BENCHOP–SLV: The BENCHmarking project in Option Pricing – Stochastic and Local Volatility problems

Lina von Sydow[a,*], Slobodan Milovanović[a], Elisabeth Larsson[a], Karel In 't Hout[b], Magnus Wiktorsson[c], Cornelis W. Oosterlee[d,e], Victor Shcherbakov[a], Maarten Wyns[b], Alvaro Leitao[d,f], Shashi Jain[d,g], Tinne Haentjens[b], and Johan Waldén[h]

[a]Department of Information Technology, Uppsala University, Sweden; [b]Department of Mathematics and Computer Science, University of Antwerp, Belgium; [c]Centre for Mathematical Sciences, Lund University, Sweden; [d]Centrum Wiskunde & Informatica, Amsterdam, The Netherlands; [e]Delft Institute of Applied Mathematics, Delft University of Technology, The Netherlands; [f]Barcelona Graduate School of Mathematics, Barcelona, Spain; [g]Department of Management Studies, Indian Institute Of Science, Bangalore;[h]University of California Berkeley, Haas School of Business, USA

ARTICLE HISTORY
Compiled July 18, 2018

ABSTRACT

In the recent project BENCHOP – the BENCHmarking project in Option Pricing we found that Stochastic and Local Volatility problems were particularly challenging. Here we continue the effort by introducing a set of benchmark problems for this type of problems. Eight different methods targeted for the Stochastic Differential Equation (SDE) formulation and the Partial Differential Equation (PDE) formulation of the problem, as well as Fourier methods making use of the characteristic function were implemented to solve these problems. Comparisons are made with respect to CPU-time to reach a certain error level in the computed solution for the different methods.

The implemented Fourier method was superior to all others for the two problems where it was implemented. Generally, methods targeting the PDE formulation of the problem outperformed the methods for the SDE formulation. Among the methods for the PDE formulation the ADI method stood out as the best performing one.

KEYWORDS
Option pricing; stochastic and local volatility; numerical methods; benchmark problem; stochastic differential equation; partial differential equation; characteristic function

AMS CLASSIFICATION
65-02; 91G60; 91G20

1. Introduction

The aim of the original BENCHOP project described in [34] was to provide researchers and practitioners in finance with a set of common benchmark problems for comparisons between methods and for evaluation of new methods. A wide range of existing

CONTACT Lina von Sydow, Email: lina@it.uu.se

[*]The contribution from each author is presented in Appendix B

numerical methods for each benchmark problem was implemented and the relative performance of the methods was compared. The problems were selected with respect to features that may be numerically challenging such as early exercise properties, barriers, discrete dividends, local volatility, stochastic volatility, jump diffusion, and two underlying assets.

In [34] we found that stochastic and local volatility problems were particularly challenging. Hence, we here continue the effort to provide benchmark problems and performance comparisons between numerical methods at the research frontier for this type of problems. Stochastic and local volatility models provide an alternative approach to jump-diffusion models, and have proven effective in matching the rich asset price structure observed in derivatives markets. As is the case for jump diffusion models, the market is typically incomplete in stochastic volatility models (in absence of assets that are directly traded on volatility risk), so additional assumptions are needed to uniquely determine asset prices. In [19], for example, it is assumed that the volatility risk-premium is proportional to the volatility level.

We present the benchmark problems with sufficient detail so that others can solve them in the future. We also provide analytical solutions where such are available. Each problem is solved using MATLAB implementations of a number of numerical methods, and error plots as a function of CPU-time are provided. For details of the methods, we refer to the original papers, listed in Table 1. The codes are not fully optimized, and the numerical results should not be interpreted as competition scores. We rather see it as a general indication of their performance in terms of obtained accuracy versus CPU-time that could be expected in the computing environment under consideration. In order to facilitate future comparisons, MATLAB p-codes of the here presented methods for each benchmark problem will be made available through the BENCHOP web site `http://www.it.uu.se/research/scientific_computing/project/compfin/benchop`.

The paper is outlined as follows. In Section 2 we state the benchmark problems while the numerical methods are briefly presented in Section 3. Section 4 is dedicated to the presentation of the numerical results and finally in Section 5 we discuss the results. In Appendix A we present how the reference values are computed for the different problems. Finally, in Appendix B we specify the contributions from the authors of the paper.

2. Benchmark problems

2.1. Problem 1 — SABR stochastic-local volatility model

The Stochastic Alpha Beta Rho (SABR) model [18] is an established Stochastic Differential Equation (SDE) system which is often used for interest rates and FX modeling in practice. A key feature of the model is that it matches the observed dynamic behavior of the volatility smile, namely that when the price of the underlying decreases, the volatility smile shifts to lower prices, and vice versa. The SABR model is based on a parametric local volatility component in terms of a model parameter, β. The formal definition of the SABR model reads

$$\begin{cases} \mathrm{d}F_t = \sigma_t F_t^{\beta} \mathrm{d}W_t^F, \\ \mathrm{d}\sigma_t = \alpha \sigma_t \mathrm{d}W_t^{\sigma}, \end{cases}$$

where $F_t = S_t \exp\left(r(T-t)\right)$ denotes the forward value of the underlying asset S_t, with r the interest rate, S_0 the spot price and T the contract's final time. The quantity σ_t denotes the stochastic volatility, W_t^F and W_t^σ are two correlated Brownian motions with constant correlation coefficient ρ (i.e. $W_t^F W_t^\sigma = \rho t$). The free model parameters are $\alpha > 0$ (the volatility of the volatility), $0 \leq \beta \leq 1$ (the elasticity) and ρ (the correlation coefficient). The corresponding Partial Differential Equation (PDE) for the valuation of options is given by:

$$\frac{\partial u}{\partial t} + \frac{1}{2}\sigma^2 s^{2\beta}\frac{\partial^2 u}{\partial s^2} + \rho\alpha s^\beta \sigma^2 \frac{\partial^2 u}{\partial\sigma\partial s} + \frac{1}{2}\alpha^2\sigma^2\frac{\partial^2 u}{\partial \sigma^2} - ru = 0$$

for $s > 0$, $\sigma > 0$ and $0 \leq t < T$.

Deliverables

The problem should be solved for a European call option with payoff $\max(S-K, 0)$ at $t = T$, with three strikes

$$\begin{aligned} K &= S_0 \exp(0.1 \times \sqrt{T} \times \delta), \\ \delta &= -1.0, 0.0, 1.0. \end{aligned}$$

Parameter and problem specifications

- Set I ([16]): $T = 2$, $r = 0$, $S_0 = 0.5$, $\sigma_0 = 0.5$, $\alpha = 0.4$, $\beta = 0.5$ and $\rho = 0$.
- Set II ([7]): $T = 10$, $r = 0$, $S_0 = 0.07$, $\sigma_0 = 0.4$, $\alpha = 0.8$, $\beta = 0.5$ and $\rho = -0.6$.

2.2. Problem 2 — Quadratic local stochastic volatility model

The Quadratic Local Stochastic Volatility (QLSV) model, can be viewed as a generalization of Heston's stochastic volatility model [19]. In the QLSV model, the square root term is multiplied by a quadratic function in the underlying asset. When the function is a constant, Heston's original model is obtained. The additional degrees of freedom provided by the quadratic function allows for improved volatility surface calibration.

We use the formulation in, e.g., [26] and define the following SDE:

$$\begin{cases} \mathrm{d}S_t &= rS_t \,\mathrm{d}t + \sqrt{V_t}\, f(S_t)\,\mathrm{d}W_t^1, \\ \mathrm{d}V_t &= \kappa(\eta - V_t)\,\mathrm{d}t + \sigma\sqrt{V_t}\,\mathrm{d}W_t^2, \end{cases}$$

with $f(s) = \frac{1}{2}\alpha s^2 + \beta s + \gamma$. The PDE for the valuation of options is then given by:

$$\frac{\partial u}{\partial t} + \tfrac{1}{2} f(s)^2 v \frac{\partial^2 u}{\partial s^2} + \rho\sigma f(s) v \frac{\partial^2 u}{\partial s \partial v} + \tfrac{1}{2}\sigma^2 v \frac{\partial^2 u}{\partial v^2} + rs\frac{\partial u}{\partial s} + \kappa(\eta - v)\frac{\partial u}{\partial v} - ru = 0,$$

for $s > 0$, $v > 0$ and $0 \leq t < T$.

We consider a case when the Feller condition, $2\kappa\eta \geq \sigma^2$ is not satisfied (as is often the case in practice). The Feller condition [9] ensures that the volatility process,

V_t, remains strictly positive. If the condition is violated, the process may reach the boundary $V_t = 0$. In this case, additional specification of the behavior at the boundary is needed to ensure (weak) positivity, and the problem is also numerically more challenging (see [8], [27]).

Deliverables

The problem should be solved for

- a European call option with payoff $\max(S - K, 0)$ and $K = 100$.
- a Double-no-touch option paying 1 if $L < S_t < U$ (for all t) and 0 else with $L = 50$, $U = 150$.

and three spot values: $(S_0, V_0) = (S_0, 0.114)$ for $S_0 = 75, 100, 125$.

Parameter and problem specifications

We consider

- a Heston model with $\alpha = 0$, $\beta = 1$, $\gamma = 0$,
- a QLSV model with $\alpha = 0.02$, $\beta = 0$, $\gamma = 0$.

For both models we use the parameter set (see [26]):

$$T = 1,\ r = 0,\ \kappa = 2.58,\ \eta = 0.043,\ \sigma = 1,\ \rho = -0.36.$$

2.3. Problem 3 — Heston–Hull–White model

The Heston–Hull–White (HHW) model is a hybrid asset price model combining the Heston stochastic volatility model, and Hull–White stochastic interest rate model, see e.g. [15, 17]. With such an approach the skew pattern for equity can be matched, while still allowing for stochastic interest rates. We define the HHW SDE:

$$\begin{cases} \mathrm{d}S_t &= R_t S_t \,\mathrm{d}t + \sqrt{V_t}\, S_t \,\mathrm{d}W_t^1, \\ \mathrm{d}V_t &= \kappa(\eta - V_t)\,\mathrm{d}t + \sigma_1\sqrt{V_t}\,\mathrm{d}W_t^2, \\ \mathrm{d}R_t &= a(b(t) - R_t)\,\mathrm{d}t + \sigma_2\,\mathrm{d}W_t^3, \end{cases}$$

and the corresponding HHW PDE:

$$\begin{aligned} \frac{\partial u}{\partial t} &+ \tfrac{1}{2}s^2 v\frac{\partial^2 u}{\partial s^2} + \tfrac{1}{2}\sigma_1^2 v\frac{\partial^2 u}{\partial v^2} + \tfrac{1}{2}\sigma_2^2\frac{\partial^2 u}{\partial r^2} \\ &+ \rho_{12}\sigma_1 s v\frac{\partial^2 u}{\partial s \partial v} + \rho_{13}\sigma_2 s\sqrt{v}\frac{\partial^2 u}{\partial s \partial r} + \rho_{23}\sigma_1\sigma_2\sqrt{v}\frac{\partial^2 u}{\partial v \partial r} \\ &+ rs\frac{\partial u}{\partial s} + \kappa(\eta - v)\frac{\partial u}{\partial v} + a(b(t) - r)\frac{\partial u}{\partial r} - ru = 0, \end{aligned}$$

for $s > 0$, $v > 0$, and $0 \leq t < T$. Again, we will consider a case when the Feller condition is violated, see Section 2.2.

Deliverables

The problem should be solved for a European call option with payoff $\max(S - K, 0)$, $K = 100$, and three spot values: $(S_0, V_0, R_0) = (S_0, 0.04, 0.10)$ for $S_0 = 75, 100, 125$.

Parameter and problem specifications

We use the parameter set (cf. [2, 17]):

$$T = 10,\ \kappa = 0.5,\ \eta = 0.04,\ \sigma_1 = 1,\ \sigma_2 = 0.09,\ \rho_{12} = -0.9,\ \rho_{13} = 0,\\ \rho_{23} = 0,\ a = 0.08,\ \text{and } b(t) \equiv 0.10.$$

3. Numerical methods

In Table 1 we display all methods that we have used. The methods are developed from the SDE or PDE formulation of the problem respectively, or make use of the characteristic function for the Fourier method. The methods are organized according to this in the table. To have a uniform presentation in Section 4 we use the same abbreviation and marker symbol for a given method in all figures in this section. Finally, Table 1 provides references to the original papers describing the methods and also indicates whether the implementation is employing parallel constructions provided in Matlab Parallel Computing Toolbox or not.

Table 1. List of methods used with abbreviations, marker symbol used in figures, references, and whether the MATLAB-implementation makes use of parallellism

Abbr.	**Symbol**	**Method**	**References**	**//-ism**
		Methods for SDE formulation		
MCA	-▲-	Monte Carlo simulation with Control and Antithetic variables	[30, 13, 4]	Yes
MLMC	-■-	Multi Level Monte Carlo	[12]	No
SGBM	-●-	Stochastic Grid Bundling Method	[22, 10]	Yes
mSABR	-◆-	The multi-step Monte Carlo simulation of the SABR model	[24]	Yes
		Fourier method		
FGL	—☆—	Fourier method with Gauss-Laguerre quadrature	[25, 23, 14, 1, 6, 5]	No
		Methods for PDE formulation		
ADI	-✱-	Alternating Direction Implicit Finite Difference method	[17, 21, 20, 36]	No
RBFFD	-×-	Radial Basis Function generated Finite Differences	[11, 3, 29, 28]	Yes
RBFPUM	-+-	Radial Basis Function Partition of Unity Methods	[31, 33, 32, 28]	No

4. Numerical results

In this section we present the results using the different methods presented in Section 3 for the problems defined in Section 2. For all problems we display errors in the computed solution as a function of CPU-time. The implementations were made in MATLAB and encrypted into p-code. The experiments were carried out at the Rackham Cluster at Uppsala Multidisciplinary Center for Advanced Computational Science

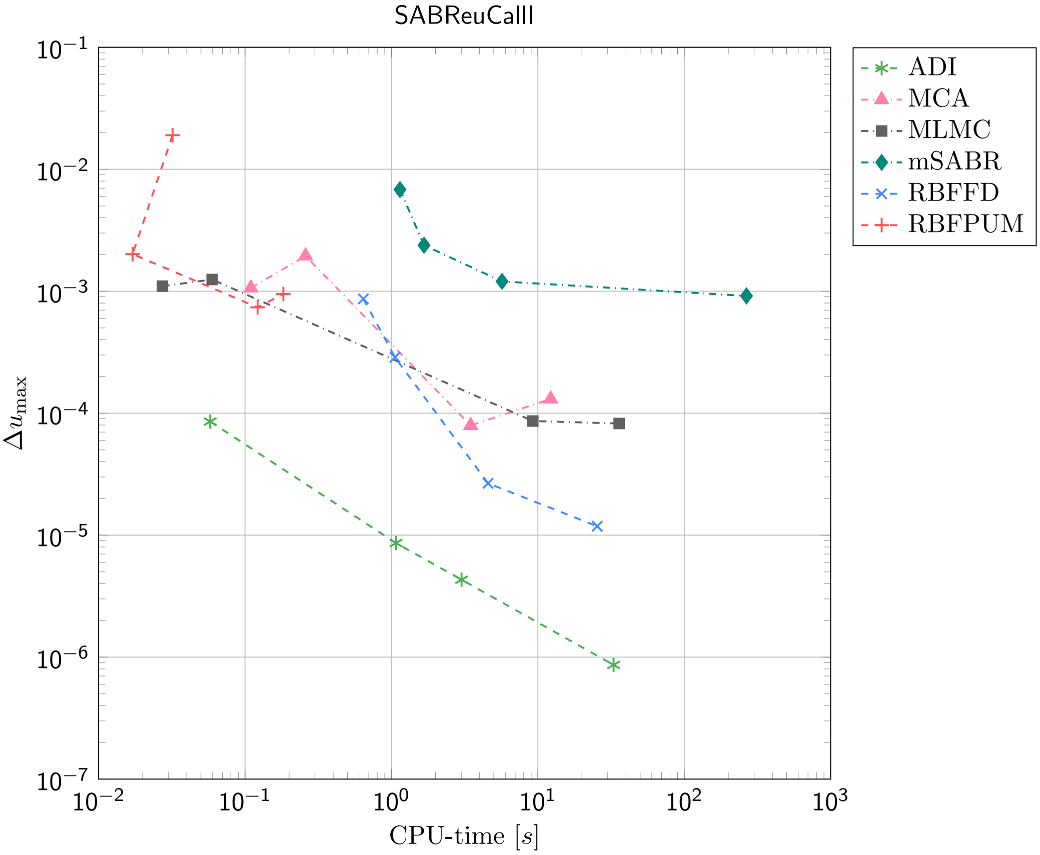

Figure 1. Results for the European call option under the SABR model, Parameter Set I. The reference values for $K_i = S_0\exp(0.1 \times \sqrt{T} \times \delta_i)$, $\delta_i = -1.0, 0.0, 1.0$ are given by 0.221383196830866, 0.193836689413803, and 0.166240814653231.

at Uppsala University http://www.uppmax.uu.se/resources/systems/the-rackham-cluster/. The Rackham Cluster consists of 334 nodes where each node has two 10-core Intel Xeon V4 CPUs and at least 128 GB RAM memory. The experiments were performed within one node, i.e. using up to 20 MATLAB-workers. For the methods based on the SDE formulation of the problems, i.e. a Monte Carlo based solution method, the code was run three times and the mean of the results were reported. Finally, note that the axes in the figures are different for the different problems.

4.1. Results for SABR model

In Figures 1 and 2 we display the error

$$\Delta u_{\max} = \max_K |u(S_0, \sigma_0, 0) - u_{\text{ref}}(S_0, \sigma_0, 0)|,$$

where u denotes the computed solution and u_{ref} the reference solution, as a function of CPU-time for the SABR stochastic-local volatility model. Figure 1 shows the results for Parameter Set I, and Figure 2 the results for Parameter Set II. For both parameter sets, the ADI method is the most favorable one to use – for a given CPU-time the error is always smallest for this method. Among the Monte Carlo based methods, mSABR is the least favorable one to use for Parameter Set I while it is quite favorable to use

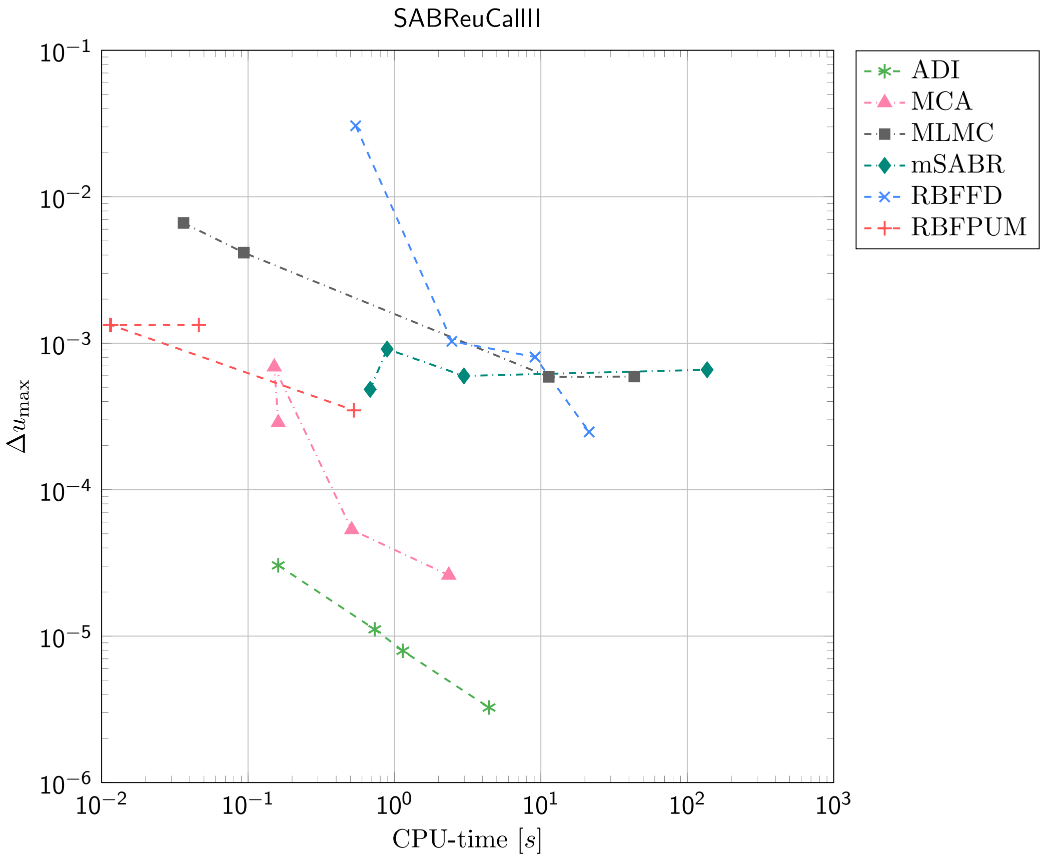

Figure 2. Results for the European call option under the SABR model, Parameter Set II. The reference values for $K_i = S_0 \exp(0.1 \times \sqrt{T} \times \delta_i)$, $\delta_i = -1.0, 0.0, 1.0$ are given by 0.052450313614407, 0.046585753491306, and 0.039291470612989.

for Parameter Set II if the required error in the solution is not too small (slightly less than 10^{-3}). For Parameter Set I MCA and MLMC behaves similarly while MCA performs better than MLMC for Parameter Set II. For the RBF methods, RBFPUM performs better than RBFFD for relatively large errors ($\sim 10^{-3}$). However, especially for Parameter Set I, RBFFD can provide smaller errors than RBFPUM does. For the SABR model, FGL was not implemented.

4.2. Results for QLSV model

The error

$$\Delta u_{\max} = \max_{S_0} |u(S_0, V_0, 0) - u_{\text{ref}}(S_0, V_0, 0)|, \tag{4.1}$$

as a function of CPU-time for the European call option under the Heston model is presented in Figure 3, and the Double-no-touch option under the same model in Figure 4. For the European call option, FGL outperforms all other methods. Already in less than 0.1 seconds, the error is below 10^{-7} for this method. The second best method is ADI which is also the method that is performing best for the Double-no-touch option. When it comes to the two RBF-methods, RBFFD is the method of choice for this problem. For the European call option, the error for RBFPUM stays slightly below

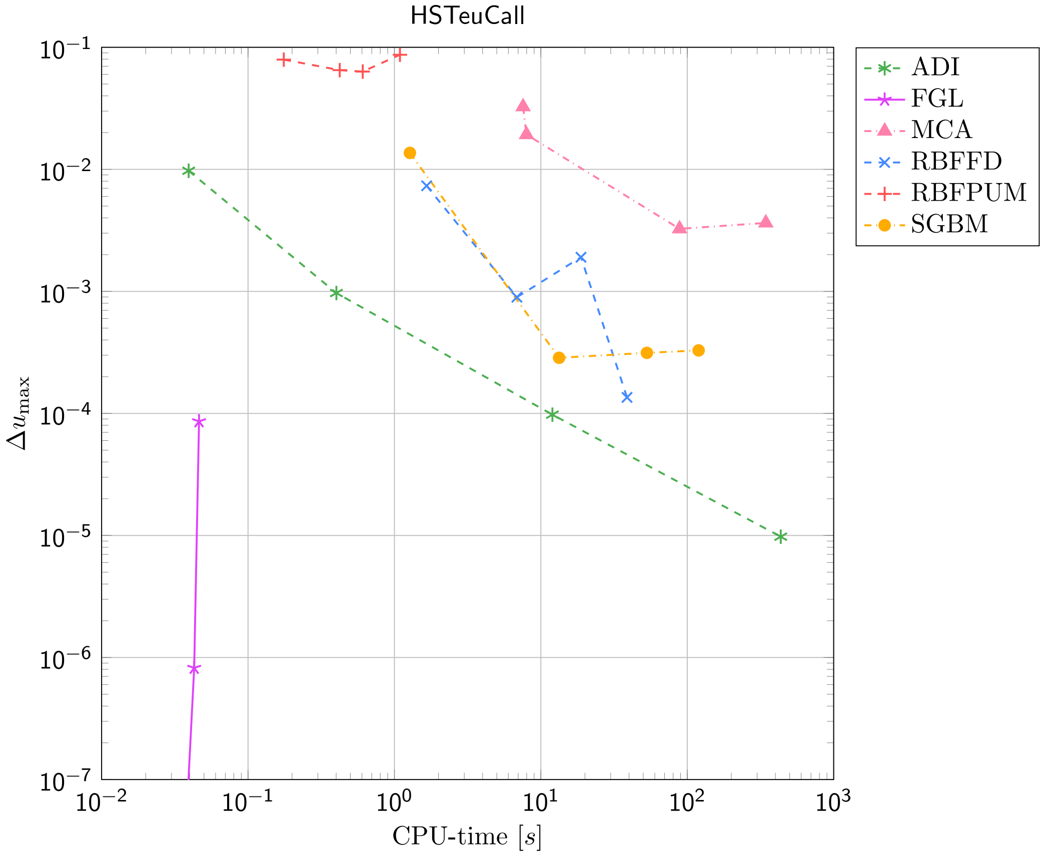

Figure 3. Results for the European call option under the Heston model. The reference values for $S_0 = 75$, 100, and 125 are given by 0.908502728459621, 9.046650119220969, and 28.514786399298796.

10^{-1} and for the Double-no-touch option it stays around 10^{-2}. RBFFD on the other provides an error that is slightly larger than 10^{-4} in less than 100 seconds. The SGBM method is the best performing Monte Carlo type method for the Heston model. Both for the European option and the Double-no-touch option the time to reach a certain accuracy for SGBM is well below the time for MCA. Finally, for the European option the performance of RBFFD and SGBM is similar, while RBFFD gives better results than SGBM for the Double-no-touch option.

In Figures 5 and 6 the results for the European and Double-no-touch options under the QLSV-model are presented. The Fourier method was not implemented for this model and the ADI-method performed best for both types of options this time. To obtain low accuracy (10^{-1} and 10^{-2} for European and Double-no-touch options respectively) the RBFPUM is the second best method that reaches these accuracies in less than 1 second. To obtain higher accuracy (10^{-3} and 10^{-4} respectively) RBFFD is preferable between the two RBF methods. For the Double-no-touch option MCA performs similarily as RBFFD while RBFFD outperforms MCA for the European call option.

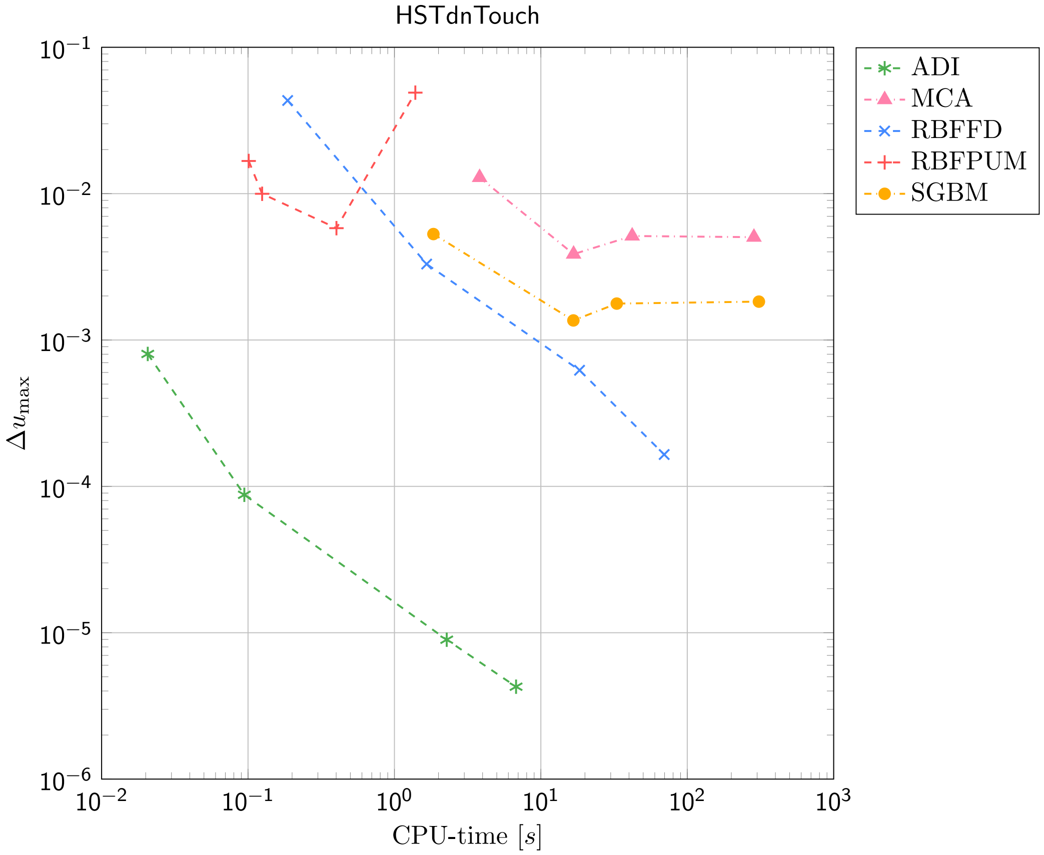

Figure 4. Results for the Double-no-touch option under the Heston model. The reference values for $S_0 = 75$, 100, and 125 are given by 0.834539127387590, 0.899829293208866, and 0.668399975738358.

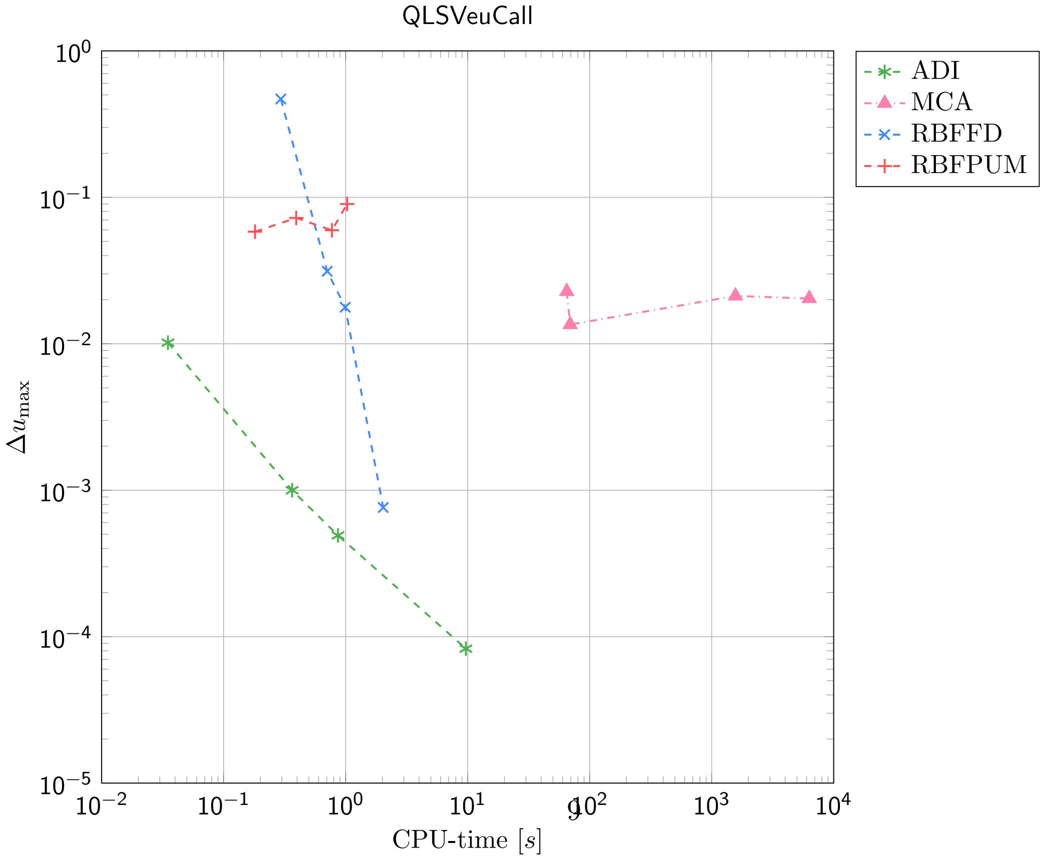

Figure 5. Results for the European call option under the QLSV model. The reference values for $S_0 = 75$, 100, and 125 are given by 0.527472759419533, 8.902347915743665, and 29.159828965633729.

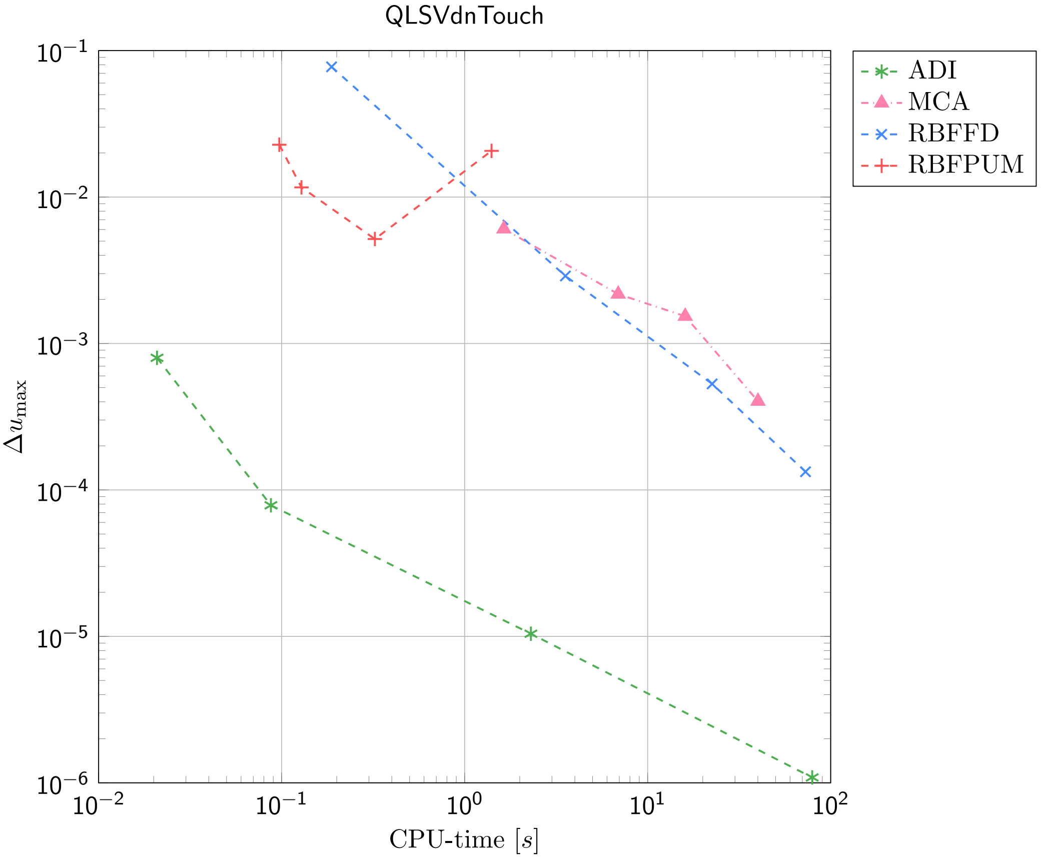

Figure 6. Results for the Double-no-touch option under the QLSV model. The reference values for $S_0 = 75$, 100, and 125 are given by 0.933800903110254, 0.914799140676374, and 0.592983062889906.

4.3. Results for HHW model

In Figure 7 we present the results from the Heston–Hull–White model which is the only three-dimensional model among our benchmark problems. The error

$$\Delta u_{\max} = \max_{S_0} |u(S_0, V_0, R_0, 0) - u_{\text{ref}}(S_0, V_0, R_0, 0)|, \tag{4.2}$$

as a function of CPU-time is presented. Again, FGL outperforms all other methods and reaches $\Delta u_{\max} < 10^{-7}$ in less than 0.1 seconds. ADI shows the same robust convergence behaviour as in the previous cases but this time MCA seems to perform equally well when considering accuracy versus CPU-time. It should be noted that the ADI code is a serial code while MCA is parallelized. Further, for MCA confidence intervals are not given. Finally, RBFPUM gives a fairly large error in this experiment.

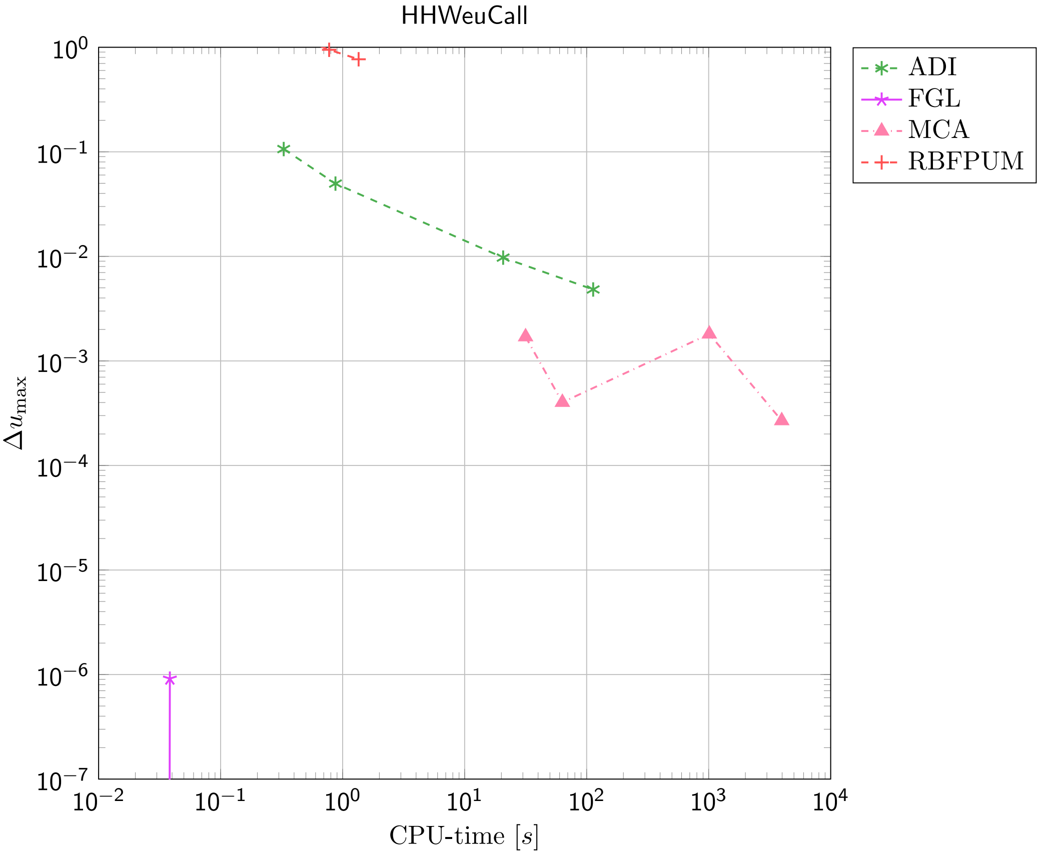

Figure 7. Results for the European call option under the HHW model. The reference values for $S_0 = 75$, 100, and 125 are given by 35.437896876285350, 54.728065308229503, and 75.397596834993621.

5. Discussion

In this paper we have defined a set of benchmarking problems in option pricing for stochastic and local volatility problems. Eight different numerical methods have been implemented and compared with respect to error versus CPU-time.

The Fourier method FGL was implemented only for the European call option with the Heston model and the Heston–Hull–White model. In both cases this method was superior to all others and reached extremely high accuracy with an error less than 10^{-7} in less than 0.1 seconds. However, Fourier methods rely on the availability of the characteristic function of the underlying stochastic process or efficient approximations of the same.

For the problems when FGL was not implemented, the most efficient method was the ADI method. This was also together with RBFPUM and MCA the only method that was implemented for all problems. For the RBF methods, RBFPUM often reached a reasonable accuracy in a short time but had troubles with convergence with increasing CPU-time. RBFFD however, was in many cases the method that reached the smallest error after ADI (and FGL where applicable). The main challenge for the RBF methods was to apply appropriate boundary conditions in the volatility and interest rate dimensions. Some commonly used conditions introduced bias, while the option of leaving the boundary open in some cases lead to instability or inaccuracy.

For the methods based on the SDE formulation of the problems it is difficult to draw any far-reaching conclusions. MLMC performed reasonably well for the SABR problems while mSABR gave reasonably small errors for Parameter Set II in a fairly short time. The mSABR method requires a smart application of a stochastic collocation-based method (see [16]) to make the algorithm affordable in terms of computational time. Once the so-called *integrated variance* and volatility processes are simulated, the SABR forward asset process has been simulated by an extended Log-Euler discretization scheme, called *Log-Euler+*. MCA also gave quite accurate results in a reasonable time, especially for the SABR problems. For the Heston model (the special case of QLSV), SGBM was the best performing Monte Carlo type method. For the 3D problem Heston–Hull–White, MCA and ADI showed comparable results in this setting. This indicates as expected that Monte Carlo methods probably will be more competitive in higher dimensions.

Acknowledgments

The computations were performed on resources provided by SNIC through Uppsala Multidisciplinary Center for Advanced Computational Science (UPPMAX) under Project SNIC 2017/1-236.

References

[1] M. Abramowitz and I. A. Stegun. Bessel functions j and y, § 9.1 in handbook of mathematical functions with formulas, graphs, and mathematical tables, 9th printing, 1972.

[2] L. Andersen. Simple and efficient simulation of the Heston stochastic volatility model. *The Journal of Computational Finance*, 11:1–42, 2008.

[3] V. Bayona, N. Flyer, B. Fornberg, and G. A. Barnett. On the role of polynomials in RBF-FD approximations: II. numerical solution of elliptic PDEs. *Journal of Computational Physics*, 332:257–273, 2017.

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

[5] M. Brodén, J. Holst, E. Lindström, J. Ströjby, and M. Wiktorsson. Sequential calibration of options. Technical report, Centre for Mathematical Sciences, Mathematical Statistics, Lund University, 2006. Contains additional material not included in the CSDA paper.

[6] P. Carr, H. Geman, D. Madan, L. Wu, and M. Yor. Option Pricing Using Integral Transforms, 2003. A slideshow presentation available at: http://www.math.nyu.edu/research/c arrp/papers/pdf/integtransform.pdf.

[7] B. Chen, C. W. Oosterlee, and H. 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.

[8] E. Ekström and J. Tysk. Boundary conditions for the single-factor term structure equation. *The Annals of Applied Probability*, pages 332–350, 2011.

[9] W. Feller. Two singular diffusion problems. *Annals of mathematics*, pages 173–182, 1951.

[10] Q. Feng and C. Oosterlee. Monte Carlo calculation of exposure profiles and Greeks for Bermudan and barrier options under the Heston Hull-White model.

In R. Melnik, R. Makarov, and J. Belair, editors, *Recent Progress and Modern Challenges in Applied Mathematics*, pages 265–301. Springer Fields Inst. Comm., 2017.

[11] N. Flyer, B. Fornberg, V. Bayona, and G. A. Barnett. On the role of polynomials in rbf-fd approximations: I. interpolation and accuracy. *Journal of Computational Physics*, 321:21–38, 2016.

[12] M. B. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56 (3):607–617, 2008.

[13] G. H. Givens and J. A. Hoeting. *Computational statistics*, volume 710. John Wiley & Sons, 2012.

[14] G. H. Golub and J. H. Welsch. Calculation of Gauss quadrature rules. *Mathematics of computation*, 23(106):221–230, 1969.

[15] L. A. Grzelak and C. W. Oosterlee. On the Heston model with stochastic interest rates. *SIAM Journal on Financial Mathematics*, 2:255 – 286, 2011.

[16] L. A. Grzelak, J. A. S. Witteveen, M. Suárez-Taboada, and C. W. Oosterlee. The Stochastic Collocation Monte Carlo Sampler: Highly efficient sampling from "expensive" distributions, 2015. URL `https://ssrn.com/abstract=2529691`. Available at SSRN: https://ssrn.com/abstract=2529691.

[17] T. Haentjens and K. J. in 't Hout. Alternating direction implicit finite difference schemes for the Heston–Hull–White partial differential equation. *The Journal of Computational Finance*, 16:83–110, 2012.

[18] P. S. Hagan, D. Kumar, A. S. Lesniewski, and D. E. Woodward. Managing smile risk. *Wilmott Magazine*, pages 84–108, 2002.

[19] S. L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 11:327–343, 1993.

[20] K. J. in 't Hout and S. Foulon. ADI finite difference schemes for option pricing in the Heston model with correlation. *Int. J. Numer. Anal. Model.*, 7(2):303–320, 2010.

[21] K. J. in 't Hout and B. D. Welfert. Unconditional stability of second-order ADI schemes applied to multi-dimensional diffusion equations with mixed derivative terms. *Appl. Numer. Math.*, 59(3-4):677–692, 2009.

[22] S. Jain and C. W. Oosterlee. The stochastic grid bundling method: Efficient pricing of Bermudan options and their Greeks. *Applied Mathematics and Computation*, 269:412–431, 2015.

[23] R. W. Lee et al. Option pricing by transform methods: extensions, unification and error control. *Journal of Computational Finance*, 7(3):51–86, 2004.

[24] A. Leitao, L. A. Grzelak, and C. W. Oosterlee. On an efficient multiple time step Monte Carlo simulation of the SABR model. *Quantitative Finance*, 17(10): 1549–1565, 2017. URL `https://doi.org/10.1080/14697688.2017.1301676`.

[25] E. Lindström, J. Ströjby, M. Brodén, M. Wiktorsson, and J. Holst. Sequential calibration of options. *Computational Statistics & Data Analysis*, 52(6):2877–2891, 2008.

[26] A. Lipton, A. Gal, and A. Lasis. Pricing of vanilla and first-generation exotic options in the local stochastic volatility framework: survey and new results. *Quantitative Finance*, 14:1899–1922, 2014.

[27] V. Lucic. Boundary conditions for computing densities in hybrid models via PDE methods. *Stochastics An International Journal of Probability and Stochastic Processes*, 84(5-6):705–718, 2012.

[28] S. Milovanović and V. Shcherbakov. Pricing derivatives under multiple stochastic

factors by localized radial basis function methods. arXiv:1711.09852 [q-fin.CP], 2017.

[29] S. Milovanović and L. von Sydow. Radial basis function generated finite differences for option pricing problems. *Computers & Mathematics with Applications*, 75(4):1462 –1481, 2018.

[30] K.-S. Moon. Efficient Monte Carlo algorithm for pricing barrier options. *Communications of the Korean Mathematical Society*, 23(2):285–294, 2008.

[31] A. Safdari-Vaighani, A. Heryudono, and E. Larsson. A radial basis function partition of unity collocation method for convection–diffusion equations arising in financial applications. *J. Sci. Comp.*, pages 1–27, 2014.

[32] V. Shcherbakov. Radial basis function partition of unity operator splitting method for pricing multi-asset American options. *BIT Num. Math.*, 56:1401–1423, 2016.

[33] V. Shcherbakov and E. Larsson. Radial basis function partition of unity methods for pricing vanilla basket options. *Comput. Math. Applic.*, 71:185–200, 2016.

[34] L. von Sydow, L. Josef Höök, E. Larsson, E. Lindström, S. Milovanović, J. Persson, V. Shcherbakov, Y. Shpolyanskiy, S. Sirén, J. Toivanen, et al. BENCHOP–the BENCHmarking project in Option Pricing. *International Journal of Computer Mathematics*, 92(12):2361–2379, 2015.

[35] M. Wiktorsson. SABR. Draft report, 2018.

[36] M. Wyns and K. J. in 't Hout. An adjoint method for the exact calibration of stochastic local volatility models. *J. Comp. Sci.*, 24:182–194, 2018.

Appendix A. The methods used for computing the reference values used in the comparisons

When there is no analytical formula available we have used the ADI-method with a very small tolerance to compute the reference values. The estimated error in these values are computed as the absolute difference between two subsequent solutions in a refinement sequence of ADI-solutions. As with all numerical methods, there could be some bias in this.

A.1. Methods for Problem 1—SABR

Case I

To our knowledge, no analytical formula allowing for highly accurate evaluation of the European call option under the SABR model was previously available in the literature. For the parameters we use in Case I, with uncorrelated Brownian motions W_S for the stock and W_σ for the volatility, a new price formula has been derived. The full details of the derivations are found in [35]. If we denote the European call option price at time t for the SABR model with strike K and maturity T by $\pi_C^E(t, S_t, T, K)$, we have that

$$\pi_C^E(0, S_0, T, K) \quad = \quad S_0(1 - F_S(K)) - K(1 - F_K(K)),$$

where the probabilities $F_S(K)$ and $F_K(K)$ are expressed through the following integrals

$$\begin{aligned} F_K(K) &= \frac{1}{\pi}\int_{-\infty}^{\infty}\int_0^{\infty} \Re\left(\frac{q(\alpha,\sigma_0,T,x)}{\bar{z}'+i\omega'}\right)\mathrm{d}\omega'\mathrm{d}x, \\ F_S(K) &= \frac{1}{\pi}\int_{-\infty}^{\infty}\int_0^{\infty} \Re\left(\frac{q(\alpha,\sigma_0,T,x)}{(\bar{z}'+i\omega')(1+\bar{z}'+i\omega')^2}\right)\mathrm{d}\omega'\mathrm{d}x, \end{aligned}$$

where

$$q(\alpha,\sigma_0,T,x) = \frac{e^{-\frac{1}{2\alpha^2 T}\left(\text{arcCosh}(e^{-x}\lambda\frac{\alpha^2}{\sigma_0^2}+\cosh(x))^2-x^2+(x+T\alpha^2/2)^2\right)}}{\sqrt{2\pi\alpha^2 T}},$$

and the variable $\bar{z}'$, which is associated with the integration path of an inverse Laplace transform, should be chosen such that

$$0 < \bar{z}' < \frac{1}{2}(x_1 + \sqrt{x_1^2 + x_2^2})$$

where

$$\begin{aligned} x_1 &= \left(\frac{(e^x-1)\sigma_0}{2\alpha}\right)^2 + \frac{S_0}{K} - 1, \\ x_2 &= 2\left(\frac{(e^x-1)\sigma_0}{2\alpha}\right)^2. \end{aligned}$$

We calculate these integrals using Gauss-Hermite quadrature points for x and Gauss-Laguerre quadrature points for ω'. Using 2000 points in the x and ω' direction, i.e., a total of 4 million points we get the values

Table A1. SABR European call prices using the parameters from case I. Error $\sim 1e-14$

K	$0.5\exp(-0.1\sqrt{2})$	0.5	$0.5\exp(0.1\sqrt{2})$
Call price	0.221383196830866	0.193836689413803	0.166240814653231

Case II

The solution was obtained by an ADI-method since the analytical method only works in the zero correlation case. The ADI-method employed 890 × 445 spatial grid points and 445 time steps.

K	$0.07\exp(-0.1\sqrt{10})$	0.07	$0.07\exp(0.1\sqrt{10})$
Call price	0.052450313614407	0.046585753491306	0.039291470612989

Table A2. SABR European call prices using the parameters from case II. Error $\sim 1e-6$.

A.2. Methods for Problem 2—QSLV

Heston European Call

The solution was obtained using a Fourier-Gauss-Laguerre implementation with 1000 quadrature points.

S_0	75	100	125
Call price	0.908502728459621	9.046650119220969	28.514786399298796

Table A3. Heston European call prices. Error $\sim 1e-16$.

Heston Double-no-touch

The solution was obtained by an ADI-method since analytical methods only work in the European case. The ADI-method employed 1514×758 spatial grid points and 757 time steps.

S_0	75	100	125
Call price	0.834539127387590	0.899829293208866	0.668399975738358

Table A4. Heston double no-touch prices with lower barrier $L = 50$ and upper barrier $U = 150$. Error $\sim 1.1e-7$.

QSLV European Call

The solution was obtained by an ADI-method since analytical methods only work in the zero correlation case. The ADI-method employed 1486×744 spatial gridpoints and 743 time steps.

S_0	75	100	125
Call price	0.527472759419533	8.902347915743665	29.159828965633729

Table A5. Case Quadratic Stochastic-Local-Volatility European call prices. Error $\sim 2.4e-6$.

QSLV Double-no-touch

The solution was obtained by an ADI-method using 2120×1061 spatial grid points and 1060 time steps

S_0	75	100	125
Call price	0.933800903110254	0.914799140676374	0.592983062889906

Table A6. Quadratic Stochastic-Local-Volatility Double-no-touch prices with lower barrier $L = 50$ and upper barrier $U = 150$. Error $\sim 7.3e-8$.

A.3. Methods for Problem 3—HHW

The solution was computed using a Fourier-Gauss-Laguerre implementation with 1000 quadrature points.

S_0	75	100	125
Call price	35.437896876285350	54.728065308229503	75.397596834993621

Table A7. Heston European call prices. Error $\sim 1e-16$.

Appendix B. The contribution from the authors

Apart from what is listed in Table B1 all authors took part in reading and correcting of the final version of the paper.

During the time for the project, three meetings were held:

A. A kick-off meeting in Uppsala in August 2016, dedicated for the set-up of the project.
B. An intermediate planning meeting in Austin in connection with the SIAM Conference on Financial Mathematics & Engineering in November 2016.
C. A final planning meeting in Leiden in connection with the Lorentz center workshop in Applied Mathematics Techniques for Energy Markets in Transition in September 2017.

In Table B1 it is presented at what meetings the authors took part.

Table B1. Name and contribution of authors

Name	**Contributions**
Lina von Sydow	Project leader Wrote Sections 1, 3, 4, 5, and Appendix B Took part in writing the rest of the paper One of two behind code for RBFFD (SABR and QLSV models) Took part in meetings A, B, and C
Slobodan Milovanović	Performed all numerical experiments Created all plots in Section 4 Took part in writing the paper One of two who wrote unit tests for contributed codes Took part in the computation of reference values One of two behind code for RBFFD (SABR and QLSV models) Took part in meetings A and B
Elisabeth Larsson	Deputy project leader Took part in the writing of Appendix A Took part in writing the rest of the paper Took part in the computation of reference values One of two behind code for RBFPUM (All models) Took part in meetings A, B, and C
Karel In 't Hout	Lead the project of defining benchmark problems Took part in the writing of Section 2 Took part in the computation of reference values One of two behind code for ADI (All models) Took part in meetings A, B, and C
Magnus Wiktorsson	Took part in the writing of Appendix A Took part in the computation of reference values Behind code for FGL (Heston European call option and HHW model) Behind code for MCA (All models) Took part in meeting A
Cornelis W. Oosterlee	Took part in defining benchmark problems Took part in the writing of Section 2 One of two behind code for mSABR (SABR model) One of two behind code for SGBM (QLSV model) Took part in meetings A, B, and C
Victor Shcherbakov	One of two who wrote unit tests for contributed codes One of two behind code for RBFPUM (All models) Took part in meetings A and B Took part in the computation of reference values
Maarten Wyns	One of two behind code for ADI (SABR and QLSV models) Took part in meeting B
Alvaro Leitao	One of two behind code for mSABR (SABR model) Behind code for MLMC (SABR model) Took part in meeting C
Shashi Jain	One of two behind code for SGBM (QLSV model)
Tinne Haentjens	One of two behind code for ADI (HHW model)
Johan Waldén	Took part in the writing of Sections 1 and 2 Took part in meeting A