Efficient one and multiple time-step Monte Carlo simulation of the SABR model

Álvaro Leitao, Lech A. Grzelak and Cornelis W. Oosterlee

Delft University of Technology - Centrum Wiskunde & Informatica

Frankfurt - April 20, 2017

"Our" definition of simulation

- Generate samples from (sampling) stochastic processes.
- The standard approach to sample from a given distribution, Z:

$$F_Z(Z) \stackrel{\mathrm{d}}{=} U \quad \text{thus} \quad z_n = F_Z^{-1}(u_n),$$

- F_Z is the cumulative distribution function (CDF).
- $\stackrel{\mathrm{d}}{=}$ means equality in the distribution sense.
- $U \sim \mathcal{U}([0,1])$ and u_n is a sample from $\mathcal{U}([0,1])$.
- The computational cost depends on inversion F_Z^{-1}.

Outline

1. SABR model
2. Distribution of the SABR's integrated variance
3. One-step SABR simulation
4. Multiple time-step SABR simulation
5. Conclusions

SABR model

- The formal definition of the SABR model [6] reads

$$\begin{aligned} \mathrm{d}S(t) &= \sigma(t)S^{\beta}(t)\mathrm{d}W_S(t), & S(0) &= S_0 \exp\left(rT\right), \\ \mathrm{d}\sigma(t) &= \alpha\sigma(t)\mathrm{d}W_{\sigma}(t), & \sigma(0) &= \sigma_0. \end{aligned}$$

- $S(t) = \bar{S}(t)\exp\left(r(T-t)\right)$ is the forward price of the underlying $\bar{S}(t)$, with r an interest rate, S_0 the spot price and T the maturity.
- $\sigma(t)$ is the stochastic volatility.
- $W_f(t)$ and $W_{\sigma}(t)$ are two correlated Brownian motions.
- SABR parameters:
  - The volatility of the volatility, $\alpha > 0$.
  - The CEV elasticity, $0 \leq \beta \leq 1$.
  - The correlation coefficient, ρ ($W_f W_{\sigma} = \rho t$).

"Exact" simulation of SABR model

- Based on Islah [7], the conditional cumulative distribution function (CDF) of $S(t)$ in a generic interval $[s,t]$, $0 \leq s \leq t \leq T$:

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

where

$$a = \frac{1}{\nu(t)}\left(\frac{S(s)^{1-\beta}}{(1-\beta)} + \frac{\rho}{\alpha}\left(\sigma(t) - \sigma(s)\right)\right)^2,$$

$$c = \frac{K^{2(1-\beta)}}{(1-\beta)^2\nu(t)},$$

$$b = 2 - \frac{1 - 2\beta - \rho^2(1-\beta)}{(1-\beta)(1-\rho^2)},$$

$$\nu(t) = (1-\rho^2)\int_s^t \sigma^2(z)\mathrm{d}z,$$

and $\chi^2(x; \delta, \lambda)$ is the non-central chi-square CDF.

- Exact in the case of $\rho = 0$, an *approximation* otherwise.

Simulation of SABR model

- Simulation of the volatility process, $\sigma(t)|\sigma(s)$:

$$\sigma(t) \sim \sigma(s) \exp(\alpha \hat{W}_\sigma(t) - \frac{1}{2}\alpha^2 t),$$

  where $\hat{W}_\sigma(t)$ is a independent Brownian motion.
- Simulation of the integrated variance process, $\int_s^t \sigma^2(z)\mathrm{d}z|\sigma(t), \sigma(s)$.
- Simulation of the forward process, $S(t)|S(s), \int_s^t \sigma^2(z)\mathrm{d}z, \sigma(t), \sigma(s)$ by inverting the CDF.
- The conditional integrated variance is a challenging part. We propose:
  - Approximate the conditional distribution by using Fourier techniques and copulas.
  - Marginal distribution based on COS method [4].
  - Conditional distribution based on copulas.
  - Improvements in performance and efficiency.

Distribution of the integrated variance

- Not available.
- For notational convenience, we will use $Y(s,t) := \int_s^t \sigma^2(z)\mathrm{d}z$.
- Discrete equivalent, M monitoring dates:

$$Y(s,t) := \int_s^t \sigma^2(z)\mathrm{d}z \approx \sum_{j=1}^{M} \Delta t \sigma^2(t_j) =: \hat{Y}(s,t)$$

  where $t_j = s + j\Delta t$, $j = 1, \ldots, M$ and $\Delta t = \frac{t-s}{M}$.
- In the logarithmic domain, where we aim to find an approximation of $F_{\log \hat{Y} | \log \sigma(s)}$:

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

  where $f_{\log \hat{Y} | \log \sigma(s)}$ is the *probability density function* (PDF) of $\log \hat{Y}(s,t) | \log \sigma(s)$.

PDF of the integrated variance

- Equivalent: Characteristic function and inversion (Fourier pair).
- Recursive procedure to derive an approximated $\phi_{\log \hat{Y} | \log \sigma(s)}$.
- We start by defining the logarithmic increment of $\sigma^2(t)$:

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

- $\sigma^2(t_j)$ can be written:

$$\sigma^2(t_j) = \sigma^2(t_0)\exp(R_1 + R_2 + \cdots + R_j).$$

- We introduce the iterative process

$$\begin{aligned} Y_1 &= R_M, \\ Y_j &= R_{M+1-j} + Z_{j-1}, \quad j = 2, \ldots, M. \end{aligned}$$

with $Z_j = \log(1 + \exp(Y_j))$.

PDF of the integrated variance (cont.)

- $\hat{Y}(s,t)$ can be expressed:

$$\hat{Y}(s,t) = \sum_{i=1}^{M} \sigma^2(t_i)\Delta t = \Delta t \sigma^2(s) \exp(Y_M).$$

- And, we compute $\phi_{\log \hat{Y} | \log \sigma(s)}(u)$, as follows:

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

- By applying COS method [4] in the support $[\hat{a}, \hat{b}]$:

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

with

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

CDF of the integrated variance

- The CDF of $\log \hat{Y}(s,t) | \log \sigma(s)$:

$$F_{\log \hat{Y} | \log \sigma(s)}(x) = \int_{-\infty}^{x} f_{\log \hat{Y} | \log \sigma(s)}(y) \mathrm{d}y$$
$$\approx \int_{\hat{a}}^{x} \frac{2}{\hat{b} - \hat{a}} \sum_{k=0}^{N-1'} C_k \cos\left((y - \hat{a}) \frac{k\pi}{\hat{b} - \hat{a}} \right) \mathrm{d}y.$$

- The efficient computation of $\phi_{\log \hat{Y} | \log \sigma(s)}$ is crucial for the performance of the whole procedure (specially, one-step case).
- The inversion of $F_{\log \hat{Y} | \log \sigma(s)}$ is relatively expensive (unafforable in the multi-step case).

Copula-based simulation of $\int_s^t \sigma^2(z)\mathrm{d}z | \sigma(t), \sigma(s)$

- In order to apply copulas, we need (logarithmic domain):
  - $F_{\log \hat{Y} | \log \sigma(s)}$.
  - $F_{\log \sigma(t) | \log \sigma(s)}$.
  - Correlation between $\log Y(s,t)$ and $\log \sigma(t)$.
- The distribution of $\log \sigma(t) | \log \sigma(s)$ is known ($\sigma(t)$ follows a log-normal distribution).
- Approximated Pearson's correlation coefficient:

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

- For some copulas, like Archimedean, Kendall's τ is required:

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

Sampling $\int_s^t \sigma^2(z)\mathrm{d}z|\sigma(t),\sigma(s)$: Steps

1. Determine $F_{\log\sigma(t)|\log\sigma(s)}$ and $F_{\log\hat{Y}|\log\sigma(s)}$.
2. Determine the correlation between $\log Y(s,t)$ and $\log\sigma(t)$.
3. Generate correlated uniform samples, $U_{\log\sigma(t)|\log\sigma(s)}$ and $U_{\log\hat{Y}|\log\sigma(s)}$ by means of copula.
4. From $U_{\log\sigma(t)|\log\sigma(s)}$ and $U_{\log\hat{Y}|\log\sigma(s)}$ invert original marginal distributions.
5. The samples of $\sigma(t)|\sigma(s)$ and $Y(s,t)=\int_s^t \sigma^2(z)\mathrm{d}z|\sigma(t),\sigma(s)$ are obtained by taking exponentials.

One time-step simulation of the SABR model

- $s = 0$ and $t = T$, with T the maturity time.
- The use is restricted to price European options up to $T = 2$.
- $\log \sigma(s)$ becomes constant.
- $F_{\log \sigma(t) | \log \sigma(s)}$ and $F_{\log \hat{Y} | \log \sigma(s)}$ turn into $F_{\log \sigma(T)}$ and $F_{\log \hat{Y}(T)}$.
- The computation of $\phi_{\log \hat{Y}(T)}$ is much simpler and very fast.
- The approximated Pearson's coefficient results in a constant value:

$$\mathcal{P}_{\log Y(T), \log \sigma(T)} \approx \frac{T^2}{2\sqrt{\frac{1}{3}T^4}} = \frac{\sqrt{3}}{2}.$$

Approximated correlation

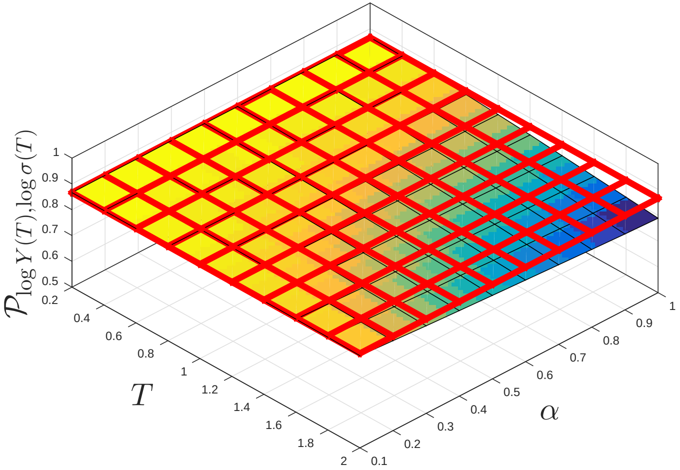


Figure: Pearson's coefficient: Empirical (surface) vs. approximation (red grid).

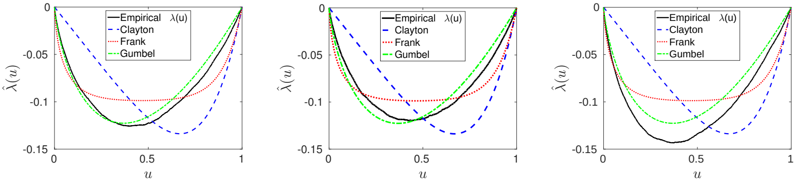
Copula analysis

- Based on the one-step simulation, a copula analysis is carried out.
- Gaussian, Student **t** and Archimedean (Clayton, Frank and Gumbel).
- A *goodness-of-fit (GOF)* for copulas needs to be evaluated.
- Archimedean: graphic GOF based on Kendall's processes.
- Generic GOF based on the so-called *Deheuvels or empirical copula*.

	S_0	σ_0	α	β	ρ	T
Set I	1.0	0.5	0.4	0.7	0.0	2
Set II	0.05	0.1	0.4	0.0	-0.8	0.5
Set III	0.04	0.4	0.8	1.0	-0.5	2

Table: Data sets.

GOF - Archimedean



(a) Set I. (b) Set II. (c) Set III.

Figure: Archimedean GOF test: $\hat{\lambda}(u)$ vs. empirical $\lambda(u)$.

	Clayton	Frank	Gumbel
Set I	1.3469×10^{-3}	2.9909×10^{-4}	5.1723×10^{-5}
Set II	1.0885×10^{-3}	2.1249×10^{-4}	8.4834×10^{-5}
Set III	2.1151×10^{-3}	7.5271×10^{-4}	2.6664×10^{-4}

Table: MSE of $\hat{\lambda}(u) - \lambda(u)$.

Generic GOF

	Gaussian	Student $\mathbf{t}$ $(\nu = 5)$	Gumbel
Set I	5.0323×10^{-3}	5.0242×10^{-3}	3.8063×10^{-3}
Set II	3.1049×10^{-3}	3.0659×10^{-3}	4.5703×10^{-3}
Set III	5.9439×10^{-3}	6.0041×10^{-3}	4.3210×10^{-3}

Table: Generic GOF: D_2.

- The three copulas perform very similarly.
- For longer maturities: Gumbel performs better.
- The Student $\mathbf{t}$ copula is discarded: very similar to the Gaussian copula and the calibration of the ν parameter adds extra complexity.
- As a general strategy, the Gumbel copula is the most robust choice.
- With short maturities, the Gaussian copula may be a satisfactory alternative.

Pricing European options

- The strike values K_i are chosen following the expression:

$$K_i(T) = S(0)\exp(0.1 \times T \times \delta_i),$$
$$\delta_i = -1.5, -1.0, -0.5, 0.0, 0.5, 1.0, 1.5.$$

- Forward asset, $S(t)$: enhanced inversion by Chen et al. [3].
- Martingale correction:

$$\begin{aligned} S(t) &= S(t) - \frac{1}{n}\sum_{i=1}^{n} S_i(t) + \mathbb{E}[S(t)], \\ &= S(t) - \frac{1}{n}\sum_{i=1}^{n} S_i(t) + S_0, \end{aligned}$$

Pricing European options - Convergence and time

	$n = 1000$	$n = 10000$	$n = 100000$	$n = 1000000$
	Gaussian (Set I, K_1)			
Error	519.58	132.39	37.42	16.23
Time	0.3386	0.3440	0.3857	0.5733
	Gumbel (Set I, K_1)			
Error	151.44	-123.76	34.14	11.59
Time	0.3492	0.3561	0.3874	0.6663

Table: Convergence in number of samples, n: error (basis points) and execution time (*sec.*).

Pricing European options - Implied volatilities

Strikes	K_1	K_2	K_3	K_4	K_5	K_6	K_7
	Set I (Reference: Antonov [1])						
Hagan	55.07	52.34	50.08	N/A	47.04	46.26	45.97
MC	23.50	21.41	19.38	N/A	16.59	15.58	14.63
Gaussian	16.23	20.79	24.95	N/A	33.40	37.03	40.72
Gumbel	11.59	15.57	19.12	N/A	25.41	28.66	31.79
	Set II (Reference: Korn [8])						
Hagan	−558.82	−492.37	−432.11	−377.47	−327.92	−282.98	−242.22
MC	5.30	6.50	7.85	9.32	10.82	12.25	13.66
Gaussian	9.93	9.98	10.02	10.20	10.57	10.73	11.04
Gumbel	−9.93	−9.38	−8.94	−8.35	−7.69	−6.83	−5.79
	Set III (Reference: MC Milstein)						
Hagan	287.05	252.91	220.39	190.36	163.87	141.88	126.39
Gaussian	16.10	16.76	16.62	15.22	13.85	12.29	10.67
Gumbel	6.99	3.79	0.67	−2.27	−5.57	−9.79	−14.06

Table: Implied volatility: errors in basis points.

- One-step SABR simulation is a fast alternative to Hagan formula.
- Overcomes the known issues, like low strikes and high volatilities.
- For longer maturities and more complex options: multiple time-step

Multiple time-step simulation of the SABR model

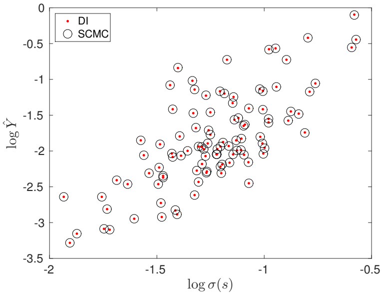
- We denote it *mSABR* simulation method (scheme).
- In intermediate steps, $\phi_{\log \hat{Y}|\log\sigma(s)}$ becomes "stochastic".
- $f_{\log \hat{Y}|\log\sigma(s)}$ needs to be computed for each sample of $\log\sigma(s)$.
- Consequently, the inversion of $F_{\log \hat{Y}|\log\sigma(s)}$ is unaffordable ($n \uparrow\uparrow$).
- Solution: *Stochastic Collocation Monte Carlo* (SCMC) sampler [5].

$$y_n|v_n \approx g_{L_{\hat{Y}},L_\sigma}(x_n) = \sum_{i=1}^{L_{\hat{Y}}} \sum_{j=1}^{L_\sigma} F^{-1}_{\log \hat{Y}|\log\sigma(s)=v_j}(F_X(x_i))\ell_i(x_n)\ell_j(v_n),$$

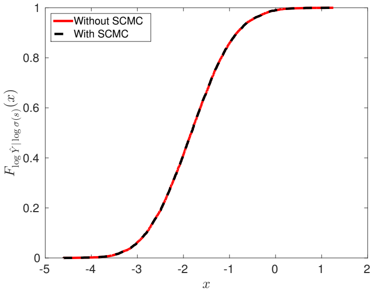
where x_n are the samples from the *cheap variable*, X, and v_n the given samples of $\log\sigma(s)$. x_i and v_j are the *collocation points* of X and $\log\sigma(s)$, respectively. ℓ_i and ℓ_j are the Lagrange polynomials defined by

$$\ell_i(x_n) = \prod_{k=1,k\neq i}^{L_{\hat{Y}}} \frac{x_n - x_k}{x_i - x_k}, \quad \ell_j(v_n) = \prod_{k=1,k\neq j}^{L_\sigma} \frac{v_n - v_k}{v_i - v_k}.$$

Application of 2D SCMC to $F_{\log \hat{Y} | \log \sigma(s)}$



(a) $\log \hat{Y} | \log \sigma(s)$ - DI. vs. SCMC.



(b) $F_{\log \hat{Y} | \log \sigma(s)}(x)$.

Samples	Without SCMC	With SCMC		
		$L_{\hat{Y}} = L_\sigma = 3$	$L_{\hat{Y}} = L_\sigma = 7$	$L_{\hat{Y}} = L_\sigma = 11$
100	1.0695	0.0449	0.0466	0.0660
10000	16.3483	0.0518	0.0588	0.0798
1000000	1624.3019	0.2648	0.5882	1.0940

Table: SCMC time

mSABR method - Experiments

- The strike values K_i are chosen following the expression:

$$K_i(T) = S(0)\exp(0.1 \times T \times \delta_i),$$
$$\delta_i = -1.5, -1.0, -0.5, 0.0, 0.5, 1.0, 1.5.$$

- Forward asset, $S(t)$: enhanced inversion by Chen et al. [3].
- Martingale correction:

$$S(t) = S(t) - \frac{1}{n}\sum_{i=1}^{n} S_i(t) + S_0,$$

- New data sets:

	S_0	σ_0	α	β	ρ	T
Set I [5]	0.5	0.5	0.4	0.5	0.0	4
Set II [3]	0.04	0.2	0.3	1.0	−0.5	5
Set III [1]	1.0	0.25	0.3	0.6	−0.5	20
Set IV [2]	0.0056	0.011	1.080	0.167	0.999	1

Table: Data sets.

mSABR method - Convergence test I

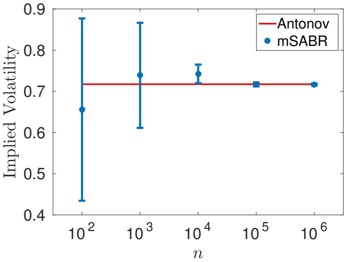
- Convergence in number of time-steps, m: Antonov vs. mSABR. Set I.

Strikes	K_1	K_2	K_3	K_4	K_5	K_6	K_7
Antonov	73.34%	71.73%	70.17%	N/A	67.23%	65.87%	64.59%
$m = T/4$	73.13%	71.75%	70.41%	69.11%	67.85%	66.64%	65.48%
Error(bp)	-21.51	2.54	24.38	N/A	61.71	76.66	89.26
$m = T/2$	73.30%	71.78%	70.29%	68.86%	67.49%	66.17%	64.93%
Error(bp)	-4.12	4.94	12.71	N/A	25.48	30.40	34.73
$m = T$	73.25%	71.67%	70.14%	68.66%	67.24%	65.89%	64.62%
Error(bp)	-9.56	-5.93	-2.79	N/A	0.92	2.21	3.17
$m = 2T$	73.32%	71.71%	70.16%	68.65%	67.22%	65.85%	64.55%
Error(bp)	-2.08	-1.56	-1.20	N/A	-1.65	-2.35	-3.36
$m = 4T$	73.34%	71.73%	70.18%	68.67%	67.24%	65.87%	64.58%
Error(bp)	0.15	0.58	0.78	N/A	0.43	0.04	-0.48

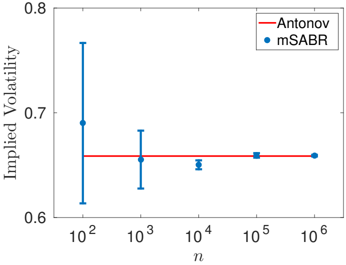
mSABR method - Convergence test II

- Convergence in number of samples, n: Antonov vs. mSABR. Set I.

Strikes	K_1	K_2	K_3	K_4	K_5	K_6	K_7
Antonov	73.34%	71.73%	70.17%	N/A	67.23%	65.87%	64.59%
$n = 10^2$	67.29%	65.55%	63.84%	62.20%	60.63%	59.01%	57.65%
RE	8.24×10^{-2}	8.61×10^{-2}	9.01×10^{-2}	N/A	9.82×10^{-2}	1.04×10^{-1}	1.07×10^{-1}
$n = 10^4$	73.41%	71.87%	70.36%	68.91%	67.51%	66.19%	64.94%
RE	9.65×10^{-4}	1.94×10^{-3}	2.75×10^{-3}	N/A	4.08×10^{-3}	4.93×10^{-3}	5.48×10^{-3}
$n = 10^6$	73.34%	71.73%	70.18%	68.67%	67.24%	65.87%	64.58%
RE	2.04×10^{-5}	8.08×10^{-5}	1.11×10^{-4}	N/A	6.39×10^{-5}	6.07×10^{-6}	7.43×10^{-5}



(c) Strike K_2.



(d) Strike K_6.

mSABR method - Stability in ρ

- Implied volatility, varying ρ: Monte Carlo (MC) vs. mSABR. Set II.

Strikes	K_1	K_2	K_3	K_4	K_5	K_6	K_7
				$\rho = -0.5$			
MC	22.17%	21.25%	20.38%	19.57%	18.88%	18.33%	17.95%
mSABR	22.21%	21.28%	20.39%	19.58%	18.88%	18.32%	17.94%
Error(bp)	3.59	2.86	1.78	0.95	-0.19	-0.96	-1.10
				$\rho = 0.0$			
MC	21.35%	20.96%	20.71%	20.63%	20.71%	20.96%	21.34%
mSABR	21.35%	20.95%	20.69%	20.60%	20.68%	20.93%	21.32%
Error(bp)	0.04	-1.04	-2.51	-3.02	-3.33	-3.19	-2.56
				$\rho = 0.5$			
MC	19.66%	20.04%	20.61%	21.34%	22.20%	23.14%	24.16%
mSABR	19.59%	19.96%	20.54%	21.28%	22.15%	23.11%	24.11%
Error(bp)	-6.93	-7.36	-6.77	-5.53	-4.35	-3.76	-4.05

mSABR method - Performance

- But, is it worth to use the mSABR method?

Error	< 100 bp	< 50 bp	< 25 bp	< 10 bp
MC Euler	6.85(200)	10.71(300)	27.42(800)	42.90(1200)
Y-Euler	2.18(4)	6.55(16)	11.85(32)	45.12(128)
Y-trpz	2.17(3)	4.24(8)	7.25(16)	14.47(32)
mSABR	3.46(1)	2.98(2)	3.72(3)	4.89(4)

Table: Execution times and time-steps, m (parentheses).

Error	< 100 bp	< 50 bp	< 25 bp	< 10 bp
MC Euler	1.98	3.59	7.37	8.77
Y-Euler	0.63	2.19	3.18	9.22
Y-trpz	0.62	1.42	1.94	2.95

Table: Speedups provided by the mSABR method.

mSABR method - Pricing barrier options

- The *up-and-out* call option is considered here
- The price, with the barrier level, B, $B > S_0, B > K_i$, reads:

$$V_i(K_i, B, T) = \exp(-rT)\,\mathbb{E}\left[(S(T) - K_i)\mathbb{1}(\max_{0<t_k\leq T} S(t_k) > B)\right],$$

where t_k are the times where the barrier condition is checked.
- Setting: $n = 10^6$ and $m = 4T$.
- We define the *mean squared error* (MSE) as

$$\text{MSE} = \frac{1}{7}\sum_{i=1}^{7}\left(V_i^{MC}(K_i, B, T) - V_i^{mSABR}(K_i, B, T)\right)^2$$

where $V_i^{MC}(K_i, B, T)$ and $V_i^{mSABR}(K_i, B, T)$ are the barrier option prices provided by standard Monte Carlo method and by the mSABR method, respectively.

mSABR method - Pricing barrier options

- Pricing barrier options with mSABR: $V_i(K_i, B, T) \times 100$. Set II:

Strikes	K_1	K_2	K_3	K_4	K_5	K_6	K_7
				$B = 0.08$			
MC	1.1702	0.9465	0.7268	0.5215	0.3423	0.1996	0.0987
mSABR	1.1724	0.9486	0.7285	0.5226	0.3428	0.1997	0.0986
MSE	1.8910×10^{-10}						
				$B = 0.1$			
MC	1.3099	1.0766	0.8462	0.6290	0.4367	0.2794	0.1626
mSABR	1.3092	1.0761	0.8456	0.6282	0.4355	0.2782	0.1618
MSE	7.5542×10^{-11}						
				$B = 0.12$			
MC	1.3521	1.1168	0.8841	0.6644	0.4695	0.3093	0.1891
mSABR	1.3518	1.1166	0.8838	0.6639	0.4686	0.3080	0.1880
MSE	6.3648×10^{-11}						

- Pricing barrier options with mSABR: $V_i(K_i, B, T) \times 100$. Set III:

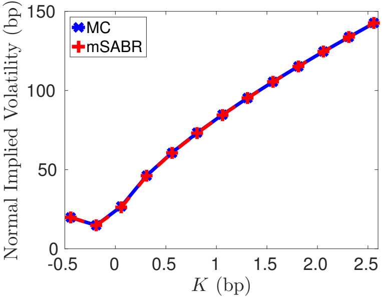
Strikes	K_1	K_2	K_3	K_4	K_5	K_6	K_7
				$B = 2.0$			
MC	29.1174	23.4804	17.2273	10.7825	5.0203	1.1750	0.0036
mSABR	29.2346	23.5828	17.3086	10.8327	5.0385	1.1805	0.0036
MSE	4.8146×10^{-7}						
				$B = 2.5$			
MC	41.3833	34.5497	26.8311	18.6089	10.7281	4.4893	0.9434
mSABR	41.3394	34.5097	26.7948	18.5747	10.6943	4.4546	0.9320
MSE	1.2131×10^{-7}						
				$B = 3.0$			
MC	48.5254	41.1652	32.7980	23.7807	14.9344	7.5364	2.6692
mSABR	48.5008	41.1515	32.7888	23.7655	14.9097	7.5117	2.6549
MSE	3.6201×10^{-8}						

mSABR method - Negative interest rates

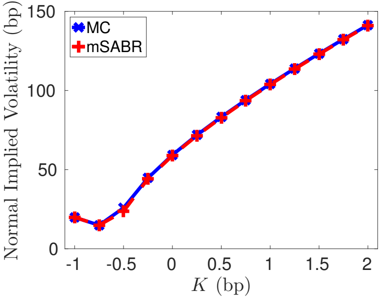
- The mSABR method in combination with the shifted SABR model:

$$\mathrm{d}S(t) = \sigma(t)(S(t)+\theta)^{\beta}\mathrm{d}W_S(t),$$
$$S(0) = (S_0+\theta)\exp(rT),$$

  where $\theta > 0$ is a displacement, or shift, in the underlying.
- Setting: $n = 10^6$, $m = 4T$ and $\theta = 0.02$.



(e) Set IV



(f) Set IV: $S_0 = 0$

Conclusions

- We propose an efficient SABR simulation based on Fourier and copula techniques.
- The one-step SABR is a fast alternative to Hagan formula for short maturities.
- Overcomes the known issues of Hagan's expression.
- When longer maturities and/or more involved options are considered, multi-step version.
- High accuracy with very few number of time-steps, even in the context of negative interest rates.
- Good balance between accuracy and computational cost.

Álvaro Leitao, Lech A. Grzelak, and Cornelis W. Oosterlee.
On a one time-step Monte Carlo simulation approach of the SABR model: application to European options. *Applied Mathematics and Computation*, 293:461–479, 2017.

Álvaro Leitao, Lech A. Grzelak, and Cornelis W. Oosterlee.
On an efficient multiple time-step Monte Carlo simulation of the SABR model. *Quantitative Finance*, 2017.

References

Alexandre Antonov, Michael Konikov, and Michael Spector.
SABR spreads its wings.
Risk Magazine, pages 58–63, August 2013.

Alexandre Antonov, Michael Konikov, and Michael Spector.
The free boundary SABR: natural extension to negative rates.
Risk Magazine, pages 58–63, August 2015.

Bin Chen, Cornelis W. Oosterlee, and Hans van der Weide.
A low-bias simulation scheme for the SABR stochastic volatility model.
International Journal of Theoretical and Applied Finance, 15(2):1250016–1 – 1250016–37, 2012.

Fang Fang and Cornelis W. Oosterlee.
A novel pricing method for European options based on Fourier-cosine series expansions.
SIAM Journal on Scientific Computing, 31:826–848, 2008.

Lech A. Grzelak, Jeroen A. S. Witteveen, M. Suárez-Taboada, and Cornelis W. Oosterlee.
The Stochastic Collocation Monte Carlo sampler: highly efficient sampling from "expensive" distributions, 2015.
Available at SSRN: https://ssrn.com/abstract=2529691.

Patrick S. Hagan, Deep Kumar, Andrew S. Lesniewski, and Diana E. Woodward.
Managing smile risk.
Wilmott Magazine, pages 84–108, 2002.

Othmane Islah.
Solving SABR in exact form and unifying it with LIBOR market model, 2009.
Available at SSRN: http://ssrn.com/abstract=1489428.

Ralf Korn and Songyin Tang.
Exact analytical solution for the normal SABR model.
Wilmott Magazine, 2013(66):64–69, 2013.

Acknowledgments

This research was supported by the European Union in the FP7-PEOPLE-2012-ITN Program under Grant Agreement Number 304617 (FP7 Marie Curie Action, Project Multi-ITN STRIKE – Novel Methods in Computational Finance).

Thank you for your attention