(\overline{X}^{(EM),K}\left(\frac{k}{K},x\right))\times\frac{T}{K} \\
&\quad + \sqrt{\frac{T}{K}}\sum_{j=1}^{d} V_j\left(\overline{X}^{(EM),K}\left(\frac{k}{K},x\right)\right) Z_{k+1}^{j},
\end{aligned}$$

where $\left(Z_k^j\right)$ is a family of independent $\mathcal{N}(0,1)$[2] random variables, as well as higher order (Milstein, Kusuoka, Ninomiya–Victoir, ...) schemes which we do not wish to detail at this moment. The *discretization error* is given by

$$\left|\mathbb{E}\left[f\left(X\left(T,x\right)\right)\right] - \mathbb{E}\left[f\left(\overline{X}^{K}\left(T,x\right)\right)\right]\right| = \begin{cases} \mathcal{O}\left(T/K\right) & \text{for Euler-Maruyama} \\ \mathcal{O}\left((T/K)^2\right) & \text{for Ninomiya-Victoir} \\ \dots & \end{cases}$$

[1] It is possible to analyze mollified/truncated versions of CIR, Heston, SABR, ... and thus provide further mathematical justification. For instance, it was only recently shown in full rigor that the classical Euler-Maruyama scheme applied to the Heston model converges; see e.g. Mao and Higham [8]. Let us also mention the work of Alfonsi [1] in this context. Such considerations are not the purpose of the present paper.

[2] Throughout the paper $\mathcal{N}(\mu,\sigma^2)$ denotes the normal distribution with mean μ and variance σ^2.

In **step 2** one has to integrate $f\left(\overline{X}^K(T,x)\right)$ over some domain of dimension $D = D(K)$ such as[3]

$$\mathbb{E}\left[f\left(\overline{X}^K(T,x)\right)\right] = \int_{[0,1)^{D(K)}} F\left(y_1,\dots,y_{D(K)}\right) dy_1 \dots dy_{D(K)}.$$

Here, F denotes the dependence of $f\left(\overline{X}^K(T,x)\right)$ on uniform random variables, i.e., $F\left(U_1,\dots,U_{D(K)}\right) = f\left(\overline{X}^K(T,x)\right)$ for a collection $\left(U_1,\dots,U_{D(K)}\right)$ of independent random variables uniformly distributed on the unit interval. The right-hand-side is approximated by Monte Carlo (MC) or Quasi Monte Carlo (QMC), essentially obtained by averaging M samples of $F\left(y_1,\dots,y_{D(K)}\right)$. These samples are random if created by Monte Carlo (MC) and deterministic if obtained by Quasi Monte Carlo (QMC). In either case, we have an *integration error* of the form

$$\left|\mathrm{MC}\left(f\left(\overline{X}^K(T,x)\right),M\right)(\omega) - \mathbb{E}\left[f\left(\overline{X}^K(T,x)\right)\right]\right|,$$
$$\left|\mathrm{QMC}\left(f\left(\overline{X}^K(T,x)\right),M\right) - \mathbb{E}\left[f\left(\overline{X}^K(T,x)\right)\right]\right|.$$

The central limit theorem roughly implies that MC-integration error is $O\left(1/\sqrt{M}\right)$. More precisely, we have in the sense of an asymptotic equality in law,

$$\mathrm{MC}\left(f\left(\overline{X}^K(t,x)\right),M\right) \approx \mathcal{N}\left(\mathbb{E}\left[f\left(\overline{X}^K(t,x)\right)\right], \mathbb{V}\left[f\left(\overline{X}^K(t,x)\right)\right]/M\right)$$

so that, using $\mathbb{V}\left[f\left(\overline{X}^K(t,x)\right)\right] \approx \mathbb{V}\left[f\left(X(t,x)\right)\right]$ we see that the number of sample points M needed to attain a given accuracy (i.e. a certain ε bound for the MC-integration error) is roughly independent of K and the discretization algorithm. The situation is somewhat different for the QMC-integration error. It is known that there exists sequences ("sample points") such that there exists $C = C(f, D(K))$ such that for all M one has

$$\left|\mathrm{QMC}\left(f\left(\overline{X}^K(T,x)\right),M\right)(\omega) - \mathbb{E}\left[f\left(\overline{X}^K(T,x)\right)\right]\right| \le C\frac{(\log M)^{D(K)}}{M}.$$

In contrast to the MC case, the number of sample points M needed by QMC to attain a given accuracy depends heavily on the dimension of integration $D(K)$ and, possibly, on the smoothness of $f\left(\overline{X}^K\right)$ as a function in the points $y_1,\dots,y_{D(K)}$. Moreover, the above error estimate is known to grossly overestimate the true error in many cases.

1.1. **Cubature on Wiener Space.** Let us briefly put the (Kusuoka–Lyons–Victoir) cubature method in this context. For simplicity of notation only, we consider the case $V_0 = 0$ here. A *cubature formula on Wiener space* is a random variable W taking values in the space $C^{1\text{-var}}([0,1],\mathbb{R}^d)$ of continuous paths of bounded variation with values in $\mathbb{R}^d$ such that we have

$$\mathbb{E}\left[\int_{0\le t_1\le\dots\le t_j\le 1} \circ \mathrm{d}B^{i_1}_{t_1}\cdots\circ\mathrm{d}B^{i_j}_{t_j}\right] = \mathbb{E}\left[\int_{0\le t_1\le\dots\le t_j\le 1} \mathrm{d}W^{i_1}_{t_1}\cdots\mathrm{d}W^{i_j}_{t_j}\right]. \tag{3}$$

[3]The dimension $D(K)$ will depend on the method (for instance $D(K) = K\times d$ for the Euler-Maruyama scheme, $D(K) = K\times(d+1)$ for the Ninomiya–Victoir scheme).

for all multi-indices $I = (i_1, \dots, i_j) \in \{1, \dots, d\}^j$ with all $1 \leq j \leq m$, where m is a fixed positive integer, the *order* of the cubature formula. Moreover, we note that since the paths of the process W are of bounded variation, the integrals on the right hand side of (3) are then understood as classical Riemann-Stieltjes integrals. In applications, the reference interval $[0, 1]$ in (3) is typically replaced by some (small) interval such as $[0, T/K]$. (Due to Brownian scaling, however, the problems are equivalent; in particular, a cubature formula on $[0, t]$ is obtained by a scaled version of the cubature formula on $[0, 1]$.) In the classical paper of Lyons and Victoir [15] the authors actually insisted that the cubature formula is *discrete* meaning that for some positive integer k, the law of W can be written as

$$\sum_{i=1}^{k} \lambda_i \delta_{W_i},$$

where δ_{W_i} is the Dirac measure on Wiener space which assign unit mass to the path $W_i(.)$, zero to every other path. (Existence and explicit knowledge of cubature formulas is a non-trivial problem!) The idea is now to approximate the stochastic differential equation (2) for $X = X(t, x)$ by a family of (random) ordinary time-inhomogeneous differential equations,

$$\overline{X}(t, x; W) = x + \sum_{j=1}^{d} \int_0^t V_j(\overline{X}(s, x; W)) \frac{\mathrm{d}W^j(s)}{\mathrm{d}s} \mathrm{d}s,$$

where W now denotes a cubature formula on the interval $[0, t]$. A stochastic Taylor-expansions (e.g. chapter 18 in [3] for a discussion in the spirit of cubature) shows that $\mathbb{E}\left[f\left(\overline{X}(t, x; W)\right)\right] - \mathbb{E}\left[f\left(X(t, x)\right)\right] = O\left(t^{\frac{m+1}{2}}\right)$ as $t \to 0$. Observe that in the case of a discrete cubature formula $\mathbb{E}\left[f\left(\overline{X}(t, x; W)\right)\right]$ is computed exactly (no integration error!) by solving k ordinary differential equations. A (big) interval $[0, T]$ can be handled by dividing it into K intervals of length T/K and iterating this procedure but now exact computation of $\mathbb{E}\left[f\left(X(t, x; W)\right)\right]$ requires to solve

$$k + k^2 + \cdots + k^K = O\left(k^K\right)$$

ordinary differential equations. When k^K becomes too big one can either perform a Monte Carlo simulation ("on the cubature tree") or resort to recombination techniques (see Litterer and Lyons [16] for the present state of art). Let us note, however, that in many practical applications K remains small, which helps to explain the numerical benefits of cubature even without recombination.

1.2. **The Ninomiya–Victoir (NV) Scheme.** The Ninomiya–Victoir "splitting" scheme, introduced in [18], is given by

$$\begin{aligned}
&\overline{X}^{(NV),K}(0, x) = x \in \mathbb{R}^N, \\
&\overline{X}^{(NV),K}\left(\frac{(k+1)T}{K}, x\right) = \\
&\qquad = \begin{cases} \mathrm{e}^{\frac{T}{2K}V_0} \mathrm{e}^{Z_k^1 \sqrt{\frac{T}{K}} V_1} \cdots \mathrm{e}^{Z_k^d \sqrt{\frac{T}{K}} V_d} \mathrm{e}^{\frac{T}{2K}V_0} \overline{X}^{(NV),K}\left(\frac{kT}{K}, x\right) & \text{if } \Lambda_k = -1, \\ \mathrm{e}^{\frac{T}{2K}V_0} \mathrm{e}^{Z_k^d \sqrt{\frac{T}{K}} V_d} \cdots \mathrm{e}^{Z_k^1 \sqrt{\frac{T}{K}} V_1} \mathrm{e}^{\frac{T}{2K}V_0} \overline{X}^{(NV),K}\left(\frac{kT}{K}, x\right) & \text{if } \Lambda_k = +1. \end{cases}
\end{aligned}$$

Here $\mathrm{e}^V x \in \mathbb{R}^N$ denotes the ODE solution at unit time to $\dot{y} = V(y), y(0) = x$ and the probability space carries independent random-variables (Λ_k), with values ± 1 at

probability $1/2$, and $\mathcal{N}(0,1)$ random variables (Z_k^j). One step in the NV scheme corresponds actually to a (non-discrete) cubature formula of order $m = 5$. To see this, assume $V_0 = 0$ for (consistent) simplicity and let $(\mathfrak{b}_i)$ denote the canonical basis of $\mathbb{R}^d$. An $\mathbb{R}^d$-valued random path $W(\omega)$, continuous and of bounded variation, is then created via $\Lambda(\omega) \in \{+1, 1\}$ and d independent $\mathcal{N}(0,1)$ realizations $Z^1(\omega), \ldots, Z^d(\omega)$. If $\Lambda(\omega) = -1$ we take $W(\omega) : [0, T/K] \to \mathbb{R}^d$, started at 0 say, to move at constant speed, first an amount $Z_k^d\sqrt{T/K}$ in $\mathfrak{b}_d$-direction, ... until the final move $Z_k^1\sqrt{T/K}$ in $\mathfrak{b}_1$-direction; if $\Lambda(\omega) = +1$ the construction is similar but in reversed order. When $V_0 \neq 0$, one follows the flow of the drift vector-field for time $T/(2K)$ in the first and last step of the scheme; at all intermediate steps V_0 is followed for a time T/K; this is inspired by classical splitting methods in operator theory. In the case of two diffusion vector fields, the coin-flipping may be related to Talay's trick of, in a weak approximation context, replacing the (difficult to sample) Lévy's area by a discrete moment-matched random variable, see Kloeden and Platen [10, page 466 f.].

The NV scheme has attracted wide attention since its introduction in [18]; it is nowadays found in various sophisticated numerical packages such as Inria's software PREMIA for financial option computations.[4] A variation of the scheme designed to deal with degeneracies arising in some affine situations is discussed in [1]. Let us also mention the "NV inspired" schemes developed in [4] and [19].

1.3. **Semi-closed form cubature.** It is clear from the preceding discussion that cubature methods, and the NV scheme in particular, heavily rely on the ability to solve, fast and accurately, ordinary differential equations. The general cubature methods involves *time-inhomogeneous* ODEs; in general, there is no alternative to solve them numerically, typically with Runge-Kutta methods. (A detailed discussion on how Runge-Kutta methods are applied in this context is found in Ninomiya and Ninomiya [17].)

On the other hand, the Ninomiya-Victoir splitting scheme only involves the composition of solution flows to *time-homogeneous* ODEs. In particular, there will be "lucky" cases of models where all (or at least most) ODE flows can be solved exactly.[5] In such a case one has effectively found a level-5 cubature method which can be implemented without relying on numerical ODE solvers. In particular, one expects the cubature methods to perform especially well in such cases. As was observed in [18], see also Section 2.1, the Heston model is such a lucky case. We thus propose the following definition.

Definition 1. A diffusion model of type (2) where a cubature method can be implemented without any numerical ODE solutions is said to be accessible to *semi-closed form cubature (SCFC).*

For instance, any model of type (2) where all ODE flows $\mathrm{e}^{tV_0}, \ldots, \mathrm{e}^{tV_d}$ can be solved in closed form falls in this class. However, one soon encounters model (e.g. the popular SABR model, see Section 2.2) in which some of the vector-fields do not allow for flows in closed form. The contribution of this paper, *beyond suggesting the*

[4] As of Sep 2010, the weblink ralyx.inria.fr/2006/Raweb/mathfi/uid21.html contains some relevant information.

[5] By this we mean a closed-form solution to the ODE $\dot{y} = V(y), y(0) = x$ *which allows for fast numerical evaluation.* In particular, we are not interested in "closed-form" solution in terms of complicated and slow-to-evaluate special functions.

systematic use of financial models that are accessible to semi-closed form cubature, is that the class of such models can be significantly enlarged by working with an almost trivial modification of the NV scheme.[6] Before explaining our modification we point out that the SABR model then becomes accessible to semi-closed form cubature. Our modification is based on the trivial equivalence of (2) with

$$\begin{aligned} dX(t,x) =& \left(V_0(X(t,x)) - \sum_{j=1}^{d} \gamma_j V_j(X(t,x)) \right) \mathrm{d}t + \sum_{j=1}^{d} V_j(X(t,x)) \circ \mathrm{d}\left(B_t^j + \gamma_j t \right) \\ \equiv& V_0^{(\gamma)}(X(t,x)) + \sum_{j=1}^{d} V_j(X(t,x)) \circ \mathrm{d}\left(B_t^j + \gamma_j t \right) \end{aligned}$$

whatever the choice of drift parameters $\gamma_1, \dots, \gamma_d$. Assume that all diffusion vector-fields $(V_1, \dots, V_d)$ allow for flows in closed form, whereas e^{tV_0} is not available in closed form. The point is that, in a variety of concrete examples, one can pick drift parameters $\gamma_1, \dots, \gamma_d$ in a way that $\mathrm{e}^{tV_0^{(\gamma)}}$ can be solved in closed form after all.

Therefore, we propose the following variant of the Ninomiya-Victoir method (which shall be referred to as the "NV scheme with drift (trick)"):

$$\overline{X}^{(NVd),K}(0,x) = x \in \mathbb{R}^N,$$

(4)

$$\overline{X}^{(NVd),K}\left(\frac{(k+1)T}{K}, x \right) =$$

(5)
$$= \begin{cases} \mathrm{e}^{\frac{T}{2K}V_0^{(\gamma)}} \mathrm{e}^{Z_k^1 V_1} \cdots \mathrm{e}^{Z_k^d V_d} \mathrm{e}^{\frac{T}{2K}V_0^{(\gamma)}} \overline{X}^{(NVd),K}\left(\frac{kT}{K}, x \right), & \text{if } \Lambda_k = -1, \\ \mathrm{e}^{\frac{T}{2K}V_0^{(\gamma)}} \mathrm{e}^{Z_k^d V_d} \cdots \mathrm{e}^{Z_k^1 V_1} \mathrm{e}^{\frac{T}{2K}V_0^{(\gamma)}} \overline{X}^{(NVd),K}\left(\frac{kT}{K}, x \right), & \text{if } \Lambda_k = +1, \end{cases}$$

where $Z_k^i \sim \mathcal{N}\left(\frac{T}{K}\gamma_i, \frac{T}{K} \right)$ independent of each other.

The bulk of this paper is devoted to implement these ideas for a handful of (stochastic volatility) models encountered in the financial industry. Since high-dimensional problems are the raison d'être for probabilistic simulation methods, a detailed discussion of a higher-dimensional (SABR-type) model is included. At last, we discuss numerical results obtained with our "drift-modified" NV scheme: relative to the classical NV scheme we observe significant and consistent savings in computational time.

Note that we want to concentrate on the method itself, without further improvements like variance reduction, optimization of code and Romberg extrapolation.

Let us also remark at this stage that the splitting nature of the Ninomiya-Victoir scheme also allows for further splitting, e.g., of the drift vector-field, in addition to or instead of the above introduced "drift trick". More precisely, the drift-trick adds more flexibility to such splittings in the sense that it allows to split the drift vector field and combine parts of it with diffusion vector fields.

A further advantage of the Ninomiya-Victoir scheme and of other higher order schemes is an induced reduction of the dimension of the integration problem. Indeed, Higher order methods allow to use coarser time-grids, thereby reducing

[6]While SCFC corresponds to the "luckiest" case of avoiding numerical ODE solvers altogether, any significant reduction of numerical ODEs to be solved will be desirable. Our modification of the NV scheme can obviously be used to this purpose as well.

the dimension of the integral. This is particularly helpful in the context of QMC simulation.

Acknowledgment: Partial support of MATHEON and the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement nr. 258237 is gratefully acknowledged.

2. Application of classical NV scheme to Heston and SABR

2.1. **Heston model.** The stochastic volatility model of Heston is given by the SDE:

$$\begin{aligned}\mathrm{d}X_1(t,x) =& \mu X_1(t,x)\mathrm{d}t + \sqrt{X_2(t,x)}X_1(t,x)\mathrm{d}B_t^1\\ \mathrm{d}X_2(t,x) =& \kappa(\theta - X_2(t,x))\mathrm{d}t + \xi\sqrt{X_2(t,x)}\mathrm{d}\left(\rho B_t^1 + \sqrt{1-\rho^2}B_t^2\right),\end{aligned}$$

where $\mu \in \mathbb{R}$ is the rate of return of the asset, $\theta \geq 0$ is the long vol, $\kappa > 0$ is the mean-reversion rate, $\xi > 0$ is the vol(atility) of vol(atility) and $-1 \leq \rho \leq 1$ is the correlation parameter between the (standard) Brownian motions B_t^1 and $\left(\rho B_t^1 + \sqrt{1-\rho^2}B_t^2\right)$. For convenience we assume that $\rho \neq 0$.

The vector fields are given by

$$\widetilde{V}_0(x) = \begin{pmatrix}\mu x_1\\ \kappa(\theta - x_2)\end{pmatrix}, \quad V_1(x) = \begin{pmatrix}\sqrt{x_2}x_1\\ \xi\rho\sqrt{x_2}\end{pmatrix}, \quad V_2(x) = \begin{pmatrix}0\\ \xi\sqrt{1-\rho^2}\sqrt{x_2}\end{pmatrix}$$

and so we get

$$\begin{pmatrix}V_0^1(x)\\ V_0^2(x)\end{pmatrix} = \begin{pmatrix}\widetilde{V}_0^1(x)\\ \widetilde{V}_0^2(x)\end{pmatrix} - \begin{pmatrix}\frac{1}{2}\sum_{j=1}^2 V_j V_j^1(x)\\ \frac{1}{2}\sum_{j=1}^2 V_j V_j^2(x)\end{pmatrix} = \begin{pmatrix}[\mu - \frac{1}{4}\xi\rho]x_1 - \frac{1}{2}x_2x_1\\ \kappa(\theta - x_2) - \frac{1}{4}\xi^2\end{pmatrix}.$$

The corresponding solutions to the ODEs are (cf. Lord et al. [14, p.8-9] and their reference to [18]; see also the Appendix)

$$\begin{aligned}\mathrm{e}^{sV_0}x &= \begin{pmatrix}x_1\exp\left([\mu - \frac{1}{4}\xi\rho - \frac{1}{2}J]s + \frac{1}{2}\frac{x_2 - J}{\kappa}[\mathrm{e}^{-\kappa s} - 1]\right)\\ (x_2 - J)\mathrm{e}^{-\kappa s} + J\end{pmatrix},\\ \mathrm{e}^{sV_1}x &= \begin{pmatrix}x_1\exp\left(\frac{\left(\frac{1}{2}\xi\rho s + \sqrt{x_2}\right)_+^2 - x_2}{\xi\rho}\right)\\ \left(\frac{1}{2}\xi\rho s + \sqrt{x_2}\right)_+^2\end{pmatrix},\\ \mathrm{e}^{sV_2}x &= \begin{pmatrix}x_1\\ \left(\frac{1}{2}\xi\sqrt{1-\rho^2}s + \sqrt{x_2}\right)_+^2\end{pmatrix},\end{aligned}$$

with $J = \frac{\kappa\theta - \frac{1}{4}\xi^2}{\kappa}$. We assume (as in [14]) that $J \geq 0$; see [1] for how to proceed otherwise.

The Heston model can be rewritten in log-coordinates. Define $Y_1(t) = \log X_1(t,x)$ and $Y_2(t) = X_2(t)$. In this new coordinate chart, the vector fields are

$$\begin{pmatrix}V_0^1(y)\\ V_0^2(y)\end{pmatrix} = \begin{pmatrix}[\mu - \frac{1}{4}\xi\rho] - \frac{1}{2}y_2\\ \kappa(\theta - y_2) - \frac{1}{4}\xi^2\end{pmatrix}, \quad V_1(y) = \begin{pmatrix}\sqrt{y_2}\\ \xi\rho\sqrt{y_2}\end{pmatrix}$$

and the corresponding solutions to the ODEs are

$$\begin{aligned}\mathrm{e}^{sV_0}y &= \begin{pmatrix}y_1 + [\mu - \frac{1}{4}\xi\rho - \frac{1}{2}J]s + \frac{1}{2}\frac{y_2 - J}{\kappa}[\mathrm{e}^{-\kappa s} - 1]\right)\\ (y_2 - J)\mathrm{e}^{-\kappa s} + J\end{pmatrix},\\ \mathrm{e}^{sV_1}y &= \begin{pmatrix}y_1 + \frac{\left(\frac{1}{2}\xi\rho s + \sqrt{y_2}\right)_+^2 - y_2}{\xi\rho}\\ \left(\frac{1}{2}\xi\rho s + \sqrt{y_2}\right)_+^2\end{pmatrix},\\ \mathrm{e}^{sV_2}y &= \begin{pmatrix}y_1\\ \left(\frac{1}{2}\xi\sqrt{1-\rho^2}s + \sqrt{y_2}\right)_+^2\end{pmatrix},\end{aligned} \tag{6}$$

with $J = \frac{\kappa\theta - \frac{1}{4}\xi^2}{\kappa}$, as before and $y = (y_1, y_2) = (\log x_1, x_2)$. As is well-known, it is far preferable to use Heston in log-coordinates when simulating with the EM scheme. Although this is less critical in the cubature context, we still recommend (6) to avoid the numerical evaluation of $\exp(\cdot)$.

2.2. **SABR model.** The SABR model is given by

$$\begin{aligned}\mathrm{d}X_1(t,x) =& aX_2(t,x)(X_1(t,x))^\beta \mathrm{d}B_t^1\\ \mathrm{d}X_2(t,x) =& bX_2(t,x)\mathrm{d}\left(\rho B_t^1 + \sqrt{1-\rho^2}B_t^2\right),\end{aligned}$$

where $\frac{1}{2} \le \beta \le 1$, $a, b > 0$ and $-1 < \rho < 1$. [7] The corresponding vector fields are

$$\widetilde{V}_0(x) = \begin{pmatrix}0\\0\end{pmatrix}, \quad V_1(x) = \begin{pmatrix}ax_2x_1^\beta\\ b\rho x_2\end{pmatrix}, \quad V_2(x) = \begin{pmatrix}0\\ b\sqrt{1-\rho^2}x_2\end{pmatrix}$$

and so we get

$$\begin{pmatrix}V_0^1(x)\\ V_0^2(x)\end{pmatrix} = -\begin{pmatrix}\frac{1}{2}\sum_{j=1}^2 V_jV_j^1(x)\\ \frac{1}{2}\sum_{j=1}^2 V_jV_j^2(x)\end{pmatrix} = \begin{pmatrix}-\frac{1}{2}[a^2\beta x_2^2 x_1^{2\beta-1} + ab\rho x_2 x_1^\beta]\\ -\frac{1}{2}b^2x_2\end{pmatrix}.$$

The solutions to the ODEs corresponding to the vector fields V_1 and V_2 are

$$\mathrm{e}^{sV_1}x = \begin{pmatrix}g_1(s)\\ x_2\exp(b\rho s)\end{pmatrix},$$

$$\mathrm{e}^{sV_2}x = \begin{pmatrix}x_1\\ x_2\exp\left(b\sqrt{1-\rho^2}s\right)\end{pmatrix},$$

where

$$\begin{aligned}g_1(s) =& \left[(1-\beta)ax_2\frac{\mathrm{e}^{b\rho s}-1}{b\rho} + x_1^{1-\beta}\right]_+^{1/(1-\beta)}, \quad 0 < \beta < 1,\\ g_1(s) =& x_1\exp\left(ax_2\frac{\mathrm{e}^{b\rho s}-1}{b\rho}\right), \quad \beta = 1.\end{aligned}$$

For details on the uniqueness of g_1 we refer to the Appendix. Note also that the term $\frac{\mathrm{e}^{b\rho s}-1}{b\rho}$ appearing in g_1 should be understood in the limiting sense when $\rho = 0$, i.e. $\frac{\mathrm{e}^{b\rho s}-1}{b\rho}\Big|_{\rho=0} = s$. Concerning the solution to the ODE corresponding to V_0, let $H(s)$ be the first component of $\mathrm{e}^{sV_0}x$, i.e.

$$\mathrm{e}^{sV_0}x = \begin{pmatrix}H(s)\\ x_2\exp\left(-\frac{1}{2}b^2s\right)\end{pmatrix}.$$

It is impossible to find H in closed-form (unless $\rho = 0$ or $\beta = 1$). This means that applying the standard NV-scheme must involve the numerical solution of auxiliary ODEs. We shall see later that with the NV scheme with drift all involved ODEs can be solved in closed form.

[7] Although in the literature the SABR model is also considered for $0 < \beta < \frac{1}{2}$ we restrict ourselves to the case $\frac{1}{2} \le \beta \le 1$ in order to avoid difficulties regarding well-posedness of X, cf. [12].

3. Models accessible to SCFC and NV with drift

3.1. **Motivation.** In the classical NV scheme, only centered Gaussian (Brownian) increments are used to flow along the diffusion vector fields. Our main observation is that one can also use non-centered Gaussian increments; this affects the drift term and, chosen in a smart way, can sometimes render all auxiliary ODE to be solvable in closed form. To motivate the class of models for which this works, we illustrate how to systematically construct models accessible to SCFC from a fairly general two-factor stochastic volatility model given in Itô form by

$$\begin{aligned}\mathrm{d}X_1(t) =& A(X_1(t))B(X_2(t))\mathrm{d}B_t^1\\ \mathrm{d}X_2(t) =& C(X_2(t))\mathrm{d}t + D(X_2(t))\mathrm{d}B_t^1 + E(X_2(t))\mathrm{d}B_t^2,\end{aligned}$$

where $(X_1(0), X_2(0)) = (x_1, x_2)$ is kept fixed. In Stratonovich form this becomes (omitting the dependence on t in the drift and diffusion coefficients),

$$\begin{aligned}\mathrm{d}X_1(t) =& -\frac{1}{2}A(X_1)\left[A'(X_1)B^2(X_2) + D(X_2)B'(X_2)\right]\mathrm{d}t + A(X_1)B(X_2)\circ\mathrm{d}B_t^1\\ \mathrm{d}X_2(t) =& \left[C(X_2) - \frac{1}{2}\left[D(X_2)D'(X_2) + E(X_2)E'(X_2)\right]\right]\mathrm{d}t\\ &+ D(X_2)\circ\mathrm{d}B_t^1 + E(X_2)\circ\mathrm{d}B_t^2,\end{aligned}$$

In the subsequent analysis we shall exhibit a number of possible choices which lead to models accessible to SCFC. First we would like to choose the coefficients $A, \dots, E$ such that we can rewrite the first SDE as

$$\mathrm{d}X_1(t) = H_1(X_1)H_2(X_2)\mathrm{d}t + A(X_1)B(X_2)\circ\mathrm{d}\left(B_t^1 + \gamma_1 t\right)$$

for a constant γ_1 and functions H_1, H_2. Three possible ways to achieve this goal are

$$\text{(i) } A'(X_1) \propto 1, \quad \text{(ii) } D(X_2) = 0 \quad \text{or} \quad \text{(iii) } D(X_2)B'(X_2) \propto B(X_2).$$

Note that the Heston model is a particular example satisfying (i). However, in case (i) we would have $\gamma_1 = 0$ and since we want to illustrate the additional benefit of the NV scheme with drift over the classical NV scheme, we will not consider case (i) in any more detail. Moreover, since we would like the volatility factor to depend on the Brownian motion driving the stock, we will also skip case (ii) and concentrate on case (iii) which implies $\frac{B(X_2)}{B'(X_2)} \propto D(X_2)$ and $\gamma_1 > 0$. E.g. if we choose $D(x) = 1$, then $B(x) \propto \exp(cx)$, if $D(x) = x$, then $B(x) \propto x^c$ with $c \neq 0$, if $D(x) = x^q$, $0 \leq q < 1$, then $B(x) \propto \exp(\frac{c}{1-q}x^{1-q})$. We focus on the most natural choice (i.e. B not being of exponential type) and therefore pick

$$B(X_2) = aX_2^\alpha \quad \text{and} \quad D(X_2) = b\rho X_2.$$

These choices give us

$$\begin{aligned}\mathrm{d}X_1(t) =& -\frac{1}{2}A(X_1)A'(X_1)a^2X_2^{2\alpha}\mathrm{d}t + A(X_1)aX_2^\alpha\circ\mathrm{d}\left(B_t^1 - \frac{1}{2}\alpha b\rho t\right)\\ \mathrm{d}X_2(t) =& \left[C(X_2) - \frac{1}{2}\left[b^2\rho^2X_2 + E(X_2)E'(X_2)\right]\right]\mathrm{d}t + b\rho X_2\circ\mathrm{d}B_t^1 + E(X_2)\circ\mathrm{d}B_t^2.\end{aligned}$$

With

$$(V_0^1, V_0^2) = \left(-\frac{1}{2}A(x_1)A'(x_1)a^2x_2^{2\alpha}, C(x_2) - \frac{1}{2}[b^2\rho^2x_2 + E(x_2)E'(x_2)]\right),$$

define $h(t;x_2) = x_2 \mathrm{e}^{tV_0^2}$. We would like $\int_0^t h(s;x_2)^{2\alpha}\mathrm{d}s$ to have an explicit expression, since it will appear in the first component of $x\mathrm{e}^{tV_0}$. Possible cases are (i) $h(t;x_2) \propto t+x_2$, (ii) $h(t;x_2) \propto \mathrm{e}^{ct}$ or (iii) very specific cases like $h(t;x_2) = p(t,x_2)^{\frac{1}{2\alpha}}$ with p nice. Both (i) and (ii) lead to C being affine and E being affine or of square root type. We pick

$$C(X_2) = \kappa(\theta - X_2) \quad \text{and} \quad E(X_2) = b\sqrt{1-\rho^2}X_2.$$

and we shall later motivate why we let E be linear. With these choices we can write

$$\begin{aligned}
\mathrm{d}X_1(t) =& -\frac{1}{2}A(X_1)A'(X_1)a^2X_2^{2\alpha}\mathrm{d}t + A(X_1)aX_2^{\alpha} \circ \mathrm{d}\left(B_t^1 - \frac{1}{2}\alpha b\rho t\right)\\
\mathrm{d}X_2(t) =& \left[\kappa(\theta - X_2) - \frac{1}{2}b^2X_2\right]\mathrm{d}t + b\rho X_2 \circ \mathrm{d}B_t^1 + b\sqrt{1-\rho^2}X_2 \circ \mathrm{d}B_t^2.
\end{aligned}$$

We now rewrite the second SDE in the form

$$\mathrm{d}X_2(t) = H(X_2)\mathrm{d}t + +b\rho X_2 \circ \mathrm{d}\left(B_t^1 + \gamma_1 t\right) + b\sqrt{1-\rho^2}X_2 \circ \mathrm{d}\left(B_t^2 + \gamma_2 t\right)$$

with $\gamma_1 = -\frac{1}{2}\alpha b\rho$ (as in the first SDE) and with γ_2 a constant such that $H(X_2)$ becomes as simple as possible (recall that we want $\int_0^t (x_2\mathrm{e}^{sH})^{2\alpha}\mathrm{d}s$ to be explicit). We get

$$\begin{aligned}
\mathrm{d}X_2(t) = \kappa\theta\mathrm{d}t + b\rho X_2 \circ \mathrm{d}\left(B_t^1 - \frac{1}{2}\alpha b\rho t\right)&\\
+ b\sqrt{1-\rho^2}X_2 \circ \mathrm{d}\left(B_t^2 + \frac{\alpha b\rho^2 - 2\kappa/b - b}{2\sqrt{1-\rho^2}}t\right)&.
\end{aligned}$$

Note that if we would have chosen E to be affine but not linear or have chosen E of square root type, we would not be able to make H so simple. (For the same reason we have chosen D linear and not generally affine.)

Finally, $A(X_1)$ is left to choose. Since we want to end up with a model accessible to SCFC, the function A should be such that $x_1\mathrm{e}^{tA}$ is explicit, which means that we want $\int_\cdot^x \frac{\mathrm{d}y}{A(y)}$ to have an explicit inverse. Also, $x_1\mathrm{e}^{tA\cdot A'}$ should be explicit. The obvious candidates are $A(X_1) = X_1^\beta$, $A(X_1) = \mathrm{e}^{cX_1}$ and $A(X_1) = X_1 + c$ and all lead to models that are accessible to SCFC. As a case study we choose the first one and apply the NV scheme with drift to the resulting model in the next section.

3.2. **Generalized SABR (with shifted log-normal 2nd factor).** In the previous section we constructed a particular class of SV-models which are accessible to SCFC, namely:

$$\begin{aligned}
\mathrm{d}X_1(t) =& aX_2(t)^\alpha X_1(t)^\beta \mathrm{d}B_t^1\\
\mathrm{d}X_2(t) =& \kappa(\theta - X_2(t))\mathrm{d}t + bX_2(t)\left(\rho\mathrm{d}B_t^1 + \sqrt{1-\rho^2}\mathrm{d}B_t^2\right),
\end{aligned}$$

with $X_1(0) = x_1$ and $X_2(0) = x_2$. We assume that the parameters satisfy $\frac{1}{2} \leq \beta \leq 1$, $\theta, \kappa \geq 0$, $\alpha > 0$, $a, b > 0$, $-1 < \rho < 1$. Details surrounding well-posedness, integrability properties and martingale properties can be found in Lions and Musiela [12, 13]. A simple application of Itô's formula shows that

$$X_2(t) = x_2\mathrm{e}^{-(\kappa+\frac{1}{2}b^2)t+bW_t} + \kappa\theta\int_0^t \mathrm{e}^{-(\kappa+\frac{1}{2}b^2)(t-s)}\mathrm{e}^{b(W_t-W_s)}\mathrm{d}s,$$

where $W_t = \rho B_t^1 + \sqrt{1-\rho^2} B_t^2$ and thus $X_2(t) > 0$ for all $t \geq 0$, provided $x_2 > 0$.

We shall now give all the ODE solutions that are required to apply the NV scheme (with drift). First note that the vector fields V_1 and V_2 corresponding to B^1 and B^2 are given by

$$V_1 = \begin{pmatrix} a x_2^\alpha x_1^\beta \\ b\rho x_2 \end{pmatrix}, \quad V_2 = \begin{pmatrix} 0 \\ b\sqrt{1-\rho^2} x_2 \end{pmatrix}$$

and the ODE solutions are

$$\mathrm{e}^{sV_1} x = \begin{pmatrix} g_1(s) \\ x_2 \exp(b\rho s) \end{pmatrix},$$

$$\mathrm{e}^{sV_2} x = \begin{pmatrix} x_1 \\ x_2 \exp\left(b\sqrt{1-\rho^2} s\right) \end{pmatrix},$$

where

$$\begin{aligned} g_1(s) =& \left[(1-\beta) a x_2^\alpha \frac{\mathrm{e}^{\alpha b \rho s} - 1}{\alpha b \rho} + x_1^{1-\beta} \right]_+^{1/(1-\beta)}, \quad \frac{1}{2} \leq \beta < 1, \\ g_1(s) =& x_1 \exp\left(a x_2^\alpha \frac{\mathrm{e}^{\alpha b \rho s} - 1}{\alpha b \rho} \right), \quad \beta = 1. \end{aligned}$$

Note again that the term $\frac{\mathrm{e}^{\alpha b \rho s} - 1}{\alpha b \rho}$ appearing in g_1 should be understood in the limiting sense when $\rho = 0$, i.e. $\left.\frac{\mathrm{e}^{\alpha b \rho s} - 1}{\alpha b \rho}\right|_{\rho=0} = s$. The Itô drift vector field $\widetilde{V}_0$ and Stratonovich drift vector field V_0 of X are given by

$$\widetilde{V}_0(x) = \begin{pmatrix} 0 \\ \kappa\theta - \kappa x_2 \end{pmatrix}, \quad V_0(x) = \begin{pmatrix} -\frac{1}{2} a^2 \beta x_2^{2\alpha} x_1^{2\beta-1} - \frac{1}{2} \alpha a b \rho x_2^\alpha x_1^\beta \\ \kappa\theta - (\kappa + \frac{1}{2} b^2) x_2 \end{pmatrix}.$$

We have $\mathrm{e}^{sV_0} x = (H(s), h(s))^T$ with

$$h(s) = \left(x_2 - \frac{\kappa\theta}{\kappa + \frac{1}{2} b^2} \right) \mathrm{e}^{-(\kappa + \frac{1}{2} b^2) s} + \frac{\kappa\theta}{\kappa + \frac{1}{2} b^2}$$

and H needs to be numerically solved. (We have already pointed to this difficulty when we discussed the classical SABR example earlier on.)

Let us now show that by using Brownian increments with drift, this problem can be resolved: all necessary flows can be computed in closed form. Recall from the previous section that we can rewrite X as

$$\begin{aligned} \mathrm{d}X_1(t) =& -\frac{1}{2} a^2 \beta X_2^{2\alpha} X_1^{2\beta-1} \mathrm{d}t + a X_2^\alpha X_1^\beta \circ \mathrm{d}\left(B_t^1 + \gamma_1 t\right) \\ \mathrm{d}X_2(t) =& \kappa\theta \mathrm{d}t + b\rho X_2 \circ \mathrm{d}\left(B_t^1 + \gamma_1 t\right) + b\sqrt{1-\rho^2} X_2 \circ \mathrm{d}\left(B_t^2 + \gamma_2 t\right). \end{aligned}$$

with

$$\gamma_1 = -\frac{1}{2} \alpha b \rho \quad \text{and} \quad \gamma_2 = \frac{\alpha b \rho^2 - 2\kappa/b - b}{2\sqrt{1-\rho^2}}.$$

We see here that the assumption $-1 < \rho < 1$ is crucial. Note that the vector fields corresponding to $B_t^1 + \gamma_1 t$ and $B_t^2 + \gamma_2 t$, respectively, are V_1 and V_2. Denote by $V_0^{(\gamma)}$ the remaining part, i.e.

$$V_0^{(\gamma)}(x) = \begin{pmatrix} -\frac{1}{2} a^2 \beta x_2^{2\alpha} x_1^{2\beta-1} \\ \kappa\theta \end{pmatrix}.$$

Then we have $\mathrm{e}^{sV_0^{(\gamma)}}x = (g_0, \kappa\theta s + x_2)^T$ with (cf. Appendix)

$$\begin{aligned}
g_0(s) =& \left(-a^2\beta(1-\beta)P(s) + x_1^{2(1-\beta)}\right)_+^{\frac{1}{2(1-\beta)}}, \quad \frac{1}{2} < \beta < 1,\\
g_0(s) =& x_1 \exp\left(-\frac{1}{2}a^2 P(s)\right), \quad \beta = 1,\\
g_0(s) =& -\frac{1}{4}a^2 P(s) + x_1, \quad \beta = \frac{1}{2},
\end{aligned}$$

where

$$P(s) = \frac{1}{(2\alpha+1)\kappa\theta}\left((\kappa\theta s + x_2)^{2\alpha+1} - x_2^{2\alpha+1}\right).$$

Note that when $\kappa = 0$ or $\theta = 0$, $P(s)$ should be understood in the limiting sense, i.e., $P(s) = sx_2^{2\alpha}$ for $\kappa = 0$ or $\theta = 0$.

Remark 2. Since the SABR model is a special case of the model presented here – corresponding to $\alpha = 1$, $\kappa = 0$ – the semi-closed form NV algorithm developed above can be, in particular, applied to the SABR model.

Remark 3. The scheme presented above is well-defined for $\beta \neq \frac{1}{2}$. For $\beta = \frac{1}{2}$, the stock can become negative in one step of the method, which would lead to imaginary values in the next step. In order to retain convergence of the method, one could, in this case, adapt the method by truncating the solution at 0, as for the case $\beta \neq \frac{1}{2}$. In any case, second order convergence will be lost when boundary effects become relevant, see Lord et al. [14] for an extended discussion of truncation in the context of an Euler scheme. We will come back to this issue in both more detail and more generality in Section 4.

3.3. **Girsanov transform.** We have seen for the example above, that if one uses the standard NV scheme, the flow of the drift vector field is not available in closed form. Besides using the 'drift trick' as we did above, it is also possible to absorb this drift in a change-of-measure; the details of this are outlined below. There is, however, a serious downside to this: the Girsanov-density which appears due to the change-of-measure will *add significantly to the variance of the object to be sampled*. Thus, without further variance reduction, we do not advertise the use of the Girsanov transform in this context.

Let $Y = (Y_1, Y_2)$ be the process defined by

$$\begin{aligned}
\mathrm{d}Y_1(t) =& aY_2(t,x)^\alpha Y_1(t)^\beta\left(-\gamma_1\mathrm{d}t + \mathrm{d}B_t^1\right)\\
\mathrm{d}Y_2(t) =& \kappa(\theta - Y_2(t))\mathrm{d}t + b\rho Y_2(t)\left(-\gamma_1\mathrm{d}t + \mathrm{d}B_t^1\right) + b\sqrt{1-\rho^2}Y_2(t)\left(-\gamma_2\mathrm{d}t + \mathrm{d}B_t^2\right).
\end{aligned}$$

and let $\mathbb{P}$ be the probability measure under which $B = (B^1, B^2)$ is a 2-dimensional standard Brownian motion. Define the probability measure $\mathbb{Q}$ by $\frac{\mathrm{d}\mathbb{Q}}{\mathrm{d}\mathbb{P}}|_{\mathcal{F}_t} = \mathcal{E}(t)$, where

$$\mathcal{E}(t) = \exp\left(\gamma_1 B_t^1 + \gamma_2 B_t^2 - \frac{1}{2}(\gamma_1^2 + \gamma_2^2)t\right).$$

Then by Girsanov, under $\mathbb{Q}$, (B^1, B^2) is equal in law to a 2-dimensional standard Brownian motion plus constant drift equal to (γ_1, γ_2). Hence under $\mathbb{Q}$, Y is equal in law to X under $\mathbb{P}$ and we have for $f : \mathbb{R}^2 \to \mathbb{R}$ measurable,

$$\mathbb{E}^{\mathbb{P}}[f(X(t))] = \mathbb{E}^{\mathbb{Q}}[f(Y_1(t), Y_2(t))] = \mathbb{E}^{\mathbb{P}}[f(Y_1(t), Y_2(t))\mathcal{E}(t)].$$

Hence a "weighted" NV scheme with explicit solutions to all ODEs can be obtained by using the NV scheme for the process Y and then multiplying the payoff $f(Y_1(t), Y_2(t))$, as is done in importance sampling, by $\mathcal{E}(t)$. Note that all the ODE solutions corresponding to the NV scheme for Y are explicit, since the vector fields corresponding to Y are $V_0^{(\gamma)}$, V_1 and V_2.

To back up our claim about the additional variance caused by the Girsanov density $\mathcal{E}(t)$, note that $\mathbb{V}(\mathcal{E}(t)) = e^{(\gamma_1^2+\gamma_2^2)t} - 1$, which is only negligible when γ_1 and γ_2 are close to zero.

3.4. **A multi-dimensional version.** Let us illustrate how the techniques introduced (until now in the context of 2-dimensional models) remain feasible in typical higher dimensional models (what we have in mind here is some multi asset SV model). Since it is precisely the curse of dimensionality that forces one to use stochastic methods (rather than PDE methods, say) we want to be fully explicit in showing how our ideas are implemented in higher dimensions. More specifically, we shall consider the following multi-dimensional version of our SABR-type model: for $i = 1, \dots, N$,

$$\begin{aligned} \mathrm{d}X_i(t) =& a_i Y_i(t)^{\alpha_i} X_i(t)^{\beta_i} \mathrm{d}\widetilde{B}_t^i \\ \mathrm{d}Y_i(t) =& \kappa_i(\theta_i - Y_i(t))\mathrm{d}t + b_i Y_i(t)\mathrm{d}\widetilde{W}_t^i, \end{aligned}$$

with $X_i(0) = x_i$, $Y_i(0) = y_i$ and $\frac{1}{2} \le \beta_i \le 1$, $\theta_i, \kappa_i \ge 0$, $\alpha_i > 0$, $a_i, b_i > 0$, $-1 < \rho_i < 1$. Here $\left(\begin{smallmatrix}\widetilde{\mathbf{B}} \\ \widetilde{\mathbf{W}}\end{smallmatrix}\right)$, with $\widetilde{\mathbf{B}} = (\widetilde{B}^1, \dots, \widetilde{B}^N)^T$ and $\widetilde{\mathbf{W}} = (\widetilde{W}^1, \dots, \widetilde{W}^N)^T$ is a 2N-dimensional Brownian motion with correlation matrix given by ρ which we assume to be positive-definite. Let $\sqrt{\rho}$ be the unique lower-triangular matrix such that $\sqrt{\rho}\sqrt{\rho}^T = \rho$ (Choleski decomposition). Then $\left(\begin{smallmatrix}\widetilde{\mathbf{B}} \\ \widetilde{\mathbf{W}}\end{smallmatrix}\right) \overset{(D)}{=} \sqrt{\rho}\left(\begin{smallmatrix}\mathbf{B} \\ \mathbf{W}\end{smallmatrix}\right)$, where $\left(\begin{smallmatrix}\mathbf{B} \\ \mathbf{W}\end{smallmatrix}\right)$, with $\mathbf{B} = (B^1, \dots, B^N)^T$ and $\mathbf{W} = (W^1, \dots, W^N)^T$ is a $2N$-dimensional standard Brownian motion. Hence we can write for $i = 1, \dots, N$,

$$\begin{aligned} \mathrm{d}X_i(t) =& a_i Y_i(t)^{\alpha_i} X_i(t)^{\beta_i} \mathrm{d}\left(\sum_{j=1}^N \sqrt{\rho}_{i,j} B_t^j + \sum_{j=1}^N \sqrt{\rho}_{i,N+j} W_t^j\right) \\ \mathrm{d}Y_i(t) =& \kappa_i(\theta_i - Y_i(t))\mathrm{d}t + b_i Y_i(t)\mathrm{d}\left(\sum_{j=1}^N \sqrt{\rho}_{N+i,j} B_t^j + \sum_{j=1}^N \sqrt{\rho}_{N+i,N+j} W_t^j\right). \end{aligned}$$

Let for $j = 1, \dots, N$, V_j and U_j be the vector fields corresponding to B^j and W^j, respectively. We have

$$V_j(x,y) = \begin{pmatrix} a_1 y_1^{\alpha_1} x_1^{\beta_1} \sqrt{\rho}_{1,j} \\ b_1 y_1 \sqrt{\rho}_{N+1,j} \\ \vdots \\ a_N y_N^{\alpha_N} x_N^{\beta_N} \sqrt{\rho}_{N,j} \\ b_N y_N \sqrt{\rho}_{2N,j} \end{pmatrix}, \quad U_j(x,y) = \begin{pmatrix} 0 \\ b_1 y_1 \sqrt{\rho}_{N+1,N+j} \\ \vdots \\ 0 \\ b_N y_N \sqrt{\rho}_{2N,N+j} \end{pmatrix}.$$

It follows that

$$V_0(x,y) = \begin{pmatrix} -\frac{1}{2}p_1 a_1^2 \beta_1 y_1^{2\alpha_1} x_1^{2\beta_1 - 1} - \frac{1}{2} q_1 \alpha_1 a_1 b_1 y_1^{\alpha_1} x_1^{\beta_1} \\ \kappa_1 \theta_1 - (\kappa_1 + \frac{1}{2} b_1^2 r_1) y_1 \\ \vdots \\ -\frac{1}{2}p_N a_N^2 \beta_N y_N^{2\alpha_N} x_N^{2\beta_N - 1} - \frac{1}{2} q_N \alpha_N a_N b_N y_N^{\alpha_N} x_N^{\beta_N} \\ \kappa_N \theta_N - (\kappa_N + \frac{1}{2} b_N^2 r_N) y_N \end{pmatrix},$$

where for $i = 1, \dots, N$,

$$\begin{aligned} p_i =& \sum_{j=1}^{N} \left[\left(\sqrt{\rho}_{i,j} \right)^2 + \left(\sqrt{\rho}_{i,N+j} \right)^2 \right] = \rho_{i,i} = 1, \\ q_i =& \sum_{j=1}^{N} \left[\sqrt{\rho}_{N+i,j} \sqrt{\rho}_{i,j} + \sqrt{\rho}_{N+i,N+j} \sqrt{\rho}_{i,N+j} \right] = \sum_{j=1}^{N} \sqrt{\rho}_{N+i,j} \sqrt{\rho}_{i,j}, \\ r_i =& \sum_{j=1}^{N} \left[\left(\sqrt{\rho}_{N+i,j} \right)^2 + \left(\sqrt{\rho}_{N+i,N+j} \right)^2 \right] = \rho_{N+i,N+i} = 1. \end{aligned}$$

We would like to write the system (X_i, Y_i), $i = 1, \dots, N$ in the following way

$$\begin{aligned} \mathrm{d}X_i(t) =& -\frac{1}{2} a_i^2 \beta_i Y_i(t)^{2\alpha_i} X_i(t)^{2\beta_i - 1} \mathrm{d}t + a_i Y_i(t)^{\alpha_i} X_i(t)^{\beta_i} \\ & \times \left(\sum_{j=1}^{N} \sqrt{\rho}_{i,j} \circ \mathrm{d} \left\{ B_t^j + \gamma_j t \right\} + \sum_{j=1}^{N} \sqrt{\rho}_{i,N+j} \circ \mathrm{d} \left\{ W_t^j + \delta_j t \right\} \right) \\ \mathrm{d}Y_i(t) =& \kappa_i \theta_i \mathrm{d}t + b_i Y_i(t) \\ & \times \left(\sum_{j=1}^{N} \sqrt{\rho}_{N+i,j} \circ \mathrm{d} \left\{ B_t^j + \gamma_j t \right\} + \sum_{j=1}^{N} \sqrt{\rho}_{N+i,N+j} \circ \mathrm{d} \left\{ W_t^j + \delta_j t \right\} \right), \end{aligned}$$

for a certain $\vec{\gamma} = (\gamma_1, \dots, \gamma_N)$ and $\vec{\delta} = (\delta_1, \dots, \delta_N)$. Looking at V_0, we should choose $\vec{\gamma}$ and $\vec{\delta}$ such that for $i = 1, \dots, N$,

$$\begin{aligned} \sum_{j=1}^{N} \sqrt{\rho}_{i,j} \gamma_j + \sum_{j=1}^{N} \sqrt{\rho}_{i,N+j} \delta_j =& -\frac{1}{2} q_i \alpha_i b_i, \\ \sum_{j=1}^{N} \sqrt{\rho}_{N+i,j} \gamma_j + \sum_{j=1}^{N} \sqrt{\rho}_{N+i,N+j} \delta_j =& -\frac{\kappa_i + \frac{1}{2} b_i^2}{b_i}. \end{aligned}$$

These are $2N$ linear equations with $2N$ unknowns. It follows that there exists a unique $\vec{\gamma}$ and $\vec{\delta}$ such that the above equalities are satisfied if $\sqrt{\rho}$ is of full rank. This is the multi-dimensional analogue of the condition $-1 < \rho < 1$ of the previous section. We see that in order to apply the NV scheme with drift, we need to find the flows corresponding to the vector fields $V_j(x,y)$, $U_j(x,y)$ and

$$V_0^{(\vec{\gamma}, \vec{\delta})}(x,y) = \begin{pmatrix} -\frac{1}{2} a_1^2 \beta_1 y_1^{2\alpha_1} x_1^{2\beta_1 - 1} \\ \kappa_1 \theta_1 \\ \vdots \\ -\frac{1}{2} a_N^2 \beta_N y_N^{2\alpha_N} x_N^{2\beta_N - 1} \\ \kappa_N \theta_N \end{pmatrix}.$$

All the solutions to these ODEs can be found explicitly as in Section 3.2.

3.5. **Numerical analysis of our NV scheme with drift.** In this section we want to prove second order weak convergence of $\mathbb{E}\left(f\left(\overline{X}^{(NVd),K}(T,x)\right)\right)$ as $K\to\infty$ for smooth f with $\overline{X}^{(NVd),K}(T,x)$ given by (4)-(5). As in the original proof by Ninomiya and Victoir [18] for the classical NV scheme, we use a Taylor expansion to get the local order of the weak error by comparison with the known local weak order of the classical Ninomiya-Victoir scheme. Let $\overline{X}^{(NV),K}(T/K,x)$ be as in Section 1.2. Then the difference in the Taylor expansion of the expectation in one step is given by

(7)

$$\begin{aligned}
&\mathbb{E}\left(f\left(\overline{X}^{(NVd),K}(T/K,x)\right)\middle|\,\Lambda_1=-1\right)-\mathbb{E}\left(f\left(\overline{X}^{(NV),K}(T/K,x)\right)\middle|\,\Lambda_1=-1\right)=\\
&\qquad\left[\frac{1}{2}\sum_{1\le i<j\le d}\gamma_i\gamma_j V_iV_jf(x)-\frac{1}{2}\sum_{1\le j<i\le d}\gamma_i\gamma_j V_iV_jf(x)+\right.\\
&\qquad+\frac{1}{4}\sum_{1\le i<j\le d}\gamma_i V_i(V_j)^2f(x)-\frac{1}{4}\sum_{1\le j<i\le d}\gamma_i V_i(V_j)^2f(x)+\\
&\left.+\frac{1}{4}\sum_{1\le i<j\le d}\gamma_j (V_i)^2V_jf(x)-\frac{1}{4}\sum_{1\le j<i\le d}\gamma_j (V_i)^2V_jf(x)\right](T/K)^2+\mathcal{O}\left((T/K)^3\right).
\end{aligned}$$

When we condition on $\Lambda_1=1$, only the order of the indices are swapped. By the peculiar structure of (7), this means that the signs of all terms of the difference change, i.e.,

$$\begin{aligned}
&\mathbb{E}\left(f\left(\overline{X}^{(NVd),K}(T/K,x)\right)\middle|\,\Lambda_1=1\right)-\mathbb{E}\left(f\left(\overline{X}^{(NV),K}(T/K,x)\right)\middle|\,\Lambda_1=1\right)=\\
&-\left[\mathbb{E}\left(f\left(\overline{X}^{(NVd),K}(T/K,x)\right)\middle|\,\Lambda_1=-1\right)-\mathbb{E}\left(f\left(\overline{X}^{(NV),K}(T/K,x)\right)\middle|\,\Lambda_1=-1,\right)\right]\\
&\qquad\qquad+\mathcal{O}\left((T/K)^3\right).
\end{aligned}$$

Thus, taking the unconditional expectation in Λ_1 gives

$$\mathbb{E}\left(f\left(\overline{X}^{(NVd),K}(T/K,x)\right)\right)-\mathbb{E}\left(f\left(\overline{X}^{(NV),K}(T/K,x)\right)\right)=\mathcal{O}\left((T/K)^3\right),$$

and second order convergence of the Ninomiya-Victoir scheme with drift follows exactly as in the case without drift.

Remark 4. The drift trick also works for classical cubature on Wiener space as in Lyons-Victoir [15], i.e., when $W^0(t)\equiv t$, simply by adding the same drift $t\gamma_i$ to the i^{th} component of the cubature path W. This procedure retains the original order of convergence for the given cubature formula, as can be trivially seen by comparing the ODEs with and without drift. Note that in the Ninomiya-Victoir scheme a slightly more difficult argument as discussed above is necessary, since here the "time" component W^0 is not linear.

4. Issues related to lack of smoothness

As in most works on numerics on SDEs, we assumed in the introduction that our payoff function f and all vector fields $V_0,V_1,\dots,V_d$ are smooth, with bounded

derivatives of any order. If this is not the case, as happens in various applications, additional arguments may be necessary. In the context related to this paper, Lyons and Victoir [15] handle (non-smooth) Lipschitz payoffs with a non-equidistant mesh, based on Kusuoka-Stroock estimates for hypoelliptic diffusions; but they still assume smooth vector fields.

When the vector fields are not smooth, or not smooth up to the boundary, as is the case in the CIR/Heston model and also in most of the models considered in this paper, one faces another potential problem: namely, ODE solutions as required in the NV scheme, may cease to exist. To be specific, let us play this through in the CIR model. In the notation of subsection 2.1, where it is used as second component of the Heston model, its dynamics are given by

$$dX_2(t,x) = \kappa(\theta - X_2(t,x))dt + \xi\sqrt{X_2(t,x)}dW_t.$$

When rewritten in Stratonovich form (as was done in that subsection) its drift vectorfield takes the form $V_0(x) = \kappa(\theta - x) - \xi^2/4$. The integral curve of this vector field, i.e. ODE solution required in the NV scheme, is given by

$$e^{sV_0}x = (x - J)e^{-\kappa s} + J, \quad J = \frac{\kappa\theta - \frac{1}{4}\xi^2}{\kappa}.$$

Clearly then, a problem occurs when $J < 0$. Indeed, for x small and/or s large enough, $e^{sV_0}x$ will be negative, and the next step in the NV scheme, where one needs to approximate the action of the diffusion vectorfield $\xi\sqrt{\cdot}$, cannot be carried out. On the other hand, the NV scheme remains well-defined as long as $\xi^2 \leq 4\kappa\theta$, as then the solution is guaranteed to remain non-negative along all possible trajectories. (Note that this does not rule out the possibility that the order of convergence is actually worse than two and indeed this breakdown was observed numerically in [6].) It is interesting to note that the well-known *Feller condition*, $\xi^2 \leq 2\kappa\theta$, which guarantees a.s. positivity of the CIR process, implies well-definedness of the NV scheme, but not vice-versa.

The bottom line of this discussion is that in most models, singularities only appear at the boundary of the domain of the solution. In many cases, the rule of "not feeling the boundary" holds in the sense that in practice, the contribution of paths which come close to the boundary is often quantitatively negligible. Then, the Ninomiya-Victoir method and any variant of it should work well, empirically exhibiting second order convergence. Supported by the numerical evidence, we argue that this is true for many stochastic-local volatility models, such as the SABR-type models primarily considered in this article. However, the rule of "not feeling the boundary" does not hold universally, but depends on the choice of parameters. If one applies the SABR model in an interest rate context, instead of an equity context, then boundary effects may well matter, as discussed, for instance, by Chen et al. [5] for one-dimensional SABR-LIBOR models. In this case, we would ideally employ the techniques of Alfonsi. In [1] he presents a variation of the NV scheme for the CIR/Heston model, which - based on a moment matching algorithm in the vicinity of the CIR singularity - retains second order convergence even close to the boundary. As of now,the method is limited by the required (explicit) knowledge of moments of the solution. Future research will be necessary to compute, or at least do so approximately, the required moments in the relevant classes of non-smooth models, beyond the familiar case of CIR/Heston.

5. Numerical results

In this section we report the results of our numerical experiments. For this we have chosen three models: the SABR model, the generalized SABR model and the multi-dimensional generalized SABR model. The numerical results for these models are given in Section 5.1, Section 5.2 and Section 5.3, respectively. For each of the models, we compare the NV scheme with drift to the regular NV scheme and the Euler scheme and in all the experiments reported in this paper, we used Quasi Monte-Carlo for the integration. In order to check the order of the three schemes, we first give plots of the relative discretization error against the number of time steps K. For these plots the number of 'simulated trajectories' M is chosen such that the integration error is negligible compared to the discretization error. (Since we are using Quasi Monte Carlo, M is strictly speaking not the number of simulations, but the size of the finite low-discrepancy sequence used for the computation.) In order to compute these errors we of course need to know the 'true value' (or a good estimate of it). For the two experiments involving the generalized SABR model, the true value was obtained by running the code longer than reported in the plots. However, in the case of the SABR model, the true value was estimated by extrapolation of the values obtained by the code (the SABR formula is not exact enough!)

Results comparing the discretization error against the number of time steps, might not really seem practically relevant, since they only relate the number of time-steps for different numerical methods, but not the corresponding computational cost or computer time. Therefore we also give tables of the computational time of the different schemes. If we want to compare the run-time for different methods, we need to ensure the fairness of the comparison, i.e., we need to compare the different methods with parameters giving similar computational errors. We basically have two parameters for the numerical method, namely the number of time-steps K and the number of simulated trajectories M – where every trajectory is given by a $D(K)$-dimensional vector from a low discrepancy sequence, in our case the Sobol-sequence. (Here, $D(K)$ depends on both the model and the method.) Unlike for Monte Carlo simulation, where accurate, but probabilistic error estimates are available, there is no simple error estimation procedure for QMC (as far as we are aware). Therefore, it is not obvious how to choose the number of trajectories for a comparison of run-times. Choosing the same number of trajectories for every method might not be appropriate, because some methods might yield "rougher" integration problems, requiring a higher number of trajectories for a comparable precision.

In our comparison we proceeded as follows. For a given model and method we first choose K (and M very large) such that the relative discretization error is around 10^{-3}. For simplicity, we take the computations run for producing the plots, which means that K is a power of two. Then we start with $M_0 = 1000$ trajectories, run the method, and double the number M of trajectories until the observed (absolute) error is *consistently* closer than 2×10^{-5} to the true (absolute) error for the fixed given K.

To explain this in more detail, first recall that the computational error in (Quasi) Monte Carlo simulations for SDEs splits into two parts: $\text{Error} = \text{Error}_{\text{dis}} + \text{Error}_{\text{int}}$. Here, $\text{Error}_{\text{dis}}$ stems from the time-discretization of the SDE. This part of the error is controlled by K. $\text{Error}_{\text{int}}$ is the error from the integration, i.e., from the numerical

computation of the expectation of the solution to the discretized SDE. This error part is controlled by M. In the comparison procedure, we first fix K such that the relative discretization error $\text{Error}_{\text{dis}}/C$ (C denoting the true result) is around 10^{-3}. Then we choose M (by a doubling procedure) such that $\text{Error}_{\text{int}} \leq 2 \times 10^{-5}$. (Since in all cases $C \approx 0.1$, this means we choose the integration error to be one fifth of the discretization error.)

In the following tables, we report the found parameters, the corresponding relative error and the computational time in seconds – all computations where performed on the same computer, a Toshiba laptop with 6 GB RAM and four Intel Core i7 CPUs with 1.6 GHz. Notice that the run-time scales linearly with the number M of trajectories.

5.1. **SABR.** In this section we give the results corresponding to the experiment with the SABR model. Recall that this SV-model is given by

$$\begin{aligned} \mathrm{d}X_1(t) =& aX_2(t)X_1(t)^{\beta}\mathrm{d}B_t^1 \\ \mathrm{d}X_2(t) =& bX_2(t)\left(\rho \mathrm{d}B_t^1 + \sqrt{1-\rho^2}\mathrm{d}B_t^2\right), \end{aligned}$$

with $X_1(0) = x_1$ and $X_2(0) = x_2$ and where the parameters satisfy $\frac{1}{2} \leq \beta \leq 1$, $a, b > 0$, $-1 < \rho < 1$. The parameters chosen for the experiment are $\beta = 0.9$, $a = 1.0$, $b = 0.4$, $\rho = -0.7$, $x_1 = 1.0$, $x_2 = 0.3$. As the derivative we choose a (European) call option with maturity time $T = 1.0$ and strike price $K = 1.05$. For simplicity we assume that the interest rate is zero. The corresponding estimated 'true result' is 0.09400046.

In Figure 1 the convergence rates of the three schemes is graphically displayed. We clearly see the second-order convergence of the two NV schemes compared to the first-order convergence of the Euler method.

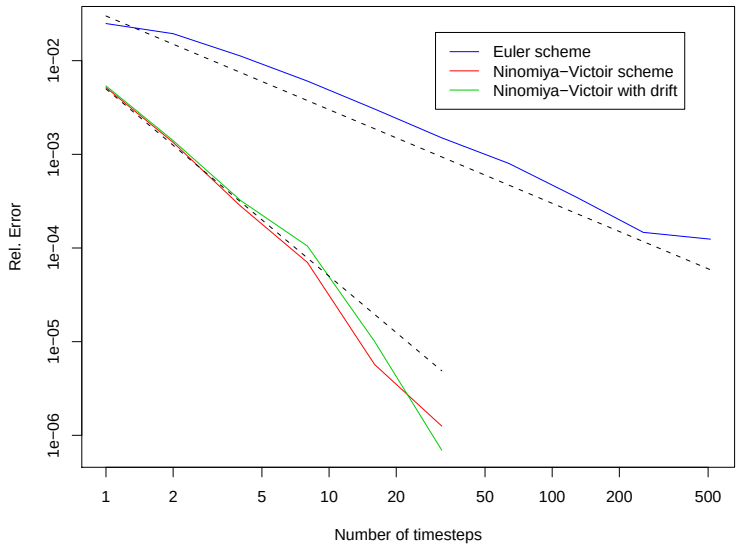

FIGURE 1. Order of convergence for the SABR model.

Method	K	M	Rel. Error	Time
Euler	32	512000	0.00150	5.87 sec
Ninomiya-Victoir	2	512000	0.00134	2.44 sec
NV with drift	2	128000	0.00140	0.28 sec

TABLE 1. Computational time for the SABR model

In Table 1 the timings are reported for the SABR model. Notice that the Ninomiya-Victoir method both in its original form and in its variant are clearly more efficient than the Euler method, by a factor two or three. On the other hand, the simpler structure of the Ninomiya-Victoir method with drift results in a considerable speed up if compared with the original method. Notice that a speed up with a factor two would even hold if we reject the empirically found choice of M for the drift variant and use the same number of trajectories as for the other two methods.

5.2. **Generalized SABR.** In this section we give the results corresponding to the experiment with the generalized SABR model of Section 3.2. For convenience we restate this SV-model:

$$\begin{aligned}\mathrm{d}X_1(t) =& aX_2(t)^\alpha X_1(t)^\beta \mathrm{d}B_t^1\\ \mathrm{d}X_2(t) =& \kappa(\theta - X_2(t))\mathrm{d}t + bX_2(t)\left(\rho \mathrm{d}B_t^1 + \sqrt{1-\rho^2}\mathrm{d}B_t^2\right),\end{aligned}$$

with $X_1(0) = x_1$ and $X_2(0) = x_2$ and $\frac{1}{2} \leq \beta \leq 1$, $\theta > 0$, $\kappa \geq 0$, $\alpha > 0$, $a, b > 0$, $-1 < \rho < 1$. For our experiment, we choose the parameters as follows: $\beta = 1.0$, $\theta = 0.3$, $\kappa = 2.0$, $\alpha = 0.5$, $a = 1.0$, $b = 0.5$, $\rho = -0.7$, $x_1 = 1.0$ and $x_2 = 0.2$. We further pick the same call option as in Section 5.1. The estimated 'true result' is 0.1767505855.

In Figure 2 the discretization error against the number of time steps is plotted. Again, we see the second-order convergence of the two NV schemes compared to the first-order convergence of the Euler method.

Method	K	M	Rel. Error	Time
Euler	32	8192000	0.00174	91.94 sec
Ninomiya-Victoir	4	2048000	0.00204	13.93 sec
NV with drift	4	1024000	0.00104	2.88 sec

TABLE 2. Computational time for the generalized SABR model

In Table 2 the timings are reported for the generalized SABR model. In the one-dimensional generalized SABR model, the Ninomiya-Victoir method is only faster than the Euler method because the integrand seems to be smoother. If one rejects our way to determine the necessary number of trajectories M as too crude and insists on taking the same number for both methods, the Ninomiya-Victoir method will require almost the same time as the Euler method in order to give comparable results at this level. However, the Ninomiya-Victoir method with drift still retains a convincing speed-up, again probably due to the simpler structure of the subroutines. Note that in the case of the Euler scheme, one has to compute a 64-dimensional integral, whereas in the case of the Ninomiya-Victoir scheme (with or

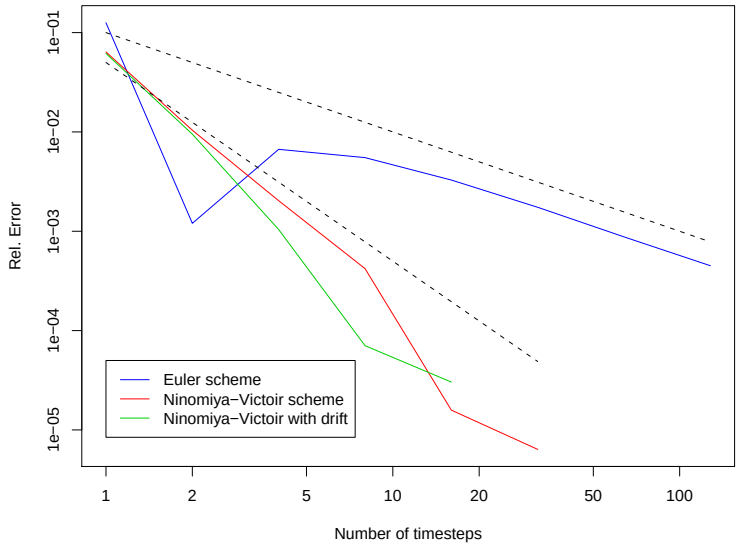

FIGURE 2. Order of convergence for the generalized SABR model.

without drift), the integration only needs to be performed on an eight-dimensional space.

5.3. **Multi-dimensional generalized SABR.** In this section we give the numerical results corresponding to the multi-dimensional generalized SABR model. Recall that from Section 3.4 this model is given by

$$\begin{aligned} \mathrm{d}X_i(t) =& a_i Y_i^{\alpha_i} X_i^{\beta_i} \mathrm{d}\widetilde{B}_t^i \\ \mathrm{d}Y_i(t) =& \kappa_i(\theta_i - Y_i)\mathrm{d}t + b_i Y_i \mathrm{d}\widetilde{W}_t^i, \end{aligned}$$

$i = 1, \dots, N$. For our experiment we choose $N = 4$ and as the derivative we take a basket option with the same weight on each stock. The parameters for the experiment have been chosen as follows: $a = (1, 0.5, 0.3, 0.7)^T$, $b = (0.5, 0.8, 0.4, 0.6)^T$, $\alpha = (0.5, 1, 0.7, 0.8)^T$, $\beta = (0.6, 0.7, 0.8, 0.9)^T$, $\kappa = (0.2, 0.7, 0.5, 0.9)^T$, $\theta = (0.3, 0.4, 0.6, 0.2)^T$, and, finally,

$$\rho \approx \begin{pmatrix} 1 & 0.0111 & 0.6395 & -0.1081 & -0.3414 & -0.0642 & -0.2054 & -0.0236 \\ 0.0111 & 1 & 0.2698 & 0.2770 & 0.1651 & -0.3504 & -0.8186 & -0.4383 \\ 0.6395 & 0.2698 & 1 & -0.1381 & -0.1379 & -0.0031 & -0.3169 & -0.0161 \\ -0.1081 & 0.2770 & -0.1381 & 1 & 0.7312 & -0.9030 & 0.0419 & -0.8121 \\ -0.3414 & 0.1651 & -0.1379 & 0.7312 & 1 & -0.5969 & 0.0747 & -0.6703 \\ -0.6420 & -0.3504 & -0.0031 & -0.9030 & -0.5969 & 1 & 0.1878 & 0.8790 \\ -0.2054 & -0.8186 & -0.3169 & 0.0419 & 0.0747 & 0.1878 & 1 & 0.2796 \\ -0.0236 & -0.4383 & -0.0161 & -0.8121 & -0.6703 & 0.8790 & 0.2796 & 1 \end{pmatrix}.$$

Note that the above choice of ρ implies that $\widetilde{B}^i$ and $\widetilde{W}^i$ are negatively correlated, as usual in equity modeling. Moreover, ρ is positive definite – and in fact, chosen at random among all such correlation matrices. The estimated 'true value' of the basket option is 0.09254183.

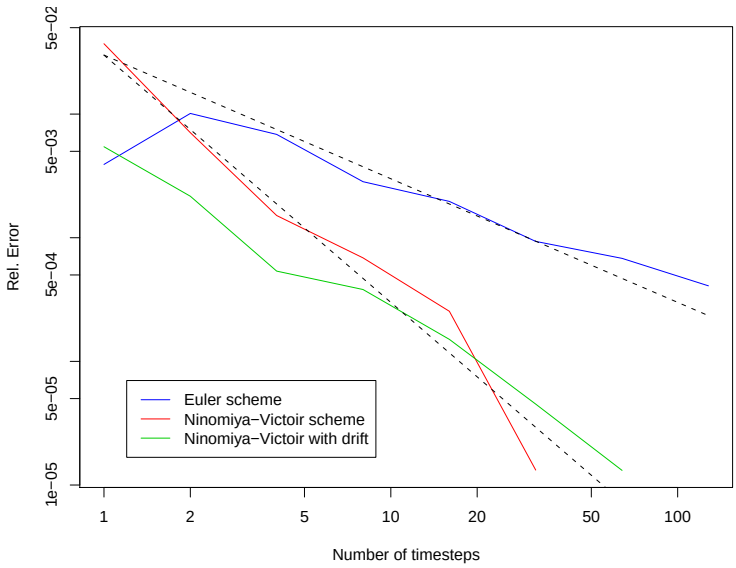

FIGURE 3. Order of convergence for the multi-dimensional generalized SABR model.

The convergence rates of the three different schemes for this experiment are graphically displayed in Figure 3. The picture is similar as in the two previous cases in the sense that there is second-order convergence for the two NV schemes and first-order convergence for the Euler scheme.

Method	K	M	Rel. Error	Time
Euler	32	2048000	0.000934	246.65 sec
Ninomiya-Victoir	4	1024000	0.002017	52.33 sec
NV with drift	4	1024000	0.000862	35.31 sec

TABLE 3. Computational time for the multi-dimensional generalized SABR model

Further, the computational time for the generalized SABR model with a four-dimensional stock market, reported in Table 5.3, again shows the usual picture. The classical Ninomiya-Victoir method gives a speed-up between factors two and four (depending on the trust of the choice of M). In this case, one might, however, note that the error from the classical Ninomiya-Victoir method is more than twice higher the the errors from the competing methods. And again, the simpler structure of the ODEs in the case of a Ninomiya-Victoir method with drift leads to a convincing speed-up as compared to both other methods.

APPENDIX

In Sections 2.1, 2.2 and 3.2, the following ODE appears:

$$\begin{aligned} y'(t) =& h(t)(y(t))^{\beta}, \\ y(0) =& x, \end{aligned} \tag{8}$$

where $0 < \beta < 1$, $x \geq 0$ and $h : [0, \infty) \to \mathbb{R}$ is continuous and either positive valued or negative valued. One can easily check that

$$\Phi_x(t) = \left((1-\beta) \int_0^t h(s) \mathrm{d}s + x^{1-\beta} \right)_+^{1/(1-\beta)}, \quad t \geq 0, \tag{9}$$

with $a_+ := \max\{a, 0\}$, is a solution to (8). We briefly provide some details about uniqueness of the solution. When h takes only negative values, then the right hand side of the ODE (8) is decreasing in the state variable and uniqueness follows for any $x \geq 0$ (see e.g. Example 2.4 on p.286 of [9]). When h takes only positive values and $x > 0$, uniqueness follows by the Picard-Lindelöf theorem. When h takes only positive values and $x = 0$, the solution (9) is not unique; for instance, $y(t) \equiv 0$ forms another solution. For this particular case, we have chosen, throughout the paper, to work with the solution $\Phi_0(t)$, since the flow $\Phi_x(t)$ is (right)-continuous at $x = 0$ for all $t \geq 0$.

References

[1] Aurélien Alfonsi, *High order discretization schemes for the CIR process: Application to affine term structure and Heston models*, Math. Comp. **79** (2010), 209-237.

[2] Christian Bayer and Peter K. Friz, *Cubature on Wiener space: pathwise convergence*, Appl. Math. Optim. To appear.

[3] Peter K. Friz and Nicolas B. Victoir., *Multidimensional stochastic processes as rough paths: theory and applications*, Cambridge Studies in Advanced Mathematics, vol. 120, Cambridge University Press, Cambridge, 2010.

[4] Takehiro Fujiwara, *Sixth order methods of Kusuoka approximation*. Tokyo University preprint, 2006. Available from kyokan.ms.u-tokyo.ac.jp/users/preprint/pdf/2006-7.pdf.

[5] Bin Chen, Cornelis W. Oosterlee, and Hans van der Weide, *Efficient unbiased simulation scheme for the SABR stochastic volatility model* (2012). Preprint.

[6] Leonhard Fichte, *The Scheme of Ninomiya and Victoir: Application to Finance* (2011). Diploma thesis, TU Berlin.

[7] Paul Glasserman, *Monte Carlo methods in financial engineering*, Applications of Mathematics (New York), vol. 53, Springer-Verlag, New York, 2004. Stochastic Modelling and Applied Probability.

[8] D.J. Higham and X. Mao, *Convergence of Monte Carlo simulations involving the mean-reverting square root process*, J. Comp Finance **8** (2005), 35–61.

[9] Ioannis Karatzas and Steven E. Shreve, *Brownian motion and stochastic calculus*, 2nd ed., Graduate Texts in Mathematics, vol. 113, Springer-Verlag, New York, 1991.

[10] Peter E. Kloeden and Eckhard Platen, *Numerical solution of stochastic differential equations*, Applications of Mathematics (New York), vol. 23, Springer-Verlag, Berlin, 1992.

[11] Shigeo Kusuoka, *Approximation of expectation of diffusion processes based on Lie algebra and Malliavin calculus*, Advances in mathematical economics. Vol. 6, 2004, pp. 69–83.

[12] P.-L. Lions and M. Musiela, *Some properties of diffusion processes with singular coefficients*, Commun. Appl. Anal. **10** (2006), no. 1.

[13] ———, *Correlations and bounds for stochastic volatility models*, Ann. Inst. H. Poincaré Anal. Non Linéaire **24** (2007), no. 1, 1–16.

[14] R. Lord, R. Koekkoek, and D. van Dijk, *A comparison of biased simulation schemes for stochastic volatility models*, Quantitative Finance **10** (2010), 177–194.

[15] Terry Lyons and Nicolas Victoir, *Cubature on Wiener space*, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. **460** (2004), no. 2041, 169–198. Stochastic analysis with applications to mathematical finance.

[16] Terry Lyons and Christian Litterer, *High order recombination and an application to cubature on Wiener space*. arXiv:1008.4942v1 [math.PR].

[17] Mariko Ninomiya and Syoiti Ninomiya, *A new higher-order weak approximation scheme for stochastic differential equations and the Runge-Kutta method*, Finance Stoch. **13** (2009), no. 3, 415–443.

[18] Syoiti Ninomiya and Nicolas Victoir, *Weak approximation of stochastic differential equations and application to derivative pricing*, Appl. Math. Finance **15** (2008), no. 1-2.
[19] Kojiro Oshima, Josef Teichmann, and Dejan Veluscek, *A new extrapolation method for weak approximation schemes with applications*. arXiv:0911.4380 [math.PR].

WIAS Berlin (first and second author), TU Berlin (second author), University of Manchester (third author)