Eötvös Loránd University **Corvinus University of Budapest**

Analysing short time asymptotic of stochastic volatility models

MSc Thesis

Written by:
Gergely Bence Szilágyi

Supervisors:
Csaba Kőrössy,
Gábor Molnár-Sáska

Budapest, 2018.

Acknowledgement

I would like to express my gratitude to my supervisors Csaba Kőrössy and Gábor Molnár-Sáska for raising this subject and guiding me on the way with their questions and suggestions.

Special thanks to my friends and collegaue at Morgan Stanley who helped me to remain calm and collected after every mistake I made. Thanks to Ákos Somogyi for his ideas about the simulations and the heuristics.

Many thanks to my family for the encouragement. Finally thanks to Zsófia Gyergyószegi for the everyday support.

Contents

1 Introduction

Since 1973, the Black-Scholes formula has been extensively used by traders in financial markets to price options. However, the original Black-Scholes derivation is based on several unrealistic assumptions which are not satisfied under real market conditions. For example, in the original Black-Scholes framework, assets are assumed to follow log-normal processes (i.e. with a constant volatility). This hypothesis can be relaxed by introducing more elaborate models called local and stochastic volatility models.

On one hand, local volatility models assume that the volatility depends only on the underlying and on the time. The market is still complete and, as shown by Dupire, there is a unique diffusion term, which can be calibrated to the current market of European option prices. On the other hand, stochastic volatility models assume that the volatility itself follows a stochastic process: in this case, the market becomes incomplete as it is not possible to hedge and trade the volatility with a single underlying asset.

For these two types of models (local and stochastic), the resulting Black-Scholes partial differential equation becomes complicated and only a few exact solutions are known. The most commonly used solutions are the Constant Elasticity of Variance model (CEV), and the Heston model which assumes a mean-reverting square-root process for the variance. In all other cases, analytical solutions are not available and singular perturbation techniques have been used to obtain asymptotic expressions for the price of European-style options.

By definition, this implied volatility is the value of the volatility that when put in the Black-Scholes formula, reproduces the market price for a European call option. In [1], the authors discovered that the local volatility models predict the wrong behavior for the smile: when the price of the underlying decreases (increases), local volatility models predict that the smile shifts to higher (lower) prices. This problem can be eliminated with the stochastic volatility models such as the SABR model (depending on 4 parameters: Sigma, Alpha, Beta, Rho) [1]. The SABR model has recently been the focus of much attention as it provides a simple asymptotic smile for European call options, assuming a small volatility.

Lewis in [3] introduced an alternative approach for computing the Asymtotic Smile in the CEV model. This method is also applicable for the SABR model, which is the main focus of this thesis. The gist of the technique is that first we define a metric in a Riemannian space using geodesics, then after some heuristic arguments and a comparison using the probability density function finally yields a solution to the asymptotic smile problem for diffusions.

Recently in [5] the authors published a new methodology about efficiently simulating the SABR dynamics. The new method is an extension of the one time-step Monte Carlo method [14], for pricing European options in the context of the model calibration. A highly efficient method results, with many highly interesting and nontrivial components, like Fourier inversion for the sum of log-normals, stochastic collocation, Gumbel copula, correlation approximation, that are not yet seen in combination within a Monte Carlo simulation. The multiple time-step Monte Carlo method what they proposed is especially useful for long-term options and for exotic options.

As for the structure of the thesis, in Section 2 the necessary theoretical background will be listed. As a part we will include the Eikonal equation and one of its possible numerical solutions the fast marching method.

Section 3 is mainly built upon [3] which includes an interesting approach for solving the asymptotic smile problem of stochastic volatility models, in case of the CEV(p) volatility model. We will use this approach to derive the asymptotic smile for the SABR model in 1 and 2 dimensions. The latter doesn't have a closed form solution so numerical approximations have to be applied.

In Section 4 a recently published method will be introduced regarding the Monte Carlo simulation. In case of the SABR model the brute force Monte Carlo simulation is hugely unefficient in terms of computation cost because of an inversion of a nontrivial distribution's CDF is required on each path. The trick for this is to only invert at some discrete points and than linearly interpolate elsewhere. The method that they presented is an "almost exact" SABR MC simulation, where rather than Taylor based simulation techniques, the probability density of the stochastic differential equation (SDE) under consideration is highly accurately approximated. It contains several interesting components, like the Gumbel copula, a recursion plus Fourier inversion to approximate the CDF of the integrated variance, and efficient interpolation by means of SCMC sampler.

Finally in Section 5 the results of my numerical experiments will be presented, which is then followed with some conclusions in Section 6.

2 Theoretical Basis

In this section we will list the necessary theorems and definitions for understanding the thesis. For a start, we will begin with some basic definitions from stochastic calculus.

2.1 The Basics of Stochastic Calculus

Definition 2.1 (Filtration) *In the theory of stochastic processes, a filtration is an increasing sequence of σ-algebras on a measurable space. That is, given a measurable space (Ω, $\mathcal{F}$), a filtration is a sequence of σ-algebras $\{\mathcal{F}_t\}_{t\geq 0}$ with $\mathcal{F}_t \subseteq \mathcal{F}$ where each t is a non-negative real number and*

$$t_1 \leq t_2 \implies \mathcal{F}_{t_1} \subseteq \mathcal{F}_{t_2}.$$

Filtrations in financial mathematics are used for modelling all the available information in the market as the time goes by. The next defined stopping time has a similar objective.

Definition 2.2 (Stopping time) *A random variable $\tau : \Omega \to I$ is called a stopping time if $\forall t \in I : \{\omega \in \Omega : \tau(\omega) \leq t\} \in \mathcal{F}_t$.*

Definition 2.3 (Stochasic process) *For a given probability space $(\Omega, \mathcal{F}, P)$ and a measurable space (S, Σ), a stochastic process is a collection of S-valued random variables, which can be written as: $\{X(t) : t \in T\}$.*

Remark 2.4 S_t can be e.g. a share price. At a certain point of time t: $S_t(\omega)$ is a random variable. For a fixed ω: $S_t(\omega)$ as t runs through the examined time interval is called a trajectory of the stochastic process.

One of the most important stochastic processes is the Brownian motion or Wiener process.

Definition 2.5 (Wiener process) *The Wiener process W_t is characterised by the following properties:*

1. *$W_0 = 0$ almost sure.*
2. *W has independent increments: $\forall t > 0$, the future increments $W_{t+u} - W_t$, $u \geq 0$, are independent of the past values W_s, $s \leq t$.*
3. *W has Gaussian increments: $W_{t+u} - W_t$ is normally distributed with mean 0 and variance u: $W_{t+u} - W_t \sim \mathcal{N}(0, u)$.*
4. *W has continuous paths: With probability 1, W_t is continuous in t.*

2.2 The Basics of Itô Calculus

Definition 2.6 (Itô integral) *Suppose that B is a Wiener process (Brownian motion) and that H is a right-continuous, adapted and locally bounded process. If $\{\pi_n\}$ is a sequence of partitions of $[0,t]$ with mesh going to zero, then the Itô integral of H with respect to B up to time t is a random variable*

$$\int_0^t H\,dB = \lim_{n\to\infty} \sum_{[t_{i-1},t_i]\in\pi_n} H_{t_{i-1}}(B_{t_i} - B_{t_{i-1}}).$$

Definition 2.7 (Itô process) *An Itô process is defined to be an adapted stochastic process that can be expressed as the sum of an integral with respect to Brownian motion and an integral with respect to time,*

$$X_t = X_0 + \int_0^t \sigma_s\,dB_s + \int_0^t \mu_s\,ds.$$

Here, B is a Brownian motion and it is required that σ is a predictable B-integrable process, and μ is predictable and (Lebesgue) integrable. That is,

$$\int_0^t (\sigma_s^2 + |\mu_s|)\,ds < \infty$$

for each t.

Remark 2.8 The stochastic integral can be extended to such Itô processes,

$$\int_0^t H\,dX = \int_0^t H_s\sigma_s\,dB_s + \int_0^t H_s\mu_s\,ds.$$

This is defined for all locally bounded and predictable integrands. More generally, it is required that $H\sigma$ be B-integrable and $H\mu$ be Lebesgue integrable, so that

$$\int_0^t (H^2\sigma^2 + |H\mu|)ds < \infty.$$

Such predictable processes H are called X-integrable.

Itô's lemma is the version of the chain rule or change of variables formula which applies to the Itô integral. It is one of the most powerful and frequently used theorems in stochastic calculus.

Theorem 2.9 (Itô's lemma) *For a continuous d-dimensional semimartingale $X = (X_1, \ldots, X_d)$ and twice continuously differentiable function f from $\mathbf{R}^d$ to $\mathbf{R}$, it states that $f(X)$ is a semimartingale and,*

$$df(X_t) = \sum_{i=1}^d f_i(X_t)\,dX_t^i + \frac{1}{2}\sum_{i,j=1}^d f_{i,j}(X_t)\,d[X^i, X^j]_t.$$

In probability theory, the Girsanov theorem (named after Igor Vladimirovich Girsanov) describes how the dynamics of stochastic processes change when the original measure is changed to an equivalent probability measure. The theorem is especially important in the theory of financial mathematics as it tells how to convert from the physical measure, which describes the probability that an underlying instrument (such as a share price or interest rate) will take a particular value or values, to the risk-neutral measure which is a very useful tool for pricing derivatives on the underlying instrument.

Theorem 2.10 (Girsanov's theorem) *Let $\{W_t\}$ be a Wiener process on the Wiener probability space $\{\Omega, \mathcal{F}, P\}$. Let X_t be a measurable process adapted to the natural filtration of the Wiener process $\{\mathcal{F}_t^W\}$ with $X_0 = 0$.*
Define the Doléans-Dade exponential $\mathcal{E}(X)_t$ of X with respect to W

$$\mathcal{E}(X)_t = \exp\left(X_t - \frac{1}{2}[X]_t\right).$$

If $\mathcal{E}(X)_t$ is a strictly positive martingale, a probability measure Q can be defined on $\{\Omega, \mathcal{F}\}$ such that we have Radon–Nikodym derivative

$$\frac{dQ}{dP}|_{\mathcal{F}_t} = \mathcal{E}(X)_t.$$

Then for each t the measure Q restricted to the unaugmented sigma fields $\mathcal{F}_t^W$ is equivalent to P restricted to $\mathcal{F}_t^W$. Furthermore, if Y is a local martingale under P, then the process

$$\tilde{Y}_t = Y_t - [Y, X]_t$$

is a Q local martingale on the filtered probability space $\{\Omega, F, Q, \{F_t^W\}\}$.
Moreover if X is a continuous process and W is Brownian motion under measure P then

$$\tilde{W}_t = W_t - [W, X]_t$$

is Brownian motion under Q.

A stochastic differential equation (SDE) is a differential equation in which one or more of the terms is a stochastic process, resulting in a solution which is also a stochastic process. SDEs are used to model various phenomena such as unstable stock prices or physical systems subject to thermal fluctuations. Typically, SDEs contain a variable which represents random white noise calculated as the derivative of Brownian motion or the Wiener process. However, other types of random behaviour are possible, such as jump processes.

Theorem 2.11 (Existence and uniqueness of solutions) *Let $T > 0$, and let $\sigma : \mathbb{R}^n \times [0, T] \to \mathbb{R}^{n\times m}$; be measurable functions for which there exist constants C and D such that*

$$\begin{gathered}\big|\mu(x,t)\big| + \big|\sigma(x,t)\big| \le C\big(1 + |x|\big);\\ \big|\mu(x,t) - \mu(y,t)\big| + \big|\sigma(x,t) - \sigma(y,t)\big| \le D|x - y|;\end{gathered}$$

for all $t \in [0, T]$ and all $x, y \in \mathbb{R}^n$, where $|\sigma|^2 = \sum_{i,j=1}^n |\sigma_{ij}|^2$. Let Z be a random variable that is independent of the σ-algebra generated by B_s, $s \ge 0$, and with finite *second moment:* $\mathbb{E}\big[|Z|^2\big] < +\infty$. *Then the stochastic differential equation/initial value problem*

$$\mathrm{d}X_t = \mu(X_t, t)\,\mathrm{d}t + \sigma(X_t, t)\,\mathrm{d}B_t \text{ for } t \in [0, T];$$
$$X_0 = Z;$$

has a Pr-almost surely unique t-continuous solution $(t, \omega) \to X_t(\omega)$ *such that* X *is adapted to the filtration* $\mathcal{F}_t^Z$ *generated by* Z *and* B_s, $s \geq t$, *and*

$$\mathbb{E}\left[\int_0^T |X_t|^2\,\mathrm{d}t\right] < +\infty.$$

We will need to describe later the time evolution of the probability density function, for which we will use the Fokker-Planck partial differential equation or Kolmogorov forward equation.

Theorem 2.12 (Fokker- Planck in one dimension) *For an Itô process driven by the standard Wiener process* W_t *and described by the stochastic differential equation (SDE)*

$$dX_t = \mu(X_t, t)\,dt + \sigma(X_t, t)\,dW_t$$

with drift $\mu(X_t, t)$ *and diffusion coefficient* $D(X_t, t) = \sigma^2(X_t, t)/2$, *the Fokker–Planck equation for the probability density* $p(x, t)$ *of the random variable* X_t *is*

$$\frac{\partial}{\partial t}p(x,t) = -\frac{\partial}{\partial x}\left[\mu(x,t)p(x,t)\right] + \frac{\partial^2}{\partial x^2}\left[D(x,t)p(x,t)\right].$$

The one dimensional case may be more suggestive, that's why it's mentioned above, but the general case will be needed as follows.

Theorem 2.13 (Fokker- Planck equation) *If*

$$d\mathbf{X}_t = \boldsymbol{\mu}(\mathbf{X}_t, t)\,dt + \boldsymbol{\sigma}(\mathbf{X}_t, t)\,d\mathbf{W}_t,$$

where $\mathbf{X}_t$ *and* $\boldsymbol{\mu}(\mathbf{X}_t, t)$ *are N-dimensional random vectors,* $\boldsymbol{\sigma}(\mathbf{X}_t, t)$ *is an* $N \times M$ *matrix and* $\mathbf{W}_t$ *is an* M*-dimensional standard Wiener process, the probability density* $p(\mathbf{x}, t)$ *for* $X\mathbf{X}_t$ *satisfies the Fokker–Planck equation*

$$\frac{\partial p(\mathbf{x},t)}{\partial t} = -\sum_{i=1}^{N}\frac{\partial}{\partial x_i}\left[\mu_i(\mathbf{x},t)p(\mathbf{x},t)\right] + \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\partial^2}{\partial x_i\,\partial x_j}\left[D_{ij}(\mathbf{x},t)p(\mathbf{x},t)\right],$$

with drift vector $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_N)$ *and diffusion tensor* $\mathbf{D} = \frac{1}{2}\boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathsf{T}}$, *i.e.*

$$D_{ij}(\mathbf{x},t) = \frac{1}{2}\sum_{k=1}^{M}\sigma_{ik}(\mathbf{x},t)\sigma_{jk}(\mathbf{x},t).$$

2.3 Models used in the Thesis

The Black–Scholes–Merton model is a mathematical model of a financial market containing derivative investment instruments. From the partial differential equation in the model, known as the Black–Scholes equation, one can deduce the Black–Scholes formula, which gives a theoretical estimate of the price of European-style options and shows that the option has a unique price regardless of the risk of the security and its expected return (instead replacing the security's expected return with the risk-neutral rate).

The Black–Scholes formula has only one parameter that cannot be directly observed in the market: the average future volatility of the underlying asset, though it can be found from the price of other options. Since the option value (whether put or call) is increasing in this parameter, it can be inverted to produce a "volatility surface" that is then used to calibrate other models, e.g. for OTC derivatives.

Remark 2.14 The Black–Scholes equation is a partial differential equation, which describes the price of the derivative over time. The equation is:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0.$$

The Black–Scholes formula calculates the price of European put and call options. This price is consistent with the Black–Scholes equation as above; this follows since the formula can be obtained by solving the equation for the corresponding terminal and boundary conditions.

Theorem 2.15 (Black-Scholes formula) *The value of a call option for a non-dividend-paying underlying stock in terms of the Black–Scholes parameters is:*

$$\begin{aligned} C(S_t, t) &= N(d_1)S_t - N(d_2)Ke^{-r(T-t)} \\ d_1 &= \tfrac{1}{\sigma\sqrt{T-t}}\left[\ln\left(\tfrac{S_t}{K}\right) + \left(r + \tfrac{\sigma^2}{2}\right)(T-t)\right] \\ d_2 &= d_1 - \sigma\sqrt{T-t} \end{aligned}$$

Definition 2.16 (CEV-model) *The CEV model describes a process which evolves according to the following stochastic differential equation:*

$$dS_t = \mu S_t dt + \sigma S_t^{\gamma} dW_t,$$

in which S is the spot price, t is time, and μ is the drift, σ is the volatility and γ is the elasticity of variance parameter, and W is a Brownian motion.

Definition 2.17 (Heston-model) *The basic Heston model assumes that S_t, the price of the asset, is determined by a stochastic process:*

$$dS_t = \mu S_t\, dt + \sqrt{\nu_t} S_t\, dW_t^S$$

where ν_t, the instantaneous variance, is a CIR process:

$$d\nu_t = \kappa(\theta - \nu_t)\, dt + \xi\sqrt{\nu_t}\, dW_t^{\nu}$$

and W_t^S, W_t^{ν} are Wiener processes with correlation ρ, or equivalently, with covariance ρdt.

Definition 2.18 (SABR-model) *The SABR model describes a single forward F, such as a LIBOR forward rate, a forward swap rate, or a forward stock price. The volatility of the forward F is described by a parameter σ . SABR is a dynamic model in which both F and σ are represented by stochastic state variables whose time evolution is given by the following system of stochastic differential equations:*

$$dF_t = \sigma_t F_t^\beta \, dW_t,$$

$$d\sigma_t = \alpha\sigma_t \, dZ_t,$$

with the prescribed time zero (currently observed) values F_0 and σ_0. Here, W_t and Z_t are two correlated Wiener processes with correlation coefficient $-1 < \rho < 1$:

$$dW_t \, dZ_t = \rho \, dt$$

The constant parameters β, α satisfy the conditions $0 \leq \beta \leq 1, \ \alpha \geq 0$.

Remark 2.19 The above dynamics is a stochastic version of the CEV model with the skewness parameter β : in fact, it reduces to the CEV model if $\alpha = 0$ The parameter α is often referred to as the volvol, and its meaning is that of the lognormal volatility of the volatility parameter σ.

2.4 Numerical methods

The Eikonal equation is a non-linear partial differential equation. It is of the form

$$|\nabla u(x)| = 1/f(x), \ x \in \Omega$$

subject to $u|_{\partial\Omega} = 0$, where Ω is an open set in $\mathbb{R}^n$ with well-behaved boundary, $f(x)$ is a function with positive values, ∇ denotes the gradient and $|\cdot|$ is the Euclidean norm. Here, the right-hand side $f(x)$ is typically supplied as known input. Physically, the solution $u(x)$ is the shortest time needed to travel from the boundary $\partial\Omega$ to x inside Ω, with $f(x)$ being the speed at x.

In the special case when $f = 1$, the solution gives the signed distance from $\partial\Omega$. We will make this assumption later on, but until then, the general case will be used.

One fast computational algorithm to approximate the solution to the Eikonal equation is the fast marching method (FMM). However there are other faster or more efficient methods such as the Bellman-Ford algorithm, the "fast sweeping method" (FSM) or some hybrid methods like the parallelized Heap Cell Method in [8], but in this thesis the FMM will be used, because as Hysing and Turek states in [12], the FMM is the best way for computation when the algorithmic complexity is a factor. Moreover later on a generalized Eikonal equation will be used for which the algorithm need to be altered. I found that this change can be done easier in case of the FMM.

Gremaud and Kuster in [9] studied the time needed for computation for FMM and FSM in various cases on Cartesian grids with obstacles. They conclude that FMM is generally faster than FSM in all but the simplest cases (with no obstacles on the Cartesian grid).

2.4.1 Fast Marching Method

We will discuss the method as described in [10].

First, assume that the domain has been discretized into a mesh. We will refer to meshpoints as nodes. Each node x_i has a corresponding value $U_i = U(x_i) \approx u(x_i)$.

The algorithm works just like Dijkstra's algorithm but differs in how the nodes' values are calculated. In Dijkstra's algorithm, a node's value is calculated using a single one of the neighboring nodes. However, in solving the PDE in $\mathbb{R}^n$, between 1 and n of the neighboring nodes are used.

Nodes are labeled as far (not yet visited), considered (visited and value tentatively assigned), and accepted (visited and value permanently assigned). Below are stated the steps of the algorithm.

1 Assign every node x_i the value of $U_i = +\infty$ and label them as far; for all nodes $x_i \in \partial\Omega$ set $U_i = 0$ and label x_i as accepted.

2 For every far node x_i, use the *Eikonal update formula* to calculate a new value for $\tilde{U}$. If $\tilde{U} < U_i$ then set $U_i = \tilde{U}$ and label x_i as considered.

3 Let $\tilde{x}$ be the considered node with the smallest value U. Label $\tilde{x}$ as accepted.

4 For every neighbor x_i of $\tilde{x}$ that is not-accepted, calculate a tentative value $\tilde{U}$.

5 If $\tilde{U} < U_i$ then set $U_i = \tilde{U}$. If x_i was labeled as far, update the label to considered.

6 If there exists a considered node, return to step 3. Otherwise, terminate.

The Eikonal update formula mentioned in step 2 is the following. A first-order accurate discretization of the Eikonal equation is obtained by using upwind finite-differences to approximate partial derivatives:

$$\max\left(D_{ij}^{-x}U, -D_{ij}^{+x}U, 0\right)^2 + \max\left(D_{ij}^{-y}U, -D_{ij}^{+y}U, 0\right)^2 = \frac{1}{f_{ij}^2},$$

where

$$u_x(x_{ij}) \approx D_{ij}^{\pm x}U = \frac{U_{i\pm1,j} - U_{ij}}{\pm h} \quad \text{and} \quad u_y(x_{ij}) \approx D_{ij}^{\pm y}U = \frac{U_{i,j\pm1} - U_{ij}}{\pm h}.$$

Due to the consistent, monotone, and causal properties of this discretization it is easy to show that if $U_H = \min(U_{i-1,j}, U_{i+1,j})$ and $U_V = \min(U_{i,j-1}, U_{i,j+1})$ and $|U_H - U_V| \leq h/f_{ij}$ then

$$\left(\frac{U_{ij} - U_H}{h}\right)^2 + \left(\frac{U_{ij} - U_V}{h}\right)^2 = \frac{1}{f_{ij}^2},$$

which means

$$U_{ij} = \frac{U_H + U_V}{2} + \frac{1}{2}\sqrt{(U_H + U_V)^2 - 2(U_H^2 + U_V^2 - \frac{h^2}{f_{ij}^2})}.$$

This can be simplified into:

$$U_{ij} = \frac{U_H + U_V}{2} + \frac{1}{2}\sqrt{\frac{2h^2}{f_{ij}^2} - (U_H - U_V)^2}.$$

This solution will always exist as long as $|U_H - U_V| \leq \sqrt{2}h/f_{ij}$ is satisfied and is larger than both, U_H and U_V, as long as $|U_H - U_V| \leq h/f_{ij}$. If $|U_H - U_V| \geq h/f_{ij}$, a lower-dimensional update must be performed by assuming one of the partial derivatives is 0:

$$U_{ij} = \min(U_H, U_V) + \frac{h}{f_{ij}}.$$

2.4.2 Runge-Kutta methods

In numerical analysis, the Runge–Kutta methods are a family of implicit and explicit iterative methods, which include the well-known routine called the Euler Method, used in temporal discretization for the approximate solutions of ordinary differential equations. These methods were developed around 1900 by the German mathematicians C. Runge and M. W. Kutta.

The most widely known member of the Runge–Kutta family is generally referred to as "RK4", "classical Runge–Kutta method" or simply as "the Runge–Kutta method".

Let an initial value problem be specified as follows:

$$\dot{y} = f(t, y), \quad y(t_0) = y_0.$$

Here y is an unknown function (scalar or vector) of time t, which we would like to approximate; we are told that $\dot{y}$, the rate at which y changes, is a function of t and of y itself. At the initial time t_0 the corresponding y value is y_0. The function f and the data t_0, y_0 are given.

Now pick a step-size $h > 0$ and define

$$\begin{aligned} y_{n+1} &= y_n + \tfrac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right), \\ t_{n+1} &= t_n + h \end{aligned}$$

for $n = 0, 1, 2, 3, \ldots$:

$$\begin{aligned} k_1 &= h\ f(t_n, y_n), \\ k_2 &= h\ f\left(t_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right), \\ k_3 &= h\ f\left(t_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right), \\ k_4 &= h\ f\left(t_n + h, y_n + k_3\right). \end{aligned}$$

The classical Runge-Kutta is a highly useful method for solving ordinary differential equations for its fast speed and the total accumulated error is on the order of $O(h^4)$.

3 Implied Volatility - The Eikonal Approach

In financial mathematics, the implied volatility of an option contract is that value of the volatility of the underlying instrument which, when input in an option pricing model (such as Black–Scholes) will return a theoretical value equal to the current market price of the option. A non-option financial instrument that has embedded optionality, such as an interest rate cap, can also have an implied volatility. Implied volatility, a forward-looking and subjective measure, differs from historical volatility because the latter is calculated from known past returns of a security.

In general, it is not possible to give a closed form formula for implied volatility in terms of call price. However, in some cases (large strike, low strike, short expiry, large expiry) it is possible to give an asymptotic expansion of implied volatility in terms of call price.

We can interpret the Black-Scholes formula as a function of the stock price, strike, maturity and the volatility: $C_{BS}(S, K, T, \sigma)$. To match the market price of an option, the implied volatility is needed so that: $C_{market} = C_{BS}(S, K, T, \sigma_{implied})$.

For a state-dependent model, an extra parameter is needed:

$$C(S, K, T, \theta) = C_{BS}(S, K, T, \sigma_{implied}),$$

which assumes that $\sigma_{implied} = f(S, K, T, \theta)$.

3.1 Geometries and Smile Asymptotics

The following method was introduced by Alan L. Lewis in [2], and it is the main focus of my thesis. His approach is based o the CEV(p)-vol model, which is:

$$dS_t = rS(t)dt + \sqrt{V(t)}S(t)dW_1(t),$$

$$dV(t) = b(V(t))dt + \xi V(t)^p \left(\rho dW_1(t) + \sqrt{1-\rho^2}dW_2(t)\right),$$

where S is the stock price, t is time, r is the risk free rate, $\sigma = \sqrt{V}$ is the volatility, and W_1 and W_2 are independent Brownian motions.
After applied Itô's lemma to the stock price we get:

$$d(\log S(t)) = \left(r - \frac{\sigma(t)^2}{2}\right)dt + \sigma(t)dW_1(t).$$

As for this class of models the stock price is level independent or translation invatiant, we can get the implied volatility as:

$$\sigma_{implied} = f(T, x, y),$$

where $x = log(S/K)$ and $y = V$.

In general, the implied volatility has to be numerically computed. But it can be written in a formal power series:

$$\sigma_{implied} = \sum_{i=0}^{\infty} f^{(i)}(x,y) \cdot T^i.$$

Our main task is to compute the leading $T \to 0$ behaviour:

$$\sigma_{imp} := f^{(0)}(x,y) = \lim_{T\to 0} \sigma_{implied}(T,x,y).$$

The call option price is determined by

$$C(T,S_0,V_0;K) = \mathbb{E}_{(S_0,V_0)}\left((S_T-K)^+\right) = e^{-rT}\int_0^{\infty} \max(0;S_T-K)q(T,S_0,V_0;S_T)dS_T,$$

where the probability transition density

$$q(T,S_0,V_0;S_T)dS_T = P_{(S_0,V_0)}\left(S_T \in dS_T\right)$$

reflects arriving at the terminal stock price with any volatility. This is distinguished from the 'complete' transition density:

$$p(T,S_0,V_0;S_T,V_T)dS_T dV_T = P_{(S_0,V_0)}\left(S_T \in dS_T, V_T \in dV_T\right).$$

Let's denote the 'state variables' by $\overline{x}_t = (S_t,V_t)$. By the Markov property, for any time sub-division $T = n\Delta t$,

$$q(T,S_0,V_0;S_T) = \int p(\Delta t,\overline{x}_0;\overline{x}_{t_1})p(\Delta t,\overline{x}_{t_1};\overline{x}_{t_2})\cdots p(\Delta t,\overline{x}_{t_{n-1}};\overline{x}_{t_n})d\overline{x}_{t_1}\cdots d\overline{x}_{t_{n-1}}dV_T. \tag{1}$$

At this point, we can generalize the problem a bit. Suppose that $\overline{x}_t$ is a D-dimensional diffusion process. It means, that we can use a model with more than one stock price or volatility.

So with our previous notations $p(\Delta t,\overline{x};\overline{y})$ is the transition density for a $2D$-dimensional diffusion process with drift $\overline{b}_t = \overline{b}(\overline{x}_t)$ and variance-covariance matrix

$$a_t = [a_{ij}(\overline{x}_t)], \quad (i,j = 1,\ldots D).$$

Let n be fixed. So we should replace $T \to 0$ with $\Delta t \to 0$. For small enough Δt, the transition densities must be approximately D-dimensional Gaussian:

$$p(\Delta t,\overline{x};\overline{y}) \approx \frac{1}{(2\pi)^{D/2}(\det a)^{1/2}} \cdot \exp\left(-\frac{1}{2\Delta t}\left(\overline{y}-\overline{x}-\overline{b}\Delta t\right)^T a^{-1}\left(\overline{y}-\overline{x}-\overline{b}\Delta t\right)\right). \tag{2}$$

To leading order, the drifts $\overline{b}\Delta t$ don't contribute. The first fraction in the equation is the normalizing constant, which can be left out from the following approximation. After comparing 1 and 2 we conclude:

$$q(T,S_0,V_0;S_T)\approx$$

$$\int \exp\left(-\frac{1}{2\Delta t}\sum_{i=1}^{n}\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)^T a^{-1}(\overline{x}_{t_{i-1}})\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)\right)d\overline{x}_{t_1}\cdots d\overline{x}_{t_{n-1}}dV_T.$$

In the limit, the points $\{x_{t_i}\}\to\{x_t\}$ create a continuous path for any diffusion. This is done by compressing the subdivision $(n\to\infty)$. The integrand is a maximum along the paths $\{x_t\}$ which minimize the sum and becomes concentrated there. We list a couple of ideas, that can explain the above statement.

Remark 3.1 Saddle point is a point on the surface of the graph of a function where the slopes (derivatives) of orthogonal function components defining the surface become zero (a stationary point) but are not a local extremum on both axes (for example a hyperbolic paraboloid).
Steepest descent or gradient descent is a first-order iterative optimization algorithm for finding the minimum of a function. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient of the function at the current point.
WKB approximation is a method for finding approximate solutions to linear differential equations with spatially varying coefficients. The name is an initialism for Wentzel–Kramers–Brillouin.

Example 3.2 We want to calculate the maximum of an integral: $\int_{-\infty}^{\infty}e^{-f(x)}dx$, where x is a vector. The idea behind computing is that after a certain level of f, the integrand is basically zero. And not just that, also the integral is less than ϵ in those part of the parameterspace, where f is above a certain limit. So we can concentrate on the minimum of f when evaluating the integral.

Interpret $g(x)=a^{-1}(x)=[g_{ij}(x)]$ as a metric tensor. This step is only needed because of the change of aspect we are going to carry out.

Remark 3.3 A metric tensor is a type of function which takes as input a pair of tangent vectors v and w at a point of a surface (or higher dimensional differentiable manifold) and produces a real number scalar $g(v,w)$ in a way that generalizes many of the familiar properties of the dot product of vectors in Euclidean space.

In the following we will use the Einstein summation convention, which means when an index variable appears twice in a single term and is not otherwise defined, it implies summation of that term over all the values of the index. This notation will be used later on without further notice. So the sum that has to be minimalized takes the form:

$$\frac{1}{2\Delta t}\sum_{i=1}^{n}\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)^T a^{-1}(\overline{x}_{t_{i-1}})\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)=$$

$$\frac{\Delta t}{2}\sum_{i=1}^{n}[g(\overline{x}_{t_{i-1}})]_{jk}\frac{\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)^j}{\Delta t}\frac{\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)^k}{\Delta t},$$

where the lower indeces mean the coordinate in the matrix and the upper indeces are vector coordinates. Let $\Delta s = \frac{\Delta t}{T}$, which is equivalent with $\Delta s = \frac{1}{n}$. As $\Delta t \to 0$ the sum approximates the following integral:

$$\frac{1}{2T}\sum_{i=1}^{n}\left([g(\overline{x}_{t_{i-1}})]_{jk}\frac{\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)^j}{\Delta s}\frac{\left(\overline{x}_{t_i}-\overline{x}_{t_{i-1}}\right)^k}{\Delta s}\Delta s\right) \to \frac{1}{2T}\int_0^1 g_{jk}(x(s))\dot{x}^j(s)\dot{x}^k(s)ds.$$

Here, $\dot{x}^j(s)$ denotes the jth coortinate of $\frac{dx}{ds}$. So we can state that as $n \to \infty$ the sum converges to the integral. Wrapping it all together we can state the following lemma:

Lemma 3.4 *As $T \to 0$ the probability transition density is approximately the following:*

$$q(T, S_0, V_0; S_T) \approx \exp\left\{-\frac{1}{2T}\min_{\substack{x(0)\,=\,(S(0),\,V(0)),\\ x(1)\,=\,(S(T),\,\cdot)}}\left(\int_0^1 g_{jk}(x(s))\dot{x}^j(s)\dot{x}^k(s)ds\right)\right\}.$$

The above described approach was the way of the probability theory (large deviation principle). In the following we turn our interest to geometry and more explicitly the geodesics. The next theorem was proved by Varadhan in [7].

Theorem 3.5 *For a D dimensional diffusion X with some set A in the metric space with $x \notin A$ as $T \to 0$*

$$\mathbb{P}_x\left(X_T \in A\right) \approx \exp\left\{-\frac{d^2(x, A)}{2T}\right\},$$

where

$$d^2(x, A) = \min_{\substack{\gamma(0)\,=\,x,\\ \gamma(1)\,\in\,A}}\left(\int_0^1 g_{jk}(\gamma(s))\dot{\gamma}^j(s)\dot{\gamma}^k(s)ds\right).$$

Remark 3.6 The minimizing paths are geodesics in a Riemannian space.

Recall the price of a call option:

$$C(T, S_0, V_0; K) = e^{-rT}\int_0^\infty \max(0; S_T - K)q(T, S_0, V_0; S_T)dS_T.$$

Since $\frac{\partial^2}{\partial K^2}\max(0; S_T - K) = \delta(S_T - K)$ (Dirac-delta), we have

$$\frac{\partial^2}{\partial K^2}C(T, S_0, V_0; K) = e^{-rT}q(T, S_0, V_0; K) \approx \exp\left\{-\frac{d^2(x_0, y_0; A_k)}{2T}\right\}, \qquad (3)$$

where $x_0 = \log S_0$, $y_0 = V_0$ and $d^2(x_0, y_0; A_k)$ is the geodesic distance to the set $A_k := \{x = k := \log K\}$ (a line) in the state space (x, y).

We will compare this formula with the Black-Scholes model. Deriving the price of a call option by the strike:

$$\frac{\partial C_{BS}}{\partial K} = S \cdot e^{-\frac{d_1^2}{2}} \cdot \frac{-1}{K\sigma\sqrt{T}} - e^{-rT}N(d_2) - K \cdot e^{-rT} \cdot e^{-\frac{d_2}{2}} \cdot \frac{-1}{K\sigma\sqrt{T}} = -e^{-rT}N(d_2).$$

The second equation comes from the identity

$$e^{-rT} \cdot e^{-\frac{d_2}{2}} = e^{-\frac{d_1^2}{2}} \cdot \frac{S}{K}.$$

The second derivative is:

$$\frac{\partial^2 C_{BS}}{\partial K^2} = e^{-rT} \cdot e^{-\frac{d_2}{2}} \cdot \frac{1}{K\sigma\sqrt{T}} = \frac{1}{K\sigma\sqrt{T}} \cdot \exp\left\{ -rT - \frac{\log^2(\frac{S}{Ke^{-rT}})}{2\sigma^2 T} + \frac{\log(\frac{S}{Ke^{-rT}})}{2} - \frac{\sigma^2 T}{8} \right\}$$

Let $x := \frac{S}{K}$. Ordering the terms respect to T:

$$\frac{\partial^2 C_{BS}}{\partial K^2} = \frac{1}{K\sigma\sqrt{T}} \cdot \exp\left\{ - T\left(\frac{r}{2} + \frac{\sigma^2}{8} + \frac{r^2}{2\sigma^2}\right) + \log x \left(\frac{1}{2} - \frac{r}{\sigma^2}\right) - \frac{\log^2 x}{2\sigma^2 T} \right\},$$

where as $T \to 0$ the driver of the convergence is the last term, so in the limit we can state, that:

$$\frac{\partial^2 C_{BS}}{\partial K^2} \approx \exp\left\{ - \frac{\log^2 x}{2\sigma_0^2 T} \right\}.$$

It is more convenient for us to analyze the logarithm of the share price rather than the price itself. For this we will use $x_0 := \log S$ and $k := \log K$.

$$\frac{\partial^2 C_{BS}}{\partial K^2} \approx \exp\left\{ - \frac{(x_0 - k)^2}{2\sigma_0^2 T} \right\}.$$

For general stochastic volatility models this formula takes the following shape:

$$\frac{\partial^2 C}{\partial K^2} \approx \exp\left\{ - \frac{(x_0 - k)^2}{2\sigma_{imp}^2(x_0 - k, y_0)T} \right\}, \tag{4}$$

where y_0 is the volatility at $t = 0$. Returning now to (3) we can realise the translation invariance in the x coordinate:

$$d^2(x_0, y_0; A_k) = d^2(x_0 - k, y_0; A_0),$$

which means that an $(x_0, \cdot)$ point has the same distance from the $\{x = k\}$ line that an $(x_0 - k, \cdot)$ point has from the $\{x = 0\}$ line. Comparing this to (4) we get our main theorem for asymptotic smiles:

$$\sigma_{imp}^2(x, y) = \frac{x^2}{d^2(x, y)}, \tag{5}$$

where $d(x, y)$ is the minimum geodesic distance from (x, y) to the y-axis. Now x and y are scalar coordinates (recall: the financial variables are $x = log(S_0/K)$ and $y = V_0$). We have suppressed the dependence on the target set A.

3.1.1 Effective local volatility

Given the metric g, and the starting point $P_0 = (x, y)$, one possible solution is to compute all the geodesics that pass through P_0. One of these geodesics will be the distance minimizer to the target. It hits the target at some optimal y_1^*.

It can be shown, that in the limit this y_1^* equals to the effective local volatility, which we can get from Dupire's equation.

Theorem 3.7 (Dupire's equation) *In local volatility models the price of a European call option $C(T, S, K, V)$ solves exactly for all T*

$$\frac{\partial C}{\partial T} = \frac{1}{2}\alpha(T, S, K, V)K^2\frac{\partial^2 C}{\partial K^2} - rK\frac{\partial C}{\partial K},$$

where $\alpha(T, S, K, V)$ is the effective local volatility.

In [11] it is proved that

$$\frac{\partial C}{\partial T} = \mathbb{E}_{(S_0,V_0)}\left(V_T | S_T = K\right) \cdot \frac{1}{2}K^2\frac{\partial^2 C}{\partial K^2}.$$

Comparing these two we get

$$\alpha(T, S, K, V) = \mathbb{E}_{(S_0,V_0)}\left(V_T | S_T = K\right).$$

One can see intuitively that the second component equals to y_1^*. Because it is the expected value of the volatility when we reach the target set at maturity from the starting point P_0.

3.2 The Eikonal Approach

There are multiple possible approaches we could follow. Lewis educes a method using purely differentialgeometry. The gist of the approach is computing all the geodesics via the Christoffel symbols, use the well known conditions of the motion and than find the optimal parameters.

Another method includes the characteristic functions. We should find d after rescaling the characteristic function and the apply a Legendre-transformation or saddle point method. The drawback of this approach is that is not applicable to the general case, as for the CEV(p) models, the characteristic functions are only known for half integers.

Our approach to computation will be solving a generalized Eikonal problem. Let's denote $\partial_i d = \partial d / \partial x^i$, where $x^1 = x$ and $x^2 = y$. The equation to be solved is $a_{ij} \cdot \partial_i d \cdot \partial_j d = 1$ with boundary condition $d(0, y) = 0$. Here $[[a_{ij}]]$ is the variance-covariance matrix and d is the minimum geodesic distance from the y axis defined in the previous subsection. The equation written into matrix-form:

$$\begin{pmatrix} d_x & d_y \end{pmatrix} \cdot \begin{pmatrix} y & \rho y^{p+1/2} \\ \rho y^{p+1/2} & y^{2p} \end{pmatrix} \cdot \begin{pmatrix} d_x \\ d_y \end{pmatrix} = 1,$$

which is equivalent with

$$y d_x^2 + 2\rho y^{p+1/2} d_x d_y + y^{2p} d_y^2 = 1.$$

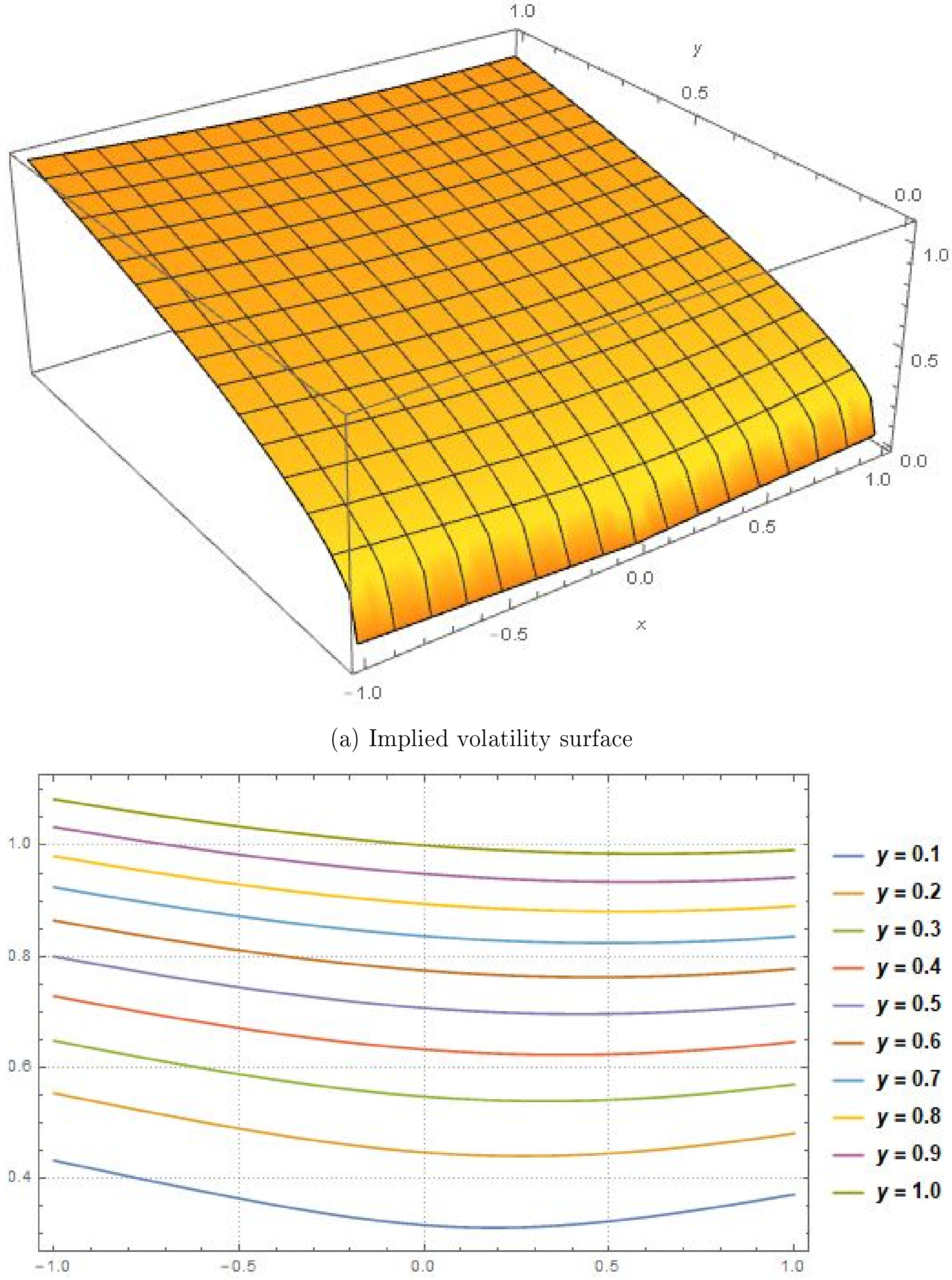
We will later prove that this is equivalent with our original problem when the SABR model's going to be in scope. The trick to solve this equation is to note that there is a scaling form solution:

$$d(x, y) = y^{1-p} F(z), \text{ where } z = x y^{p-3/2}.$$

This yields, using $\alpha = p - 3/2$, to the first-order nonlinear ODE:

$$(1 + 2\rho\alpha z + \alpha^2 z^2)(F'(z))^2 + 2(1-p)(\rho + \alpha z) F(z) F'(z) + (1-p)^2 (F(z))^2 = 1.$$

This equation can be solved numerically in Mathematica. The following graphics have been made with the parameter settings: $\rho = 0.2$, $p = 1$.



(a) Implied volatility surface

(b) Meshes from imp. vol. surface

Figure 2. Implied volatility surface of the CEV(p) model with $\rho = 0.2$ and $p = 1$

3.3 Applied to the SABR model

Consider now the SABR model with $\beta = 1$. The dinamics of the logarithm of the forward and its volatility are the following:

$$dx_t = -\frac{y_t^2}{2}\,dt + y_t\,dW_t,$$

$$dy_t = \alpha y_t\,dZ_t,$$

where

$$dW_t dZ_t = \rho dt.$$

So the variance-covariance matrix is the following:

$$a(x,y) = \begin{pmatrix} y^2 & \alpha\rho y^2 \\ \alpha\rho y^2 & \alpha^2 y^2 \end{pmatrix}.$$

In section 3.1 we've intuitively shown the following lemma, for detailed proof see Varadhan [6].

Lemma 3.8 *In short time limit t the probability density function can be written in the following form:*

$$p(x,y;t) = \frac{c}{\sqrt{t}} \exp\left(-\frac{d^2(x,y)}{2t}\right).$$

Remark 3.9 This lemma is also the special case of Theorem 3.1 in Labordere's paper [2].

We are going to prove that this minimum geodesic distance function d satisfies a generalized Eikonal equation.

Theorem 3.10 *Let $d(x,y)$ be a function as described before and $a(x,y)$ is the variance-covariance matrix as above. Then the following equality holds:*

$$\begin{pmatrix} d_x & d_y \end{pmatrix} \cdot a(x,y) \cdot \begin{pmatrix} d_x \\ d_y \end{pmatrix} = 1.$$

Proof. Let's write the dinamics into matrix-form.

$$d\begin{pmatrix} x_t \\ y_t \end{pmatrix} = \begin{pmatrix} -\frac{y^2}{2} \\ 0 \end{pmatrix} dt + \begin{pmatrix} y & 0 \\ 0 & \alpha y \end{pmatrix} d\begin{pmatrix} w_t \\ z_t \end{pmatrix}.$$

We want to use the Fokker-Planck equation. For this we have to transform the Wiener processes into independent ones: $w_t^1 = w_t$, $w_t^2 = \rho w_t + \sqrt{1-\rho^2} z_t$. So the transormed equation has the form:

$$d\begin{pmatrix} x_t \\ y_t \end{pmatrix} = \begin{pmatrix} -\frac{y^2}{2} \\ 0 \end{pmatrix} dt + \begin{pmatrix} y & 0 \\ \rho\alpha y & \alpha\sqrt{1-\rho^2} y \end{pmatrix} d\begin{pmatrix} w_t^1 \\ w_t^2 \end{pmatrix}.$$

Now apply the 2-dimensional Fokker-Planck equation:

$$\frac{\partial p}{\partial t}+\frac{\partial}{\partial x}\left(-\frac{y^2}{2}\cdot p\right)+\frac{\partial}{\partial y}(0\cdot p)-\frac{1}{2}\frac{\partial^2}{\partial x^2}\left(y^2\cdot p\right)-\frac{1}{2}\frac{\partial^2}{\partial y^2}(\alpha^2 y^2\cdot p)-\frac{\partial^2}{\partial x\partial y}\left(\alpha\rho y^2\cdot p\right)=0.$$

After some reduction and usage of the common notation for the partial derivatives this yields to:

$$p_t-\left(\frac{y^2}{2}+2\rho\alpha y\right)p_x-2\alpha^2 y\cdot p_y-\frac{y^2}{2}p_{xx}-\rho\alpha y^2\cdot p_{xy}-\frac{\alpha^2 y^2}{2}p_{yy}=\alpha^2 p.$$

Now we apply our lemma, the partial derivatives are:

$$p_t=p\left(-\frac{1}{2t}-\frac{d\cdot d_t}{t}+\frac{d^2}{2t^2}\right),$$

$$p_x=p\left(-\frac{d\cdot d_x}{t}\right),$$

$$p_y=p\left(-\frac{d\cdot d_y}{t}\right),$$

$$p_{xx}=p\left(\left(\frac{d\cdot d_x}{t}\right)^2-\frac{d_x^2+d\cdot d_{xx}}{t}\right),$$

$$p_{yy}=p\left(\left(\frac{d\cdot d_y}{t}\right)^2-\frac{d_y^2+d\cdot d_{yy}}{t}\right),$$

$$p_{xy}=p\left(\frac{d^2\cdot d_x\cdot d_y}{t^2}-\frac{d_x\cdot d_y+d\cdot d_{xy}}{t}\right).$$

Writing these into the equation, then multiplying by t^2 and taking the limit $t\to 0$ we get:

$$p\left(\frac{d^2}{2}-\frac{y^2d^2d_x^2}{2}-\frac{\alpha^2y^2d^2d_y^2}{2}-\rho\alpha y^2d^2d_xd_y\right)=0.$$

Since p is positive everywhere and d is also positive apart from the boundary (the y axis) we get back our equation:

$$y^2d_x^2+2\rho\alpha y^2d_xd_y+\alpha^2y^2d_y^2=1.$$

□

Remark 3.11 In the end of the proof we acknowledged that it is an important condition for the equation that d cannot be 0. This is why the equation is usually defined in an open set with well-behaved boundary.

3.3.1 Solution of the equation in 1 dimension

To solve this equation we will apply a dimension reduction as in the case of the CEV-model. Define

$$z = \alpha\frac{x}{y} \qquad \text{and} \qquad F(z) = d(x,y).$$

For the partial derivatives of d

$$d_x(x,y) = \frac{\alpha}{y}\cdot F'(z) \qquad \text{and} \qquad d_y(x,y) = -\frac{\alpha x}{y^2}\cdot F'(z)$$

stands. Putting this into the original equation we get:

$$(F'(z))^2(1-2\rho z+z^2) = \frac{1}{\alpha^2} \Rightarrow F'(z) = \frac{1}{\alpha\sqrt{1-2\rho z+z^2}}.$$

The solution is the following:

$$F(z) = \frac{1}{\alpha}\cdot\log\left(\sqrt{1-2\rho z+z^2}-\rho+z\right)+c.$$

The boundary condition of the original case was $d(0,y)=0$ this implies $F(0)=0$. So the constant is

$$0 = \frac{\log(1-\rho)}{\alpha}+c \quad \Rightarrow \quad c = \frac{\log\left(\frac{1}{1-\rho}\right)}{\alpha}.$$

Hence

$$F(z) = \frac{1}{\alpha}\cdot\log\left(\frac{\sqrt{1-2\rho z+z^2}-\rho+z}{1-\rho}\right).$$

Finally we can get the implied volatility surface from the following formula:

$$\sigma_{imp}(x,y) = \frac{x}{F(\alpha\frac{x}{y})} = \frac{\alpha x}{\log\left(\frac{\sqrt{1-2\rho\alpha\frac{x}{y}+(\alpha\frac{x}{y})^2}-\rho+\alpha\frac{x}{y}}{1-\rho}\right)}.$$

See Figure 3 for a volatility surface of the parameter settings $\alpha = 0.4$, $\rho = 0.1$.

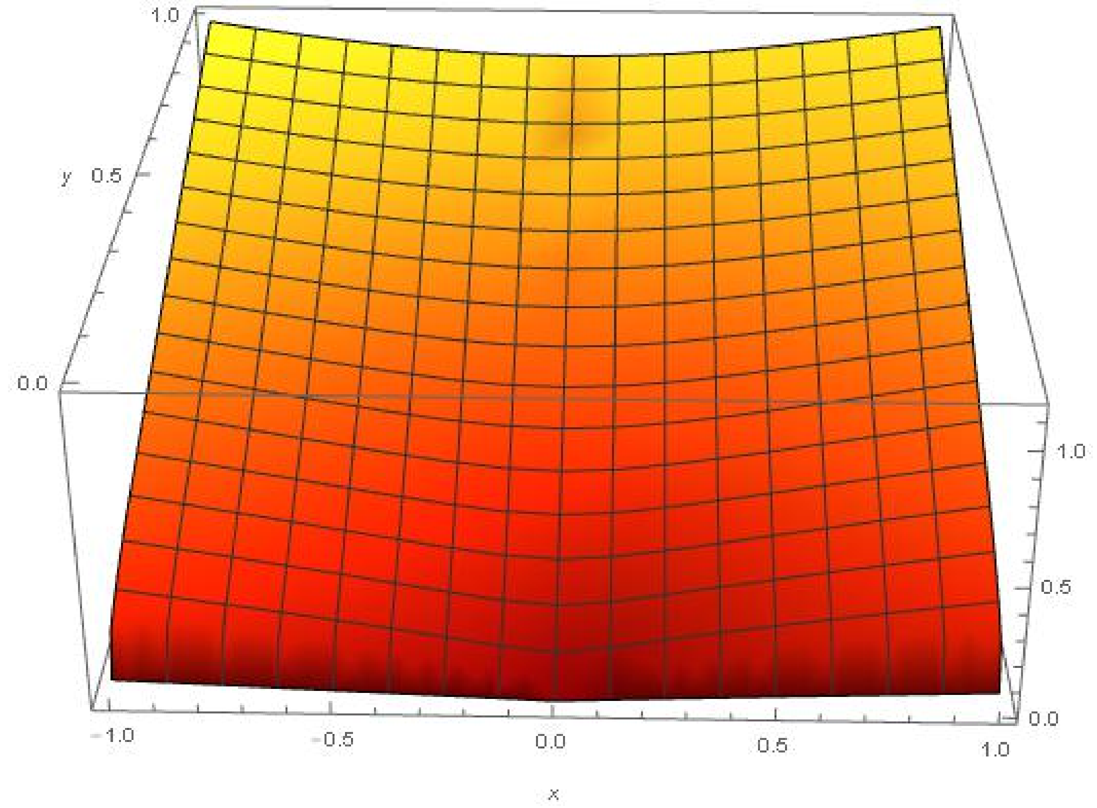


Figure 3. Implied volatility surface from the SABR model with $\alpha = 0.4$ and $\rho = 0.1$

In Section 5 we will confront this formula with the Runge-Kutta method and with Monte Carlo simulation. The idea will be to fix y (the starting volatility) and compute the implied volatility for various x values.

But what if we fix $x = 0$ and looking for the implied vols? Intuitively it's easy, it must be y. But in the formula $\frac{0}{0}$ is resulted. Using the L'Hospital's rule at $x = 0$:

$$\lim_{x\to 0}\left(\frac{\alpha x}{\log\left(\frac{\sqrt{1-2\rho\alpha\frac{x}{y}+(\alpha\frac{x}{y})^2}-\rho+\alpha\frac{x}{y}}{1-\rho}\right)}\right)=\lim_{x\to 0}\left(\frac{\alpha}{\frac{\frac{\alpha}{y}+\frac{-2\rho\alpha\frac{1}{y}+2\alpha^2\frac{x}{y^2}}{2\sqrt{1-2\rho\alpha\frac{x}{y}+(\alpha\frac{x}{y})^2}}}{\sqrt{1-2\rho\alpha\frac{x}{y}+(\alpha\frac{x}{y})^2}-\rho+\alpha\frac{x}{y}}}\right)=\frac{\alpha}{\frac{\alpha(1-\rho)\frac{1}{y}}{1-\rho}}=y.$$

3.3.2 2 dimensional case

Mercurio and Moreni in [13] published a way of using multiple SABR processses when modelling forward inflation rates. This is the theoretical background for this section.
Let us have two SABR processes with their own volatilities:

$$\frac{df_1}{f_1} = \sigma_1 dw_1, \qquad \frac{df_2}{f_2} = \sigma_2 dw_2,$$

$$\frac{d\sigma_1}{\sigma_1} = \nu_1 dz_1, \qquad \frac{d\sigma_2}{\sigma_2} = \nu_2 dz_2,$$

and their correlations

$$\langle dw_1, dw_2 \rangle = \rho dt; \qquad \left\langle dw_i, dz_j \right\rangle = \rho_{ij} dt; \qquad \langle dz_1, dz_2 \rangle = \kappa dt.$$

Consider now $f = f_1 \cdot f_2$. The idea behind this if we can construct the implied volatility of the product of two inflation rates, than it can be constructed via induction for the product of n inflation rates. The dinamic of f using Itô's lemma:

$$\frac{df}{f} = \sigma_1 dw_1 + \sigma_2 dw_2 + \frac{1}{2}\sigma_1\sigma_2 \langle dw_1, dw_2 \rangle = \sigma_1 dw_1 + \sigma_2 dw_2 + \frac{1}{2}\sigma_1\sigma_2 \rho dt$$

Using Girsanov's theorem, we are zeroing out the drift, so in an equivalent martingale-measure (we refer to it as lognormal measure)

$$\frac{df}{f} = \sigma_1 dw_1 + \sigma_2 dw_2.$$

Define $x_1 = \ln(f)$, $x_2 = \sigma_1$ and $x_3 = \sigma_2$. The variance-covariance matrix of these variables is

$$[[a_{ij}]] = \begin{pmatrix} x_2^2 + x_3^2 + 2\rho x_2 x_3 & x_2^2 \nu_1 \rho_{11} + x_2 x_3 \nu_1 \rho_{21} & x_3^2 \nu_2 \rho_{22} + x_2 x_3 \nu_2 \rho_{12} \\ x_2^2 \nu_1 \rho_{11} + x_2 x_3 \nu_1 \rho_{21} & x_2^2 \nu_1^2 & x_2 x_3 \kappa \nu_1 \nu_2 \\ x_3^2 \nu_2 \rho_{22} + x_2 x_3 \nu_2 \rho_{12} & x_2 x_3 \kappa \nu_1 \nu_2 & x_3^2 \nu_2^2 \end{pmatrix}.$$

Writing the Eikonal equation as before $a_{ij}\left(\frac{\partial d}{\partial x_i}\right)\left(\frac{\partial d}{\partial x_j}\right) = 1$.
For a homogeneous solution we apply the following variable transformation

$$z_1 = \nu_1 \frac{x_1}{x_2}, \qquad z_2 = \nu_2 \frac{x_1}{x_3}, \qquad d(z_1, z_2) = d(x_1, x_2, x_3).$$

The result is the following PDE

$$f_1 (d_{z_1})^2 + 2 f_2 d_{z_1} d_{z_2} + f_3 (d_{z_2})^2 = 1,$$

where

$$f_1(z_1, z_2) = \nu_1^2 + \nu_2^2 \frac{z_1^2}{z_2^2} + 2\rho\nu_1\nu_2 \frac{z_1}{z_2} + \nu_1^2 z_1^2 - 2\rho_{11}\nu_1^2 z_1 - 2\rho_{21}\nu_1\nu_2 \frac{z_1^2}{z_2},$$

$$f_2(z_1,z_2) = 2\rho\nu_1\nu_2 + \nu_1^2\frac{z_2}{z_1} + \nu_2^2\frac{z_1}{z_2} - \rho_{11}\nu_1^2 z_2 - \rho_{21}\nu_1\nu_2 z_1 - \rho_{22}\nu_2^2 z_1 - \rho_{12}\nu_1\nu_2 z_2 + \kappa\nu_1\nu_2 z_1 z_2,$$

$$f_3(z_1,z_2) = \nu_2^2 + \nu_1^2\frac{z_2^2}{z_1^2} + 2\rho\nu_1\nu_2\frac{z_2}{z_1} + \nu_2^2 z_2^2 - 2\rho_{22}\nu_2^2 z_2 - 2\rho_{12}\nu_1\nu_2\frac{z_2^2}{z_1}.$$

Unfortunately there's no closed form solution for this 2 dimensional PDE. But we can apply our numerical scheme which has been introduced in Section 2.4.1. Of course, there's one important change yet to be made, the original Eikonal update formula have to be modified.
Using the same notation as in Section 2.4.1 the discretized version of the equation is

$$f_1(U_{ij} - U_H)^2 + 2f_2(U_{ij} - U_H)(U_{ij} - U_V) + f_3(U_{ij} - U_V)^2 = h^2,$$

and the solution for U_{ij} is

$$U_{ij} = \frac{U_H(f_1 + f_2) + U_V(f_2 + f_3) + \sqrt{(f_2^2 - f_1 f_3)(U_H - U_V)^2 + h^2(f_1 + 2f_2 + f_3)}}{f_1 + 2f_2 + f_3}.$$

To get the implied volatility one should apply the following formula with v_1 and v_2 being the initial values of the underlying volatilities:

$$\sigma_{imp} = \frac{x}{d(\nu_1\frac{x}{v_1}, \nu_2\frac{x}{v_2})}.$$

The 1 dimensional SABR case (as in Section 3.3.1) can also be solved numerically using the fast marching method. We can get the Eikonal uptade formula of that case if we write

$$f_1 = y^2, \qquad f_2 = \alpha\rho y^2, \qquad f_3 = \alpha^2 y^2.$$

The resulted formula for 1 dimensional case is

$$U_{ij} = \frac{U_H(1 + \alpha\rho) + U_V(\alpha\rho + \alpha^2) + \sqrt{\frac{h^2}{y^2}(1 + 2\alpha\rho + \alpha^2) - \alpha^2(1 - \rho^2)(U_H - U_V)^2}}{1 + 2\alpha\rho + \alpha^2}.$$

4 Monte Carlo simulation - The mSABR method

4.1 Introduction to the method

In this Section we will introduce a new efficient technique for simulating SABR dynamics, that has been published in a recent paper [5]. This effective method is called the multiple time-step Monte Carlo simulation technique for SABR dynamics or shorter the mSABR method.

This is an extension of the author's previous work [14], where the one time step Monte Carlo method has been introduced. The technique consists of many highly interesting and nontrivial components, like Fourier inversion for the sum of log-normals, stochastic collocation, Gumbel copula, correlation approximation, that are not yet seen in combination within a Monte Carlo simulation.

Using the notation in the paper, the SABR model reads

$$\begin{aligned} dS(t) &= \sigma_t S^{\beta}(t) dW_S(t), & S(0) &= S_0 \exp(rT), \\ d\sigma(t) &= \alpha\sigma(t) dW_\sigma(t), & \sigma(0) &= \sigma_0, \end{aligned}$$
$$\big\langle dW_S(t) dW_\sigma(t) \big\rangle = \rho dt.$$

Here $S(t) = \overline{S}(t)\exp(r(T-t))$ denotes the forward price of the underlying asset $\overline{S}(t)$, with r the interest rate, S_0 the spot price and T the maturity.
If we want to work with independent Brownian motions, consider the following transformation:

$$W_\sigma(t) = \hat{W}_\sigma(t), \qquad W_S(t) = \rho\hat{W}_\sigma(t) + \sqrt{1-\sigma^2}\hat{W}_S(t)$$

It is known that for some generic time interval $[s;t]$, $0 \leq s < t \leq T$, assuming $S(s) > 0$, the conditional cumulative distribution for forward $S(t)$ with an absorbing boundary at$S(t) = 0$, given $\sigma(s)$, $\sigma(t)$ and $\int_s^t \sigma^2(z)dz$, is given by

$$\mathbb{P}\left(S(t) \leq K \,|\, S(s) > 0,\, \sigma(s),\, \sigma(t),\, \int_s^t \sigma^2(z)dz \right) = 1 - \chi^2(a; b, c),$$

where a, b and c are fixed parameters respect to $S(s)$, $\sigma(s)$, $\sigma(t)$ and $\int_s^t \sigma^2(z)dz$ and $\chi^2(a; b, c)$ is the non-central chi-square cumulative distribution function.
For the algorithm, several steps need to be performed, that are described in the following:

- *Simulation of the SABR volatility process, $\sigma(t)$ given $\sigma(s)$.* The stochastic volatility process of the SABR model exhibits a lognormal distribution. The solution is a geometric Brownian motion, i.e. the exact simulation of $\sigma(t)|\sigma(s)$ reads
$$\sigma(t) \sim \sigma(s)\exp(\alpha(W_\sigma(t) - W_\sigma(s)) - \frac{1}{2}\alpha^2(t-s))$$
- *Simulation of the SABR integrated variance process, $\int_s^t \sigma^2(z)dz \,|\, \sigma(t), \sigma(s)$.* This conditional distribution is not available in closed-form. We will therefore derive an approximation of the conditional distribution of the SABR integrated variance given $\sigma(t)$ and $\sigma(s)$. The integrated variance sampling can be done by simply inverting it.

- *Simulation of the SABR forward price process.* The forward price $S(t)$ can be simulated by inverting the CDF above. By this, we avoid negative forward prices in the simulation, as an absorbing boundary at zero is considered. There is no analytic expression for the inverse distribution and therefore this inversion has to be computed by means of some numerical approximation.

4.1.1 The Stochastic Collocation Monte Carlo sampler

The authors have proposed a procedure to sample $\int_s^t \sigma^2(z)dz \,|\, \sigma(t), \sigma(s)$ based on the Gumbel copula. For this, the CDF of the integrated variance given the initial volatility, $\sigma(s)$, (as a marginal distribution) must be derived. They used a recursive technique to obtain an approximation of this CDF.

Because we need to apply this recursion to approximate the characteristic function, the PDF and the CDF of $\int_s^t \sigma^2(z)dz \,|\, \sigma(s)$ for each sample of $\sigma(s)$ at each time-step, this approach is expensive in terms of computational cost. To overcome this drawback, an efficient alternative will be employed here, based on Lagrange interpolation, as in the Stochastic Collocation Monte Carlo sampler (SCMC).

The SCMC technique relies on the property that a CDF of a distribution (if it exists) is uniformly distributed. A well-known standard approach to sample from a given distribution, Y, with CDF F_Y reads

$$F_Y(Y) \overset{d}{=} U \text{ thus } y_n = F_Y^{-1}(u_n);$$

where u_n are samples from $\mathcal{U}[0;1]$. The computational cost of this approach highly depends on the cost of the inversion F_Y^{-1}, which is assumed to be rather expensive.

We therefore consider another, "cheap", random variable X, whose inversion, F_X^{-1}, is computationally much less expensive. In this framework, the following relation holds

$$F_Y(Y) \overset{d}{=} U \overset{d}{=} F_X(X).$$

However, this does not yet imply any improvement since the number of expensive inversions F_Y^{-1} remains the same. The goal is to compute y_n by using a function $g(\cdot) = F_Y^{-1}(F_X(\cdot))$, such that

$$F_X(x) = F_Y(g(x)) \text{ and } Y \overset{d}{=} g(X);$$

where evaluations of function $g(\cdot)$ do not require many inversions F_Y^{-1}.

This function $g(\cdot)$ is approximated by means of Lagrange interpolation, which is a well-known interpolation also used in the Uncertainty Quantification (UQ) context. The result is a polynomial, $g_{N_Y}(\cdot)$, which approximates function $g(\cdot) = F_Y^{-1}(F_X(\cdot))$, and the samples y_n can be obtained by employing $g_{N_Y}(\cdot)$ as

$$y_n \approx g_{N_Y}(x_n) = \sum_{i=1}^{N_Y} y_i l_i(x_n), \qquad l_i(x_n) = \prod_{j=1, j\neq i}^{N_Y} \frac{x_n - \overline{x}_j}{\overline{x}_i - \overline{x}_j}$$

where x_n is a vector of samples from X and $\overline{x}_j$ are the so-called collocation points. N_Y represents the number of collocation points and y_i the exact inversion value of F_Y at the collocation point $\overline{x}_i$, i.e. $y_i = F_Y^{-1}(F_X(x_i))$. By applying this interpolation, the number of inversions is reduced and, with only N_Y expensive inversions $F_Y^{-1}(F_X(x_i))$, we can generate any number of samples by evaluating the polynomial $g_{N_Y}(x_n)$.

A crucial aspect for the computational cost is the parameter N_Y. The collocation points must be optimally chosen in a way to minimize their number. The optimal collocation points are here chosen to be Gauss quadrature points that are defined as the zeros of the related orthogonal polynomial. This approach leads to a stable interpolation under the probability distribution of X. Since we deal with a conditional distribution, the 2D version of SCMC needs to be used.

4.2 Components of the mSABR method

In this section, we will discuss the different components of the multiple time-step Monte Carlo method for the SABR model. For simplicity, hereafter, we denote the SABR's integrated variance process by $Y(s,t) := \int_s^t \sigma^2(z)dz$. We will explain how to efficiently sample the integrated variance given the initial and the final volatility, as well as its use in a complete SABR simulation. Since the distribution is not available in closed-form, some approximations need to be made.

The authors proposed an accurate sampling method based on copula theory, which is employed to approximate the required conditional distributions. The copula relies on the availability of the marginal distributions to simulate the joint distribution. As the marginal distributions, $Y(s,t)|\sigma(s)$ and $\sigma(t)|\sigma(s)$ appear as the natural choices. A procedure to recover the CDF of the integrated variance process given the initial volatility will be presented.
The algorithm to sample $Y(s,t)$ given $\sigma(t)$ and $\sigma(s)$ consists of the following steps:

1. Determine $F_{\log\sigma(t)|\log\sigma(s)}$. For this to approximate we need to determine the correlation between $\log Y(s,t)$ and $\log\sigma(t)$.

2. Determine $F_{\log\hat{Y}|\log\sigma(s)}$, where $\hat{Y}$ is the discretized version of Y.

3. Generate correlated uniform samples, $U_{\log\sigma(t)|\log\sigma(s)}$ and $U_{\log\hat{Y}|\log\sigma(s)}$ from the Gumbel copula.

4. From $U_{\log\sigma(t)|\log\sigma(s)}$, invert the CDF $F_{\log\sigma(t)|\log\sigma(s)}$ to get the samples $\hat{\sigma}$ of $\log\sigma(t)|\log\sigma(s)$. This procedure is straightforward since the normal distribution inversion is analytically available.

5. From $U_{\log\hat{Y}|\log\sigma(s)}$, invert $F_{\log\hat{Y}|\log\sigma(s)}$ to get the samples $\hat{y}_n$ of $\log\hat{Y}|\log\sigma(s)$. We propose an inversion procedure based on linear interpolation. First we evaluate the CDF function at some discrete points. Then, the insight is that, by rotating the CDF under consideration, we can interpolate over probabilities. This is possible when the CDF function is a smoothly increasing function. The interpolation polynomial provides the quantiles of the original distribution from some given probabilities. Since $F_{\log\hat{Y}|\log\sigma(s)}$ is indeed a smooth and increasing function, the interpolationbased inversion is definitely applicable. This procedure together with the use of 2D SCMC sampler results in a fast and accurate inversion.

6. The samples σ_n of $\sigma(t)|\sigma(s)$ and y_n of $Y(s,t) = \int_s^t \sigma^2(z)dz|\sigma_t, \sigma_s$ are obtained by simply taking exponentials as

$$\sigma_n = \exp(\hat{\sigma}_n), \qquad\qquad y_n = \exp(\hat{y}_n).$$

Step 1. Determining $F_{\log\sigma(t)|\log\sigma(s)}$

For the first step we employ the expression of a conditional normal distribution. Hence, the distribution of $\log\sigma(t)|\log\sigma(s) = z$ is given by

$$\mathcal{N}\left(\mu_{\log\sigma(t)} + \mathcal{P}_{\log\sigma(t);\log\sigma(s)}\frac{s_{\log\sigma(t)}}{s_{\log\sigma(s)}}\left(z - \mu_{\log\sigma(t)}\right)\, ,\, s_{\log\sigma(t)}\sqrt{1 - \mathcal{P}^2_{\log\sigma(t);\log\sigma(s)}}\right),$$

where $\mu_{\log\sigma(t)}$ and $\mu_{\log\sigma(s)}$ are the means and $s_{\log\sigma(t)}$ and $s_{\log\sigma(s)}$ are the standard deviations of $\log\sigma(t)$ and $\log\sigma(s)$, respectively. $\mathcal{P}_{\log\sigma(t);\log\sigma(s)}$ is the Pearson correlation coefficient which is approximated as follows

$$\mathcal{P}_{\log\sigma(t);\log\sigma(s)} \approx \frac{t^2 - s^2}{2\sqrt{\frac{1}{3}t^4 + \frac{2}{3}ts^3 - t^2s^2}}.$$

Step 2. Determining $F_{\log\hat{Y}|\log\sigma(s)}$

The CDF of $\log\hat{Y}|\log\sigma(s)$ is resulted from

$$F_{\log\hat{Y}|\log\sigma(s)} = \int_{-\infty}^{x} f_{\log\hat{Y}|\log\sigma(s)}(y)dy,$$

where $f_{\log\hat{Y}|\log\sigma(s)}$ is the PDF of $\log\hat{Y}|\log\sigma(s)$. This can be found by approximating the associated characteristic function, $\phi_{\log\hat{Y}|\log\sigma(s)}$, and applying a Fourier inversion procedure. We can define a recursive procedure to recover the characteristic function of $f_{\log\hat{Y}|\log\sigma(s)}$.
We start by defining the sequence,

$$R_j = \log\left(\frac{\sigma^2(t_j)}{\sigma^2(t_{j-1})}\right), \qquad j = 1, \ldots, M.$$

At this point, a backward recursion procedure in terms of R_j will be defined by which we can recover $\phi_{\log\hat{Y}|\log\sigma(s)}$. We define

$$Y_1 = R_M,\, Y_j = R_{M+1-j} + Z_{j-1}, \qquad j = 2, \ldots, M$$

with $Z_j = \log(1 + \exp(Y_j))$.
After applying the definition of characteristic function, we determine $\phi_{\log\hat{Y}|\log\sigma(s)}$ as follows

$$\phi_{\log\hat{Y}|\log\sigma(s)}(u) = \exp(iu\log(\Delta t\sigma^2(s)))\phi_{Y_M}(u).$$

As Y_M is defined recursively, its characteristic function can be obtained by a recursion as well. According to the definition of the (backward) sequence Y_j, the initial and recursive characteristic functions are given by the following expressions,

$$\phi_{Y_1}(u) = \phi_{R_M}(u) = \phi_R(u) = \exp(-iu\alpha^2\Delta t - 2u^2\alpha^2\Delta t),$$

$$\phi_{Y_j}(u) = \phi_{R_{M+1-j}}(u)\phi_{Z_{j-1}}(u) = \phi_R(u)\phi_{Z_{j-1}}(u).$$

By definition, the characteristic function of Z_{j-1} reads

$$\phi_{Z_{j-1}}(u) = \int_{-\infty}^{\infty} (\exp(x)+1)^{iu} f_{Y_{j-1}}(x)dx.$$

Probability density function $f_{Y_{j-1}}$ is not known. To approximate it, the Fourier cosine series expansion on $f_{Y_{j-1}}$ is applied. Based on the cumulant-based approach we truncate the integration range to $[a,b]$.

$$\phi_{Z_{j-1}}(u) \approx \frac{2}{b-a}\sum_{l=0}^{N-1} B_l \int_a^b (\exp(x)+1)^{iu} \cos\left((x-a)\frac{l\pi}{b-a}\right) dx =: \hat{\phi}_{Z_{j-1}}(u),$$

$$B_l = \Re\left(\hat{\phi}_{Y_{j-1}}(\frac{l\pi}{b-a})\exp\left(-ia\frac{l\pi}{b-a}\right)\right),$$

with N the number of cosine expansion elements, and where

$$\hat{\phi}_{Y_1}(u) := \phi_R(u), \qquad \hat{\phi}_{Y_j}(u) := \phi_R(u)\hat{\phi}_{Z_{j-1}}(u).$$

Considering the equation for $\hat{\phi}_{Z_{j-1}}(u)$ in matrix-vector form, by recursion procedure, we obtain the approximation $\hat{\phi}_{Y_M}$ of the characteristic function ϕ_{Y_M} of Y_M.

The authors have shown that for numerical approximation of the integral based on a piecewise linear approximation provides a good balance between performance and accuracy. For an efficient sampling from the logistic distribution, we also have to introduce a scale parameter so that the quantiles have to be more evenly distributed.

Now we have every component to derive the PDF of $\log\hat{Y}|\log\sigma(s)$ using the so-called COS method

$$f_{\log\hat{Y}|\log\sigma(s)}(x) \approx \frac{2}{b-a}\sum_{k=0}^{N-1} C_k \cos\left((x-a)\frac{k\pi}{b-a}\right),$$

with

$$C_k = \Re\left(\phi_{\log\hat{Y}|\log\sigma(s)}\left(\frac{k\pi}{b-a}\right)\exp\left(-ia\frac{k\pi}{b-a}\right)\right).$$

Step 3. Generating samples

For generating correlated uniform samples in Step 3, we will use the Archimedean Gumbel copula. Considering $F_1, \dots F_d \in [0,1]^d$ as the marginal distributions, the Gumbel copula reads

$$C_\theta(F_1,\dots F_d) = \exp\left(-\left(\sum_{i=1}^{d}(-\log(F_i))^\theta\right)^{1/\theta}\right),$$

where the parameter θ is found to be equal $\theta = 1/(1-\tau)$, where τ is the Kendall's coefficient which we can get from the Pearson's coefficient using

$$\mathcal{P} = \sin\left(\frac{\pi}{2}\tau\right).$$

Step 5. Efficient sampling of $\log \hat{Y} | \log \sigma(s)$

By employing the SCMC technique, instead of directly computing $F_{\log \hat{Y} | \log \sigma(s)}$ for each sample of $\log \sigma(s)$, we only have to compute it at the collocation points. In general, only a few collocation points are sufficient to obtain accurate approximations, which is a well-known fact from the UQ research field. This fact allows us to drastically reduce the computational cost of sampling the required distribution.

For the problem at hand, we require samples from the integrated variance conditional on the initial volatility, $\log \hat{Y}(s,t) | \log \sigma(s)$. Therefore, we need to make use of the 2D version of the SCMC technique. Two levels of collocation points need to be chosen, one for each dimension. If we denote them by $N_{\hat{Y}}$ and N_σ, respectively, the resulting number of inversions equals $N_{\hat{Y}} \cdot N_\sigma$. The formal definition of the 2D SCMC technique applied to our context reads

$$y_n | v_n \approx g_{N_{\hat{Y}}, N_\sigma}(x_n) = \sum_{i=1}^{N_{\hat{Y}}} \sum_{j=1}^{N_\sigma} F^{-1}_{\log \hat{Y} | \log \sigma(s) = \overline{v}_j}(F_X(\overline{x}_i)) l_i(x_n) l_j(v_n),$$

where x_n are the samples from the standard normal distribution, which is used as the cheap variable, and v_n the samples of $\log \sigma(s)$; $\overline{x}_i$ and $\overline{v}_j$ are the collocation points for approximating variables $\log Y$ and $\log \sigma(s)$, respectively. The l_i and l_j are Lagrange polynomials fitted to their collocation points respectively.

4.3 Simulation of $S(t)$ given $S(s)$, $\sigma(s)$, $\sigma(t)$ and $\int_s^t \sigma^2(z)dz$

We complete the mSABR method by the conditional sampling of $S(t)$. The most commonly used techniques can be classified in two categories: direct inversion of the SABR distribution function given in Section 4.1 and moment-matching approaches. The direct inversion procedure has a higher computational cost because of the evaluation of the non-central χ^2 distribution.

Note however that, for some specific values of β, the simulation of the conditional $S(t)$ given $S(s)$, $\sigma(s)$, $\sigma(t)$ and $\int_s^t \sigma^2(z)dz$ enables analytic expressions. This is the case for $\beta = 0$ and $\beta = 1$ and we will describe the latter.

Case $\beta = 1$

The asset dynamics of the SABR model become log-normal and the solution is given by

$$S(t) = S(s) \exp\left(-\frac{1}{2} \int_s^t \sigma^2(z)dz + \rho \int_s^t \sigma(z) d\hat{W}_\sigma(z) + \sqrt{1-\rho^2} \int_s^t \sigma(z) d\hat{W}_S(z) \right).$$

If we take the log-transform

$$\log\left(\frac{S(t)}{S(s)}\right) = -\frac{1}{2} \int_s^t \sigma^2(z)dz + \rho \int_s^t \sigma(z) d\hat{W}_\sigma(z) + \sqrt{1-\rho^2} \int_s^t \sigma(z) d\hat{W}_S(z),$$

and by considering

$$\int_s^t \sigma(z) d\hat{W}_\sigma(z) = \frac{1}{\alpha}(\sigma(t) - \sigma(s)),$$

and

$$\int_s^t \sigma(z) d\hat{W}_S(z) | \sigma(t), \sigma(s) \sim \mathcal{N}\left(0, \sqrt{\int_s^t \sigma^2(z) dz}\right)$$

we obtain the distribution of $\log\left(\frac{S(t)}{S(s)}\right) \mid \int_s^t \sigma^2(z) dz, \sigma(t), \sigma(s)$, which reads

$$\mathcal{N}\left(-\frac{1}{2}\int_s^t \sigma^2(z) dz + \frac{\rho}{\alpha}(\sigma(t) - \sigma(s)), \sqrt{(1-\rho^2)\int_s^t \sigma^2(z) dz}\right).$$

So as a conclusion, the asset dynamics $S(t)$ can be sampled from

$$S(t) = S(s) \exp\left(-\frac{1}{2}\int_s^t \sigma^2(z) dz + \frac{\rho}{\alpha}(\sigma(t) - \sigma(s)) + X\sqrt{(1-\rho^2)\int_s^t \sigma^2(z) dz}\right),$$

where X is the standard normal.

5 Numerical results

The experiments were performed on a computer with CPU Intel Core i7-3610QM 2.30GHz and RAM memory of 6 Gigabytes. The employed software package was Mathematica v10.1.

5.1 Results for 1D SABR

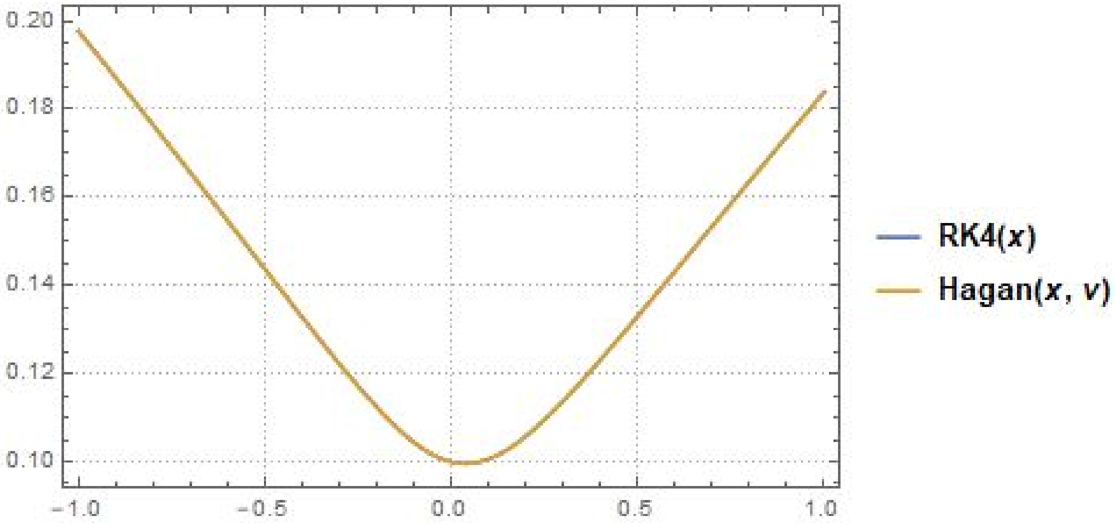
The first experiment was to compare the analytical solution to the classical Runge-Kutta method. Because of the rather similar results we got when changing the parameters, we will fix them as $\alpha = 0.4$, $\rho = 0.1$ and $\sigma_0 =: v = 0.1$.

As a result we plotted the implied volatility curves, the relative and absolute differences (respect to the analytical solution) as a function of $x = \log\left(\frac{F_0}{K}\right)$ in the interval [-1,1] (see Figure 4).

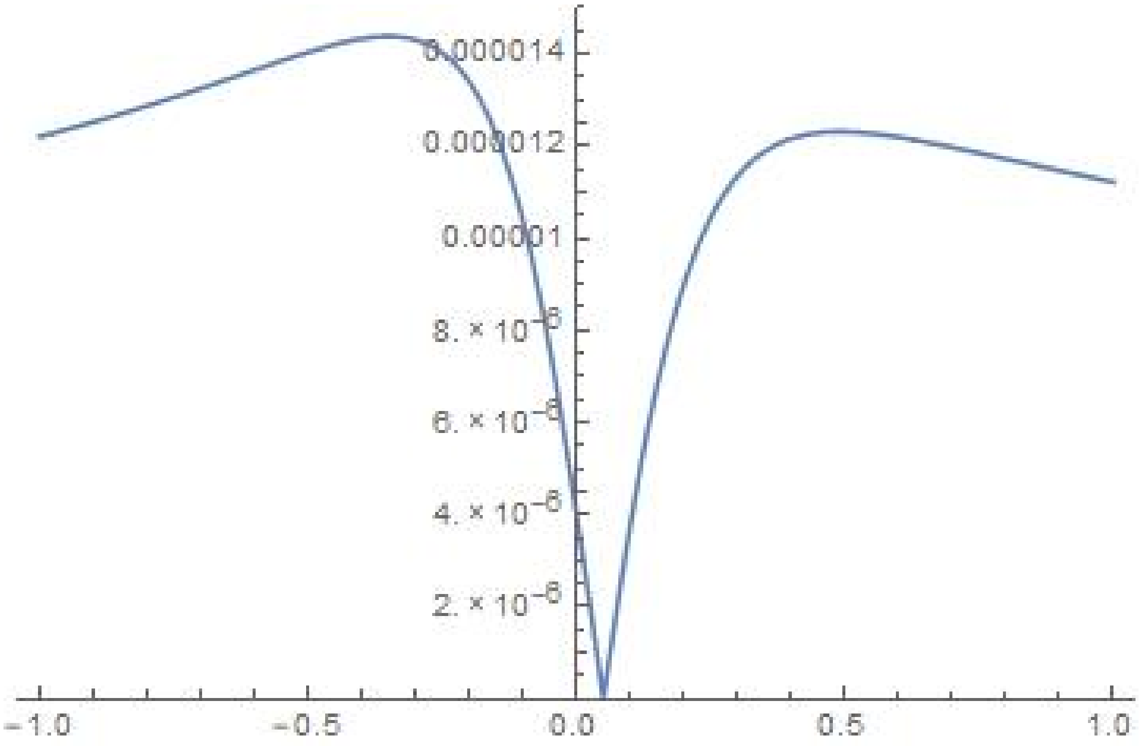
The only variable that remains is the stepsize h because from that the necessary number of data points can easily be calculated as $\left(\frac{2}{hv}+1\right)$. The boundary condition for the RK4 method was that at 0 the geodesic distance equals to 0. See below Table for the results.

Stepsize (h)	Calculation time (sec.)	Relative difference	Absolute difference
0.1	0.015	$1.5 \cdot 10^{-2}$	$2.5 \cdot 10^{-3}$
0.01	0.062	$1.4 \cdot 10^{-3}$	$2.5 \cdot 10^{-4}$
0.001	0.641	$1.4 \cdot 10^{-4}$	$2.5 \cdot 10^{-5}$
0.0001	6.546	$1.4 \cdot 10^{-5}$	$2.5 \cdot 10^{-6}$

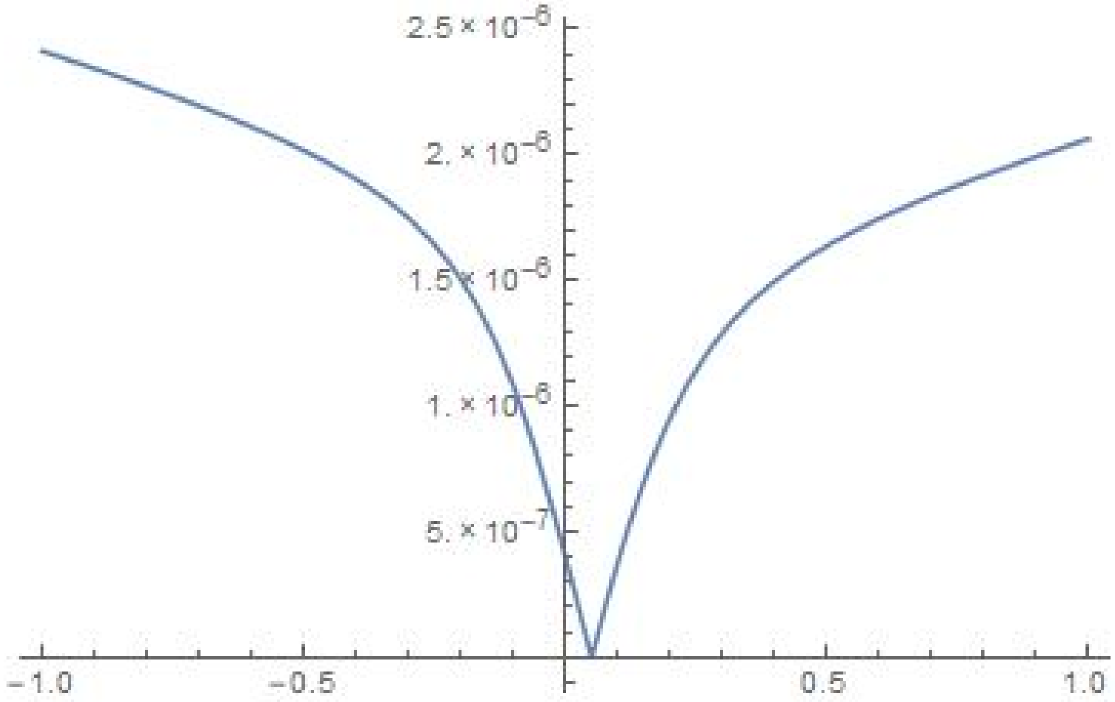
As it can be observed the computation time is a linear function of the nuber of data points and the differences are linear functions of the step size.



(a) Implied volatility smile



(b) Relative difference of the curves (h=0.0001)



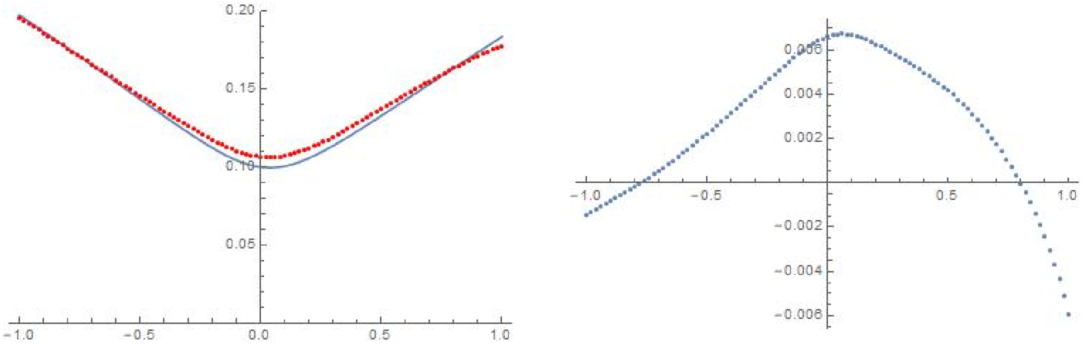
(c) Absolute difference of the curves (h=0.0001)

Figure 4. Implied volatility smile of the SABR model with $\alpha = 0.4$, $\beta = 1$, $\rho = 0.1$, $\sigma_0 = 0.1$

The second experiment was to compare the analytical solution to the result we get from Monte Carlo simulation. I worked with the same parameters as previously.

It's an ongoing project to compare the result to the mSABR method from Section 4.

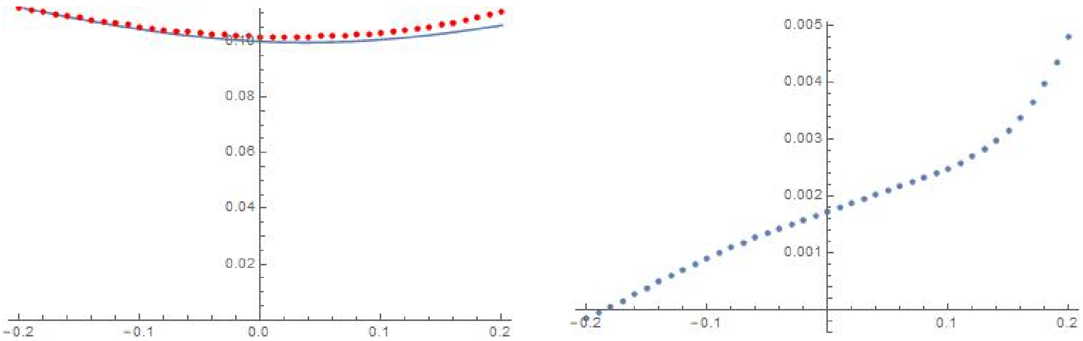
I used a brute force Monte Carlo simulation in comparison with the analytical solution using the Euler scheme. The calculations were made with 5Y, 1Y and 1M European Call options with an initial value of the underlying 1 and various strikes, which are evenly distributed on a logarithmic scale, on 1,000,000 paths. The stock price and the volatility was simulated in discrete time instants. There was 50 timesteps in the 5Y case ($\approx$ 2.5 months), 20 timesteps in the 1Y case ($\approx$ 0.5 month) and 10 timesteps in the 1M case ($\approx$ 3 days). As $T \to 0$ the result near ATM is improving.



(a) First Monte Carlo simulation

(b) Absolute difference

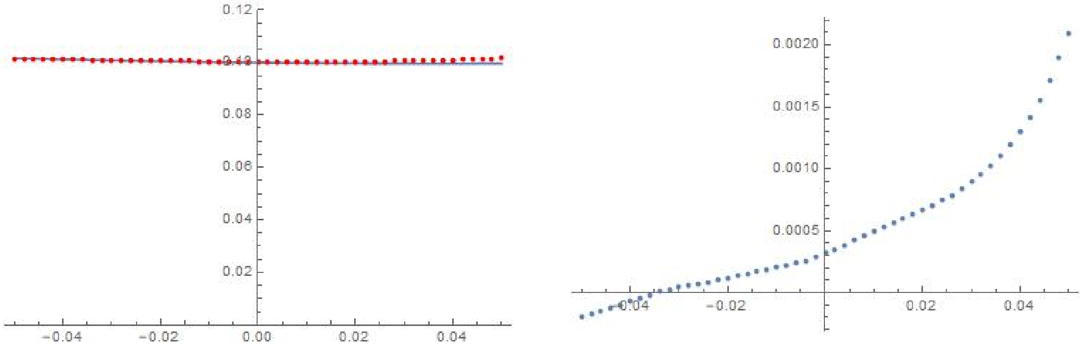
Figure 5. Implied volatility of a 5Y European Call



(a) Second Monte Carlo simulation

(b) Absolute difference

Figure 6. Implied volatility of a 1Y European Call

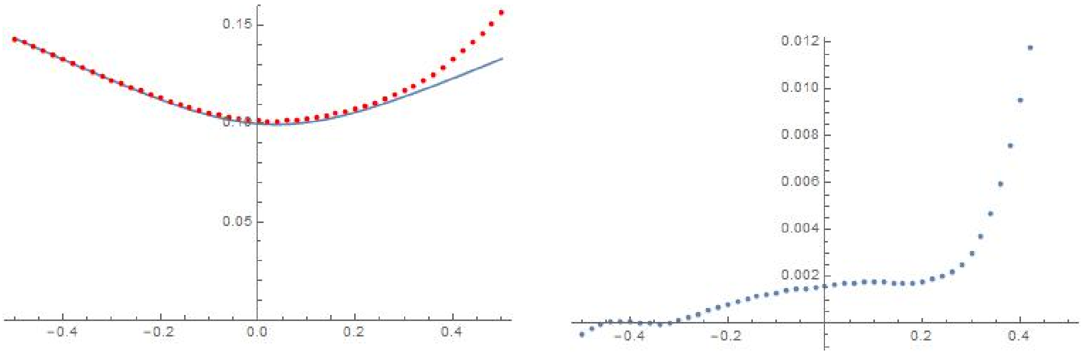


(a) Third Monte Carlo simulation

(b) Absolute difference

Figure 7. Implied volatility of a 1M European Call

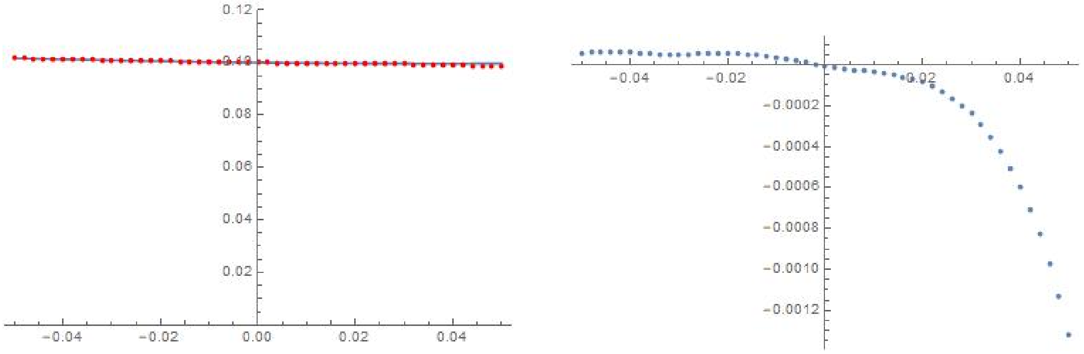
We checked the effect of quadrupling the timesteps in case of the 1Y and the 1M European call options. As we can see from the figures, the accuracy is greatly improved in case of the 1M but not so much for the 1Y.



(a) First Monte Carlo simulation with timestep bump

(b) Absolute difference

Figure 8. Implied volatility of a 1Y European Call



(a) Second Monte Carlo simulation with timestep bump

(b) Absolute difference

Figure 9. Implied volatility of a 1M European Call

5.2 Results for 2D SABR

The simulation was the following. First I defined a discrete mesh as to represent a square in the Euclidian space $[0, L]^2 \in \mathbb{R}^2$. (The same algorithm will have to be done for $[-L, 0]^2 \in \mathbb{R}^2$ as well.)

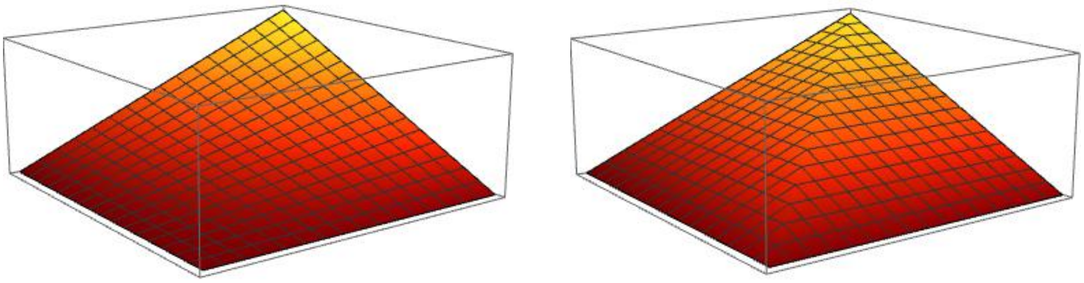
I implemented the FMM in Mathematica as described in Section 3.3.2 and performed the algorithm for fixed parametersets and algorithm parameters. The two types of parametersets included are:

- **Parameterset:** ν_1, ν_2, ρ, κ, ρ_{11}, ρ_{12}, ρ_{21}, ρ_{22}, v_1 and v_2 (the initial value of the volatilities);
- **Algorithm parameters:** h (step size), n (number of gridpoints-1), $L = h \cdot n$ (grid size) and $V = L \cdot \max\left(\frac{v_i}{\nu_i}\right)$ (size of the plot).

After the run we aquire the $d(z_1, z_2)$ values on the points of the mesh. Now comes the interpolation problem. However it seems obvious that we should use bilinear interpolation to receive the remaining values of the square but in this case it's not applicable for the later described reasons. I used a linear interpolation instead, where the values on the square are approximated via the convex combination of the highest value of the four edges and the two remaining sides.
To get the implied volatility one should apply the formula from Section 3.3.2

$$\sigma_{imp} = \frac{x}{d(\nu_1 \frac{x}{v_1}, \nu_2 \frac{x}{v_2})}.$$

Note that the only variable in this is x and the values we need from the (z_1, z_2) space are on a line that has a positive steepness and is passing through the origin. If we would apply bilinear interpolation, it would result a collapse at zero, because of the quadratic behaviour of the interpolation.



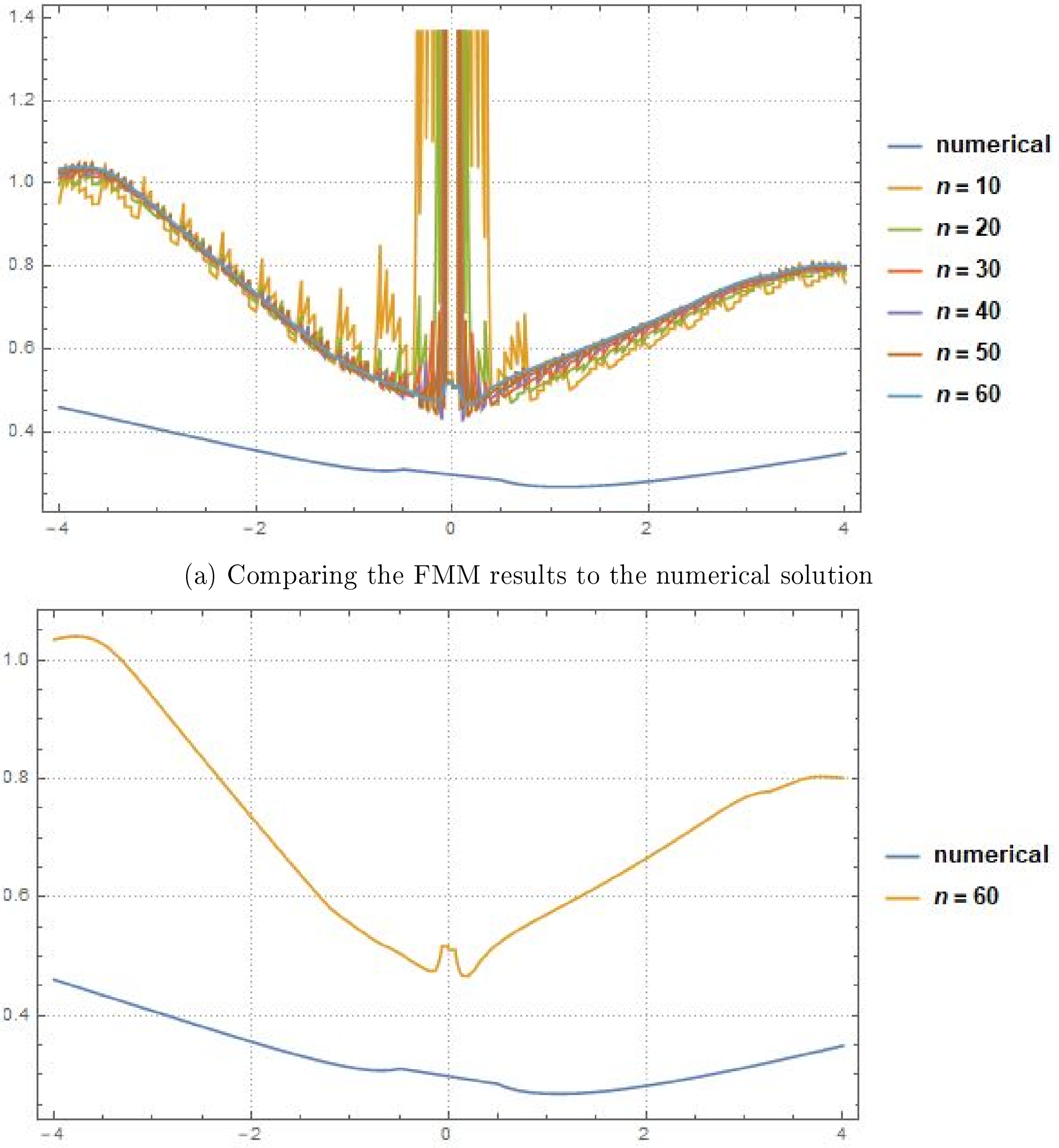
(a) Bilinear interpolation

(b) Quasilinear interpolation

Figure 10. Result from the two different interpolation techniqes around 0.

The first experiment was to test the numerical scheme resulted from FMM against the complete numerical solution. The test's parameterset was the same as in Figure 12, the algorithm parameters were determined by fixing the overall grid size ($L = 4$) and changing n and h accordingly. The following table includes the evaluation time of the algorithm while in Figures 11a and 11b the resulted implied volatility curves are compared. (Note that these time measures contain two FMM algorithms.)

n	10	20	30	40	50	60
Calculation time (sec.)	0.44	3.17	13.72	72.95	93.61	329.76



(a) Comparing the FMM results to the numerical solution

(b) Comparing the best FMM result to the numerical solution

Figure 11. Result of the first experiment

The FMM algorithm found to be inaccurate for determining the implied volatility so I wouldn't suggest using it for solving general Hamilton-Jacobi equations. Because of the Wolfram Language being a general multi-paradigm programming language, its built-in numerical solver is much more efficient than my algorithm.

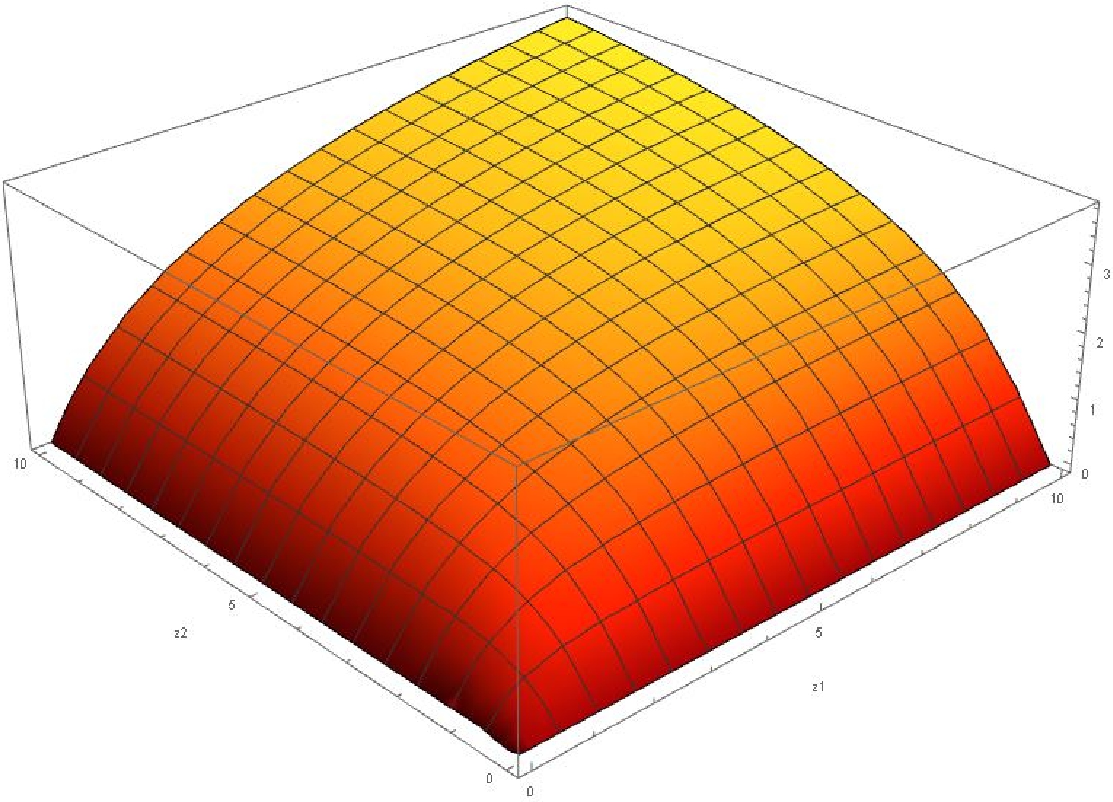


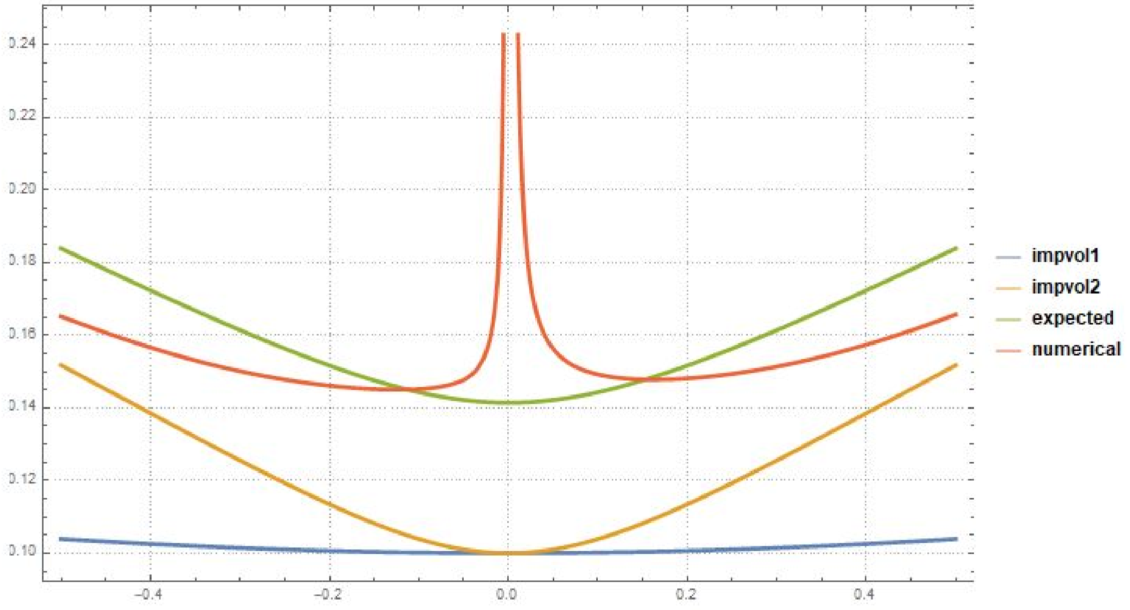
Figure 12. Numerical solution for $\nu_1 = 0.6$; $\nu_2 = 0.8$; $\rho = 0.4$; $\kappa = 0.5$; $\upsilon_1 = 0.5$; $\upsilon_2 = 0.3$; $\rho_{11} = -0.6$; $\rho_{12} = 0.2$; $\rho_{21} = -0.2$; $\rho_{22} = 0.4$

The second experiment is now to compare the numerical result to our expectations. The numerical solver is selected automatically from families of arbitrary-order implicit Runge-Kutta methods. Our expectation comes from the variance of two correlated samples

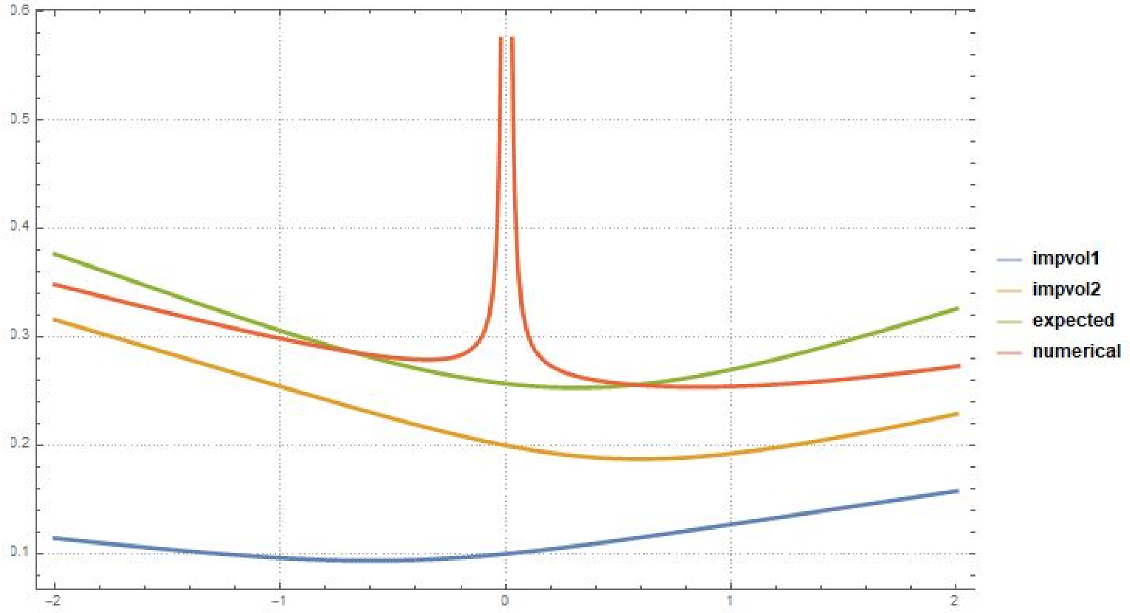
$$\sigma = \sqrt{\sigma_1^2 + \sigma_2^2 + 2\rho\sigma_1\sigma_2},$$

which is not completely accurate, but gives an intuitive picture of the total implied volatility. This expectation is perfectly accurate when the vol of vol parameters are becoming 0. Then the two rate processes has constant volatility, so from the normal property of the Wiener process we get the equation above.

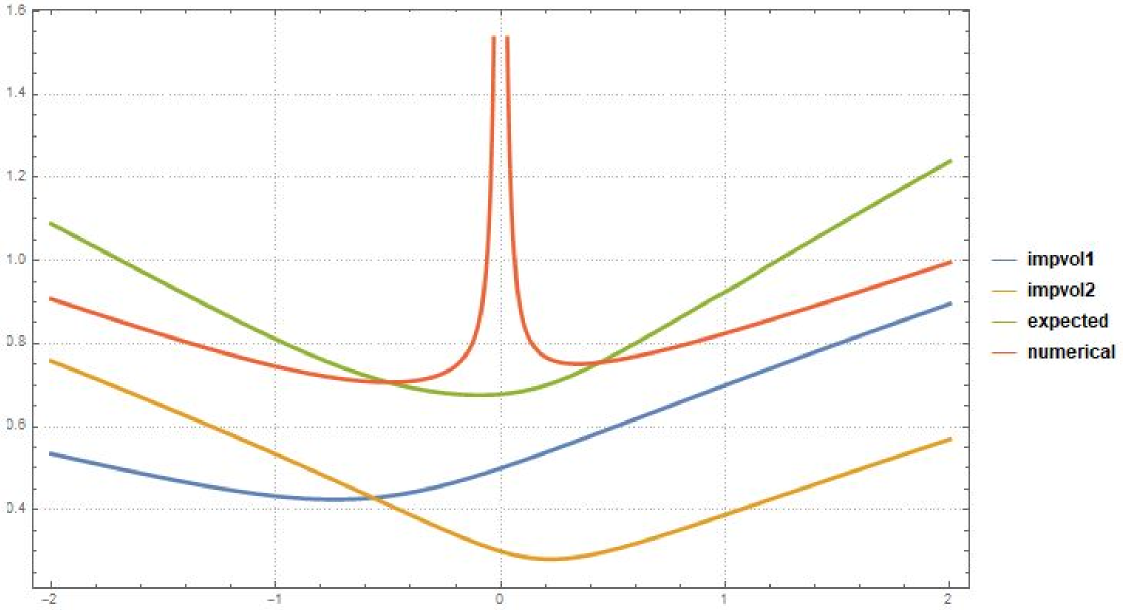
The tests were run on various parametersets and the calculation performed the best on those where at least one of the steepness ratios $(\frac{\nu_i}{\upsilon_i})$ is large enough. If this principle is used, the region around 0, where the implied volatility is mainly affected by the singularity in the origin, appeared to be small enough. The results are the following.



(a) Implied volatility smile with parameterset $\nu_1 = 0.1$; $\nu_2 = 0.5$; $\rho = 0$; $\kappa = 0$; $v_1 = 0.1$; $v_2 = 0.1$; $\rho_{11} = 0$; $\rho_{12} = 0$; $\rho_{21} = 0$; $\rho_{22} = 0$



(b) Implied volatility smile with parameterset $\nu_1 = 0.1$; $\nu_2 = 0.2$; $\rho = 0.4$; $\kappa = 0.5$; $v_1 = 0.1$; $v_2 = 0.2$; $\rho_{11} = -0.4$; $\rho_{12} = 0.2$; $\rho_{21} = -0.2$; $\rho_{22} = 0.4$



(c) Implied volatility smile with parameterset $\nu_1 = 0.6$; $\nu_2 = 0.8$; $\rho = 0.4$; $\kappa = 0.5$; $v_1 = 0.5$; $v_2 = 0.3$; $\rho_{11} = -0.6$; $\rho_{12} = 0.2$; $\rho_{21} = -0.2$; $\rho_{22} = 0.4$

Figure 13. Results of the second experiment

As a result the numerical solution performed reasonably well apart from the ATM's small radius. It is hard to make improvements there, because it's only affected by the d function's behaviour inside a small radius of the origin. That is why the steepness ratios should be large enough to quickly "escape" from there.

One other thing to notice is that near the ATM, the total volatility is close to our expectation. The explanation is that ATM volatilities are roughly the same as the starting volatilities so the intuitive formula is applicable there.

6 Conclusions

We applied a new method of computing the short time asymptotic implied volatility to the SABR model. A new interesting Monte Carlo simulation has been introduced, its implementation is an ongoing project.

We tested numerically the analytic solution of the Eikonal equation and found it satisfying. The solution has been verified via brute force Monte Carlo simulation. We also shown that for short maturities, raising the number of timesteps improves the accuracy. However to calculate the implied vol in case of short maturities for strikes far from ATM, a huge number of paths needed. This is really time consuming in case of the brute force Monte Carlo method so this is one more reason for implementing the mSABR method.

The 2 dimensional FMM scheme performed poorly. To mitigate the miscalculation in terms of the level of the curve probably a higher level approximation should be applied in the algorithm. The experiments showed that the numerical solution is close to our expectation inside a realistic range of the ATM.

Bibliography

[1] Hagan P., D. Kumar, A. S. Lesniewski and D. E.Woodward (2002), “Managing Smile Risk”

[2] Henry-Labordere, P. (2005), “A General Asymptotic Implied Volatility for Stochastic Volatility Models”

[3] Lewis, A. (2007), “Geometries and smile asymptotics for a class of Stochastic Volatility models”

[4] A. Antonov, M. Spector (2012), “Advanced analytics for the SABR model”

[5] A. Leitao, Lech A. Grzelaky and Cornelis W. Oosterleez (2016), “On an efficient multiple time-step Monte Carlo simulation of the SABR model”

[6] Varadhan, S.R.S., “On the behavior of the Fundamental Solution of the Heat Equation with Variable Coefficients”, Comm. Pure. Appl. Math. 20, 431-455 (1967)

[7] Varadhan, S.R.S., “Diffusion Processes in a Small Time Interval” Comm. Pure. Appl. Math. 20, 659-685 (1967)

[8] A. Chacon and A. Vladimirsky, “A parallel two-scale method for Eikonal equations” SIAM J. on Scientific Computing 37/1: A156-A180, (2015)

[9] Pierre A. Gremaud and Christofer M. Kuster, “Computational Study of Fast Methods for the Eikonal Equation” SIAM J. Sci. Comput., 27(6), 1803–1816. (2006)

[10] Adam Chacon and Alexander Vladimirsky, “Fast Two-scale Methods for Eikonal Equations” (2011)

[11] Jim Gatheral, “The Volatility Surface” (2006)

[12] Shu-Ren Hysing and Stefan Turek, “The Eikonal equation: Numerical efficiency vs. algorithmic complexity on quadrilateral grids” (2005)

[13] Fabio Mercurio and Nicola Moreni, “A Multi-Factor SABR Model for Forward Inflation Rates” (2009)

[14] A. Leitao, Lech A. Grzelaky and Cornelis W. Oosterleez, “On a one time-step SABR simulation approach: Application to European options”, Submitted to Applied Mathematics and Computation, (2016)