

# Mixing Monte-Carlo and Partial Differential Equations for Pricing Options

Tobias Lipp, Grégoire Loeper, Olivier Pironneau

► **To cite this version:**

Tobias Lipp, Grégoire Loeper, Olivier Pironneau. Mixing Monte-Carlo and Partial Differential Equations for Pricing Options: In honor of the scientific heritage of Jacques-Louis Lions. Chinese Annals of Mathematics - Series B, Springer Verlag, 2013, 34 (B2), pp.255 - 276. <10.1007/s11401-013-0763-2>. <hal-01558826>

**HAL Id: hal-01558826**

**<http://hal.upmc.fr/hal-01558826>**

Submitted on 10 Jul 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Mixing Monte-Carlo and Partial Differential Equations for Pricing Options

Tobias LIPP<sup>1</sup> Grégoire LOEPER<sup>2</sup> Olivier PIRONNEAU<sup>1</sup>

*(In honor of the scientific heritage of Jacques-Louis Lions)*

**Abstract** There is a need for very fast option pricers when the financial objects are modeled by complex systems of stochastic differential equations. Here the authors investigate option pricers based on mixed Monte-Carlo partial differential solvers for stochastic volatility models such as Heston's. It is found that orders of magnitude in speed are gained on full Monte-Carlo algorithms by solving all equations but one by a Monte-Carlo method, and pricing the underlying asset by a partial differential equation with random coefficients, derived by Itô calculus. This strategy is investigated for vanilla options, barrier options and American options with stochastic volatilities and jumps optionally.

**Keywords** Monte-Carlo, Partial differential equations, Heston model, Financial mathematics, Option pricing

**2000 MR Subject Classification** 91B28, 65L60, 82B31

## 1 Introduction

Since the pioneering work has been achieved by Phelim Boyle [6], Monte-Carlo (or MC for short) methods introduced and shaped financial mathematics as barely any other method can compare. They are often appreciated for their flexibility and applicability in high dimensions, although they bear as well a number of drawbacks: error terms are probabilistic and a high level of accuracy can be computationally burdensome to achieve. In low dimensions, deterministic methods as quadrature and quadrature based methods are strong competitors. They allow deterministic error estimations and give precise results.

We propose several methods for pricing basket options in a Black-Scholes framework. The methods are based on a combination of Monte-Carlo, quadrature and partial differential equations (or PDE for short) methods. The key idea was studied by two of the authors a few years ago in [14], and it tries to uncouple the underlying system of stochastic differential equations (or SDE for short), and then applies the last-mentioned methods appropriately.

In Section 2, we begin with a numerical assessment on the use of Monte-Carlo methods to generate boundary conditions for stochastic volatility models, but this is a side remark independent of what follows.

---

Manuscript received September 3, 2012.

<sup>1</sup>LJLL-UPMC, Boite 187, Place Jussieu, 75252 Paris cedex 5, France.

E-mail: tobias.lipp@web.de Olivier.Pironneau@upmc.fr

<sup>2</sup>BNP-Paribas, 20 Boulevard des Italiens, 75009 Paris, France. E-mail: gregoire.loeper@bnpparibas.com

The way of mixing MC and PDE for stochastic volatility models is formulated in Section 3. A numerical evaluation of the method is made by using closed form solutions to the PDE. In Sections 6 and 4, the method is extended to the case of American options and to the case where the underlying asset is modeled with jump-diffusion processes.

In Section 5, a method reducing the number of samples is given based on the smooth dependence of the option price on the volatility.

Finally, in Section 7, the strategy is extended to multidimensional problems like basket options, and numerical results are also given.

## 2 Monte-Carlo Algorithm to Generate Boundary Conditions for the PDE

The diffusion process that we have chosen for our examples is the Heston stochastic volatility model (see [12]). Under a risk neutral probability, the risky asset  $S_t$  and the volatility  $\sigma_t$  follow the diffusion process

$$dS_t = S_t(rdt + \sigma_t dW_t^1), \quad (2.1)$$

$$dv_t = k(\theta - v_t)dt + \delta\sqrt{v_t}dW_t^2, \quad (2.2)$$

and the put option price is given by

$$P_t = e^{-r(T-t)}\mathbb{E}[(K - S_T)^+ | S_t, v_t], \quad (2.3)$$

where  $v_t = \sigma_t^2$ ,  $\mathbb{E}(dW_t^1 \cdot dW_t^2) = \rho dt$ ,  $\mathbb{E}(\cdot)$  is the expectation with respect to the risk neutral measure, and  $r$  is the interest rate on a risk less commodity.

The pair  $(W^1, W^2)$  is a two-dimensional correlated Brownian motion, with the correlation between the two components being equal to  $\rho$ . As it is usually observed in equity option markets, options with low strikes have an implied volatility higher than that of options at the money or with high strikes, and it is known as the smile. This phenomenon can be reproduced in the model by choosing a negative value of  $\rho$ .

The time is discretized into  $N$  steps of length  $\delta t$ . Denoting by  $T$  the maturity of the option, we have  $T = N\delta t$ . Full Monte-Carlo simulation (see [10]) consists in a time loop starting at  $S_0, v_0 = \sigma_0^2$  of

$$v_{i+1} = v_i + k(\theta - v_i)\delta t + \sigma_i\sqrt{\delta t}N_{0,1}^2\delta \quad \text{with } \sigma_i = \sqrt{v_i}, \quad (2.4)$$

$$S_{i+1} = S_i(1 + r\delta t + \sigma_i\sqrt{\delta t}(N_{0,1}^1\rho + N_{0,1}^2\sqrt{1 - \rho^2})), \quad (2.5)$$

where  $N_{0,1}^j$  ( $j = 1, 2$ ) are realizations of two independent normal Gaussian variables. Then set  $P_0 = \frac{e^{-rT}}{M} \sum (K - S_N^m)^+$ , where  $\{S_N^m\}_{m=1}^M$  are  $M$  realizations of  $S_N$ .

The method is slow, and at least 300000 samples are necessary for a precision of 0.1%. Of course acceleration methods exist (quasi-Monte-Carlo, multi-level Monte-Carlo etc.), but alternatively, we can use the PDE derived by Itô calculus for  $u$  below and set  $P_0 = u(S_0, v_0, T)$ .

If the return to volatility is 0 (i.e., zero risk premium on the volatility (see [1])), then

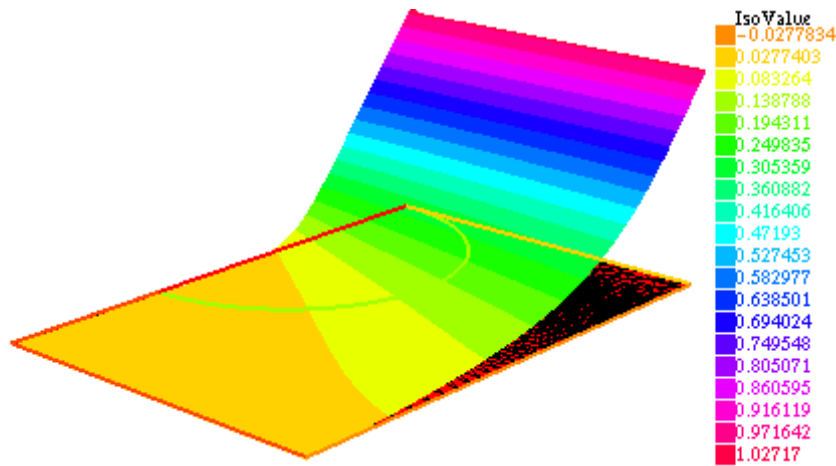
$u(S, y, \tau)$  is given by

$$\partial_\tau u - \frac{yS^2}{2} \partial_{SS} u - \rho \lambda S y \partial_{S_y} u - \frac{\lambda^2 y}{2} \partial_{yy} u - rS \partial_S u - k(\theta - y) \partial_y u + ru = 0, \tag{2.6}$$

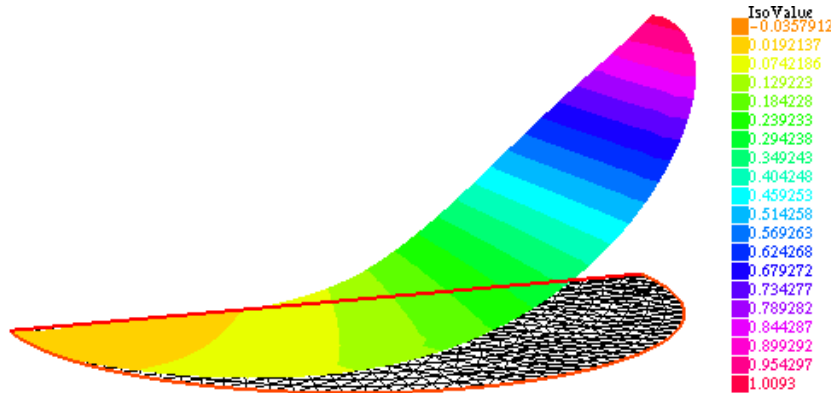
$$u(S, y, 0) = (K - S)^+.$$

Now instead of integrating (2.6) on  $\mathbb{R}^+ \times \mathbb{R}^+ \times (0, T)$ , let us integrate it on  $\Omega \times (0, T)$ ,  $\Omega \subset \mathbb{R}^+ \times \mathbb{R}^+$ , and add Dirichlet conditions on  $\partial\Omega$  computed with MC by solving (2.4)–(2.5).

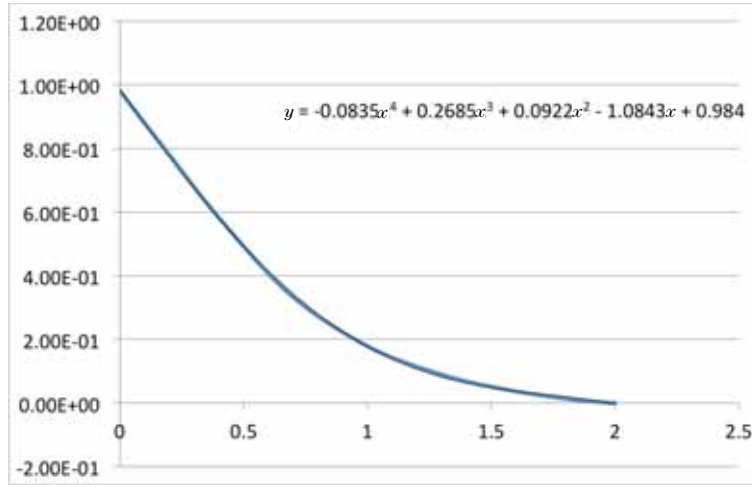
Notice that this domain reduction does not change the numerical complexity of the problem. Indeed to reach a precision  $\varepsilon$  with the PDE, one needs at least  $O(\varepsilon^{-3})$  operations to compute the option at all points of a grid of size  $\varepsilon$  with a time step of size  $\varepsilon$ . Monte-Carlo needs  $O(\varepsilon^{-2})$  per point  $S_0, v_0$ , and there are  $O(\varepsilon^{-1})$  points on the artificial boundary, when the number of discretization points in the full domain is  $O(\varepsilon^{-2})$ . However, the computation shown in Figure 1 validates the methodology, and it may be attractive to use it to obtain more precision on a small domain.



(a) The computational domain is  $(0, y_{\max}) \times (0, v_{\max})$  with Neumann condition at  $v = v_{\max}$ .



(b) The computational domain is the half circle shown on (a); the Dirichlet boundary condition on the circle is obtained by a spline approximation (shown at the bottom) of the solution to Heston’s model solved by MC on a few points on the circle.



(c) Computation result.

Figure 1 Put option with Heston’s model computed by solving the PDE by implicit Euler + FEM using the public domain package freefem++ (see [11]).

### 3 Monte-Carlo Mixed with a 1-Dimensional PDE

Let us rewrite (2.1) as

$$dS_t = S_t[r dt + \sigma_t \sqrt{1 - \rho^2} d\widetilde{W}_t^{(1)} + \sigma_t \rho d\widetilde{W}_t^{(2)}], \tag{3.1}$$

where  $\widetilde{W}_t^1, \widetilde{W}_t^2$  are now independent Brownian motions.

Drawing a trajectory of  $v_t$  by (2.4), with the same  $\delta t$  and the same discrete trajectory  $W_{i+1}^{(2)} = W_i^{(2)} + N_{0,1}^2 \sqrt{\delta t}$ , we consider

$$dS_t = S_t[\mu_t dt + \sigma_t \sqrt{1 - \rho^2} d\widetilde{W}_t^{(1)}], \tag{3.2}$$

$$\mu_t = r + \rho \sigma_t \frac{W_{i+1}^{(2)} - W_i^{(2)}}{\delta t} - \frac{1}{2} \rho^2 \sigma_t^2, \quad t \in [t_i, t_{i+1}[. \tag{3.3}$$

**Proposition 3.1** *As  $\delta t \rightarrow 0$ ,  $S_t$  given by (2.4) and (3.2)–(3.3) converges to the solution to Heston’s model (2.1)–(2.2). Moreover, the put  $P = e^{-rT} \mathbb{E}(K - S_T)^+$  is also the expected value of  $u(S_0, 0)$ , with  $u$  given by*

$$\partial_t u + \frac{1}{2} (1 - \rho^2) \sigma_t^2 S^2 \partial_{SS} u + S \mu_t \partial_S u - ru = 0, \quad u(S, T) = (K - S)^+ \tag{3.4}$$

with  $\sigma_t$  given by (2.4) and  $\mu_t$  given by (3.3).

**Proof** By Itô’s formula, we have

$$\begin{aligned} d \log(S_t) &= \frac{dS_t}{S_t} + \frac{1}{2} (\log S)'' (S_t^2 \sigma_t^2 (1 - \rho^2) dt) = \frac{dS_t}{S_t} - \frac{\sigma_t^2}{2} (1 - \rho^2) dt \\ &= \mu_t dt + \sqrt{1 - \rho^2} \sigma_t d\widetilde{W}_t^{(1)} - (1 - \rho^2) \frac{\sigma_t^2}{2} dt \end{aligned}$$

$$\begin{aligned} &\approx r\delta t + \rho\sigma_t\delta W_t^{(2)} - \frac{\rho^2\sigma_t^2}{2}\delta t + \sqrt{1-\rho^2}\sigma_t\delta\widetilde{W}_t^{(1)} - (1-\rho^2)\frac{\sigma_t^2}{2}\delta t \\ &\approx rdt + \rho\sigma_t dW_t^{(2)} + \sqrt{1-\rho^2}\sigma_t d\widetilde{W}_t^{(1)} - \frac{\sigma_t^2}{2}dt. \end{aligned} \tag{3.5}$$

Consequently,

$$S_t = S_0 \exp\left(\int_0^t \mu_t dt + \int_0^t \sqrt{1-\rho^2}\sigma_t dW_t^{(1)} - \int_0^t \frac{1}{2}(1-\rho^2)\sigma_t^2 dt\right). \tag{3.6}$$

**Proposition 3.2** *If we restrict the MC samples to those that give  $0 < \sigma_m \leq \sigma_t \leq \sigma_M$ , for some given  $\sigma_m, \sigma_M$ , then equations (2.4) and (3.3)–(3.4) are well-posed.*

**Proof** Let

$$\Lambda_\tau = \int_{T-\tau}^T \mu_\xi d\xi, \quad y = \frac{S}{K}e^{\Lambda(\tau)}. \tag{3.7}$$

Then  $u(t, S) = v(T - t, \frac{S}{K}e^{\Lambda(\tau)})$ , where  $v$  is the solution to

$$\partial_\tau v - \frac{1}{2}(1-\rho^2)\sigma_{T-\tau}^2 y^2 \partial_{yy} v = 0, \quad v(0, y) = (1-y)^+. \tag{3.8}$$

If  $0 < \sigma_m \leq \sigma_t \leq \sigma_M$  almost surely and for all  $t$ , then the solution exists in the sense of Barth et al. [3].

**Remark 3.1** Note that (3.6) is also

$$\bar{\sigma}^2 = \frac{1-\rho^2}{T} \int_0^T \sigma_t^2 dt, \quad m = r - \frac{\bar{\sigma}^2}{2} + \frac{\rho}{T} \sum_i \sigma_{t_i} (W_{t_{i+1}}^{(2)} - W_{t_i}^{(2)}), \tag{3.9}$$

$$S_T(x) = S_0 \exp(mT + \bar{\sigma}Tx). \tag{3.10}$$

Therefore,

$$\mathbb{E}[u(S_0, 0)] = e^{-rT} \int_{\mathbb{R}^+} (K - S_0 e^{mT + \bar{\sigma}Tx})^+ \frac{e^{-\frac{x^2}{2T}}}{\sqrt{2\pi T}} dx. \tag{3.11}$$

There is a closed form for this integral, namely the Black-Scholes (or BS for short) formula with the interest rate  $r$ , the dividend  $m + r$  and the volatility  $\bar{\sigma}$ .

### 3.1 Numerical tests

In the simulations, the parameters are  $S_0 = 100, K = 90, r = 0.05, \sigma_0 = 0.6, \theta = 0.36, k = 5, \lambda = 0.2, T = 0.5$ . We compared a full MC solution with  $M$  samples to the new algorithm with  $M'$  samples for  $\mu_t$  and  $\sigma_t$  given by (2.4). The Black-Scholes formula is used as indicated in Remark 3.1.

To observe the precision with respect to  $\rho$ , we have taken a large number of Monte-Carlo samples, i.e.,  $M = 3 \times 10^5$  and  $M' = 10^4$ . Similarly, the number of time steps is 300 with 400 mesh points and  $S_{\max} = 600$  (i.e.,  $\delta S = 1.5$ ).

To study the precision, we let  $M$  and  $M'$  vary. Table 2 shows the results for 5 realizations of both algorithms and the corresponding mean value for  $P_N$  and variance.

Note that one needs many more samples for pure MC than those for the mixed strategy MC+BS. This variance reduction explains why MC+BS is much faster.

Table 1 Precision versus  $\rho$ .

$\rho$	-0.5	0	0.5	0.9
Heston MC	11.135	10.399	9.587	8.960
Heston MC+BS	11.102	10.391	9.718	8.977
Speed-up	42	44	42	42

Table 2 Precision study with respect to  $M$  and  $M'$ . Five realizations of pure MC and MC+PDE for various  $M'$  and  $M$ .

	MC+BS: $M'=$			MC: $M=$		
	100	1000	10000	3000	30000	300000
$P^1$	10.475	11.129	11.100	11.564	11.481	11.169
$P^2$	10.436	11.377	11.120	11.6978	11.409	11.249
$P^3$	11.025	11.528	11.113	11.734	11.383	11.143
$P^4$	11.205	11.002	11.113	11.565	11.482	11.169
$P^5$	11.527	11.360	11.150	11.085	11.519	11.208
$P = \frac{1}{5} \sum P^i$	10.934	11.279	11.119	11.529	11.454	11.187
$\sqrt{\frac{1}{5} \sum (P^i - P)^2}$	0.422	0.188	0.0168	0.232	0.0507	0.0370

### 4 Lévy Processes

Consider Bates model (see [4]), i.e., an asset modeled with stochastic volatility and a jump process,

$$dv_t = k(\theta - v_t)dt + \xi\sqrt{v_t}dW_t^{(2)}, \quad \sigma_t = \sqrt{v_t}, \tag{4.1}$$

$$dX_t = \left(r - \frac{\sigma_t^2}{2}\right)dt + \sigma_t(\sqrt{1 - \rho^2}d\widetilde{W}_t^{(1)} + \rho d\widetilde{W}_t^{(2)}) + \eta dN_t, \tag{4.2}$$

where  $X_t = \ln S_t$  and  $N_t$  is a Poisson process. As before, this is

$$dX_t = \widetilde{\mu}_t dt + \sigma_t \sqrt{1 - \rho^2} d\widetilde{W}_t^{(1)} + \eta dN_t, \tag{4.3}$$

$$\widetilde{\mu}_t = r - \frac{\sigma_t^2}{2} + \rho \sigma_t \frac{\delta W^{(2)}}{\delta t}. \tag{4.4}$$

By Itô, a put on  $S_t$  with  $u(T) = (K - e^x)^+$  satisfies

$$\begin{aligned} &\partial_t u - ru + \frac{1}{2}(1 - \rho^2)\sigma_t^2 \partial_{xx} u + \widetilde{\mu}_t \partial_x u \\ &= - \int_{\mathbb{R}} [(u(x+z) - u(x))J(z) - \partial_x u(x)(e^z - 1)J(z)] dz. \end{aligned} \tag{4.5}$$

Let us apply a change of variables  $\tau = T - t$ ,  $y = x - \int_{T-\tau}^T \bar{\mu}_t dt$  with  $\bar{\mu}_t = \widetilde{\mu}_t - \int_{\mathbb{R}} (e^z - 1)J(z) dz$ , and use

$$v(y, \tau) = e^{(r + \int_{\mathbb{R}} J(z) dz)\tau} u\left(y + \int_{T-\tau}^T \bar{\mu}_t dt, T - \tau\right). \tag{4.6}$$

**Proposition 4.1**

$$\partial_\tau v - \frac{1}{2}(1 - \rho^2)\sigma_{T-\tau}^2 \partial_{yy} v - \int_{\mathbb{R}} v(y+z)J(z) dz = 0, \quad v(y, 0) = (K - e^y)^+. \tag{4.7}$$

**Proof** Let  $\bar{r} = r + \int_{\mathbb{R}} J(z)dz$ . Then

$$\partial_{\tau}v = e^{\bar{r}\tau} \left[ - \left( r + \int_{\mathbb{R}} J(z)dz \right) u + \bar{\mu}_{T-\tau} \partial_x u - \partial_t u \right], \quad \partial_y v = e^{\bar{r}\tau} \partial_x u, \quad \partial_{yy} v = e^{\bar{r}\tau} \partial_{xx} u. \quad (4.8)$$

Therefore,

$$\begin{aligned} & e^{-\bar{r}\tau} \left[ \partial_{\tau}v - \frac{1}{2}(1 - \rho^2)\sigma_{\tau}^2 \partial_{yy} v - \int_{\mathbb{R}} v(y+z)J(z)dz \right] \\ &= \left( r + \int_{\mathbb{R}} J(z)dz \right) u + \bar{\mu}_t \partial_x u - \partial_t u - (1 - \rho^2) \frac{\sigma_t^2}{2} \partial_{xx} u - \int_{\mathbb{R}} u(x+z)J(z)dz, \end{aligned}$$

which is zero by (4.5).

**Remark 4.1** Once more, we notice that the PDE depends on time integrals of  $\tilde{\mu}_t$  and  $\sigma_t$ , and integrals damp the randomness and make the partial integro-differential equation (or PIDE for short) (4.7) easier to solve. Table 3 displays 9 realizations of  $\sqrt{\frac{1}{T} \int_0^T \sigma_t^2 dt}$  for  $M' = 100$  and 500.

Table 3 9 realizations of  $(\frac{1}{T} \int_0^T \sigma_t^2 dt)^{\frac{1}{2}}$  for  $M' = 100$  and 500.

$M'$	T1	T2	T3	T4	T5	T6	T7	T8	T9	mean
100	0.3470	0.3482	0.3496	0.3484	0.3474	0.3548	0.3492	0.3492	0.3502	0.3493± 0.002
500	0.3490	0.3481	0.3488	0.3493	0.3502	0.3501	0.3501	0.3489	0.3488	0.3493± 0.0007

**Remark 4.2** Let  $\bar{f}_{\tau} = \frac{1}{\tau} \int_{T-\tau}^T f(t)dt$ . From (4.6), we see that the option price is recovered by

$$\begin{aligned} u(S, t) = & e^{-(r+\int_{\mathbb{R}} J(z)dz)(T-t)} v \left( \ln S - \left( r - \frac{\overline{\sigma_t^2}|_t}{2} - \int_{\mathbb{R}} (e^z - 1)J(z)dz \right. \right. \\ & \left. \left. + \rho \sigma_t \frac{\overline{\delta W^{(2)}}}{\delta t} \Big|_t \right) (T-t), T-t \right), \end{aligned}$$

where  $v$  is the solution to (4.7). For a European put option, with the standard diffusion-Lévy process model and the dividend  $q$ , the formula is

$$\begin{aligned} u(S, t) = & e^{-(r+\int_{\mathbb{R}} J(z)dz)(T-t)} v \left( \ln S - \left( r - q - \frac{\sigma^2}{2} - \int_{\mathbb{R}} (e^z - 1)J(z)dz \right) (T-t), T-t \right), \\ \partial_{\tau}v - \frac{1}{2}\sigma^2 \partial_{yy} v - \int_{\mathbb{R}} v(y+z)J(z)dz = & 0, \quad v(y, 0) = (K - e^y)^+. \end{aligned} \quad (4.9)$$

It means that any solver for the European put option, with the standard diffusion-Lévy process model and the dividend  $q$ , can be used provided that the following modifications are made:

- (1) In the solver, change  $\sigma^2$  into  $(1 - \rho^2)\overline{\sigma_t^2}|_t$ .
- (2) Change  $q$  into  $q + \rho^2\overline{\sigma_t^2}|_t - \rho \sigma_t \frac{\overline{\delta W^{(2)}}}{\delta t} \Big|_t$ .

#### 4.1 The numerical solution to the PIDE by the spectral method

Let the Fourier transform operators be

$$\mathbb{F}(u) = \int_{\mathbb{R}} e^{-i\omega x} u(x)dx \quad \text{and} \quad \mathbb{F}^{-1}(\hat{u}) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{i\omega x} \hat{u}(\omega)d\omega. \quad (4.10)$$



Applying the operator  $\mathbb{F}$  to the PIDE (4.7) for a call option gives

$$\partial_\tau \widehat{v} - \Psi \widehat{v} = 0 \quad \text{in } \mathbb{R}, \quad \widehat{v}(\omega, 0) = \mathbb{F}(e^x - K)^+, \tag{4.11}$$

where  $\Psi$  is

$$\Psi(\omega) = -(1 - \rho^2) \frac{\sigma_t^2}{2} \omega^2 - \varphi(\omega), \quad \varphi(\omega) = \int_{\mathbb{R}} e^{i\omega y} J(y) dy. \tag{4.12}$$

So, with  $m$  indicating a realization, the solution is

$$u\left(x - \int_{T-\tau}^T \bar{\mu}_t dt\right) = \frac{1}{M'} \sum_m e^{-rT} (K - \mathbb{F}^{-1}[\{\mathbb{F}v^0\}(\omega) e^{-\varphi(\omega)\tau - \omega^2 \frac{1-\rho^2}{2} \int_{T-\tau}^T \sigma_t^2 dt}])^+ \tag{4.13}$$

with  $\tilde{\mu}_t$  given by (3.3) and  $\bar{\mu}_t = \tilde{\mu}_t + \int_{\mathbb{R}} (e^z - 1) J(z) dz$ .

**Remark 4.3** The Car-Madan trick in [7] must be used, and  $v^0$  must be replaced by  $e^{-\eta S}(S - K)^+$ , which has a Fourier transform, in the case of a call option. Then in (4.13)  $\mathbb{F}^{-1}\widehat{\chi}$  must be changed into

$$\frac{K^\eta}{\pi} \int_0^\infty \Re(e^{-i\omega S} \widehat{\chi}(\xi + i\eta)) d\xi.$$

**Remark 4.4** As an alternative to the fast Fourier transform (or FFT for short) methods, following Lewis [13], for a call option, when  $\Im\omega > 1$ ,

$$\mathbb{F}v^0 = \mathbb{F}(e^y - K)^+ = -\frac{e^{\ln K(i\omega+1)}}{\omega^2 - i\omega}. \tag{4.14}$$

Using such extended calculus in the complex plane, Lewis obtained for the call option,

$$u(S, T) = S - \frac{\sqrt{KS}}{\pi} \int_0^\infty \Re\left[e^{iuk} \phi_T\left(u - \frac{i}{2}\right)\right] \frac{du}{u^2 + \frac{1}{4}} \tag{4.15}$$

with  $k = \ln \frac{S}{K}$ , where  $\phi_t$  is the characteristic function of the process, which, in the case of (4.7) with Merton Kernel (see [15])

$$J(x) = \lambda \frac{e^{-\frac{(x-\mu)^2}{\delta^2}}}{\sqrt{2\pi\delta^2}},$$

is

$$\phi_T(u) = \exp\left(iuwT - \frac{1}{2}u^2\Sigma^2T + T\lambda(e^{-\frac{\delta^2u^2}{2} + i\mu u} - 1)\right)$$

with  $\Sigma^2 = \frac{1}{T} \int_0^T \sigma_\tau^2 d\tau$  and  $w = \frac{1}{2}\Sigma^2 - \lambda(e^{\frac{\delta^2}{2} + \mu} - 1)$ . The method has been tested with the following parameters:

$$\begin{aligned} T = 1, \quad \mu = -0.5, \quad \lambda = 0.1, \quad \delta = 0.4, \quad K = 1, \quad r = 0.03, \quad \sigma_0 = 0.4, \quad \theta = 0.4, \quad \kappa = 2, \\ \rho = -0.5, \quad \xi = 0.25, \quad M' = 10000, \quad \delta t = 0.001. \end{aligned} \tag{4.16}$$

Results for a put are reported in Figure 2. The method is not precise out of the money, i.e.,  $S > K$ . The central processing unit (or CPU for short) is 0.8'' per point on the curve.

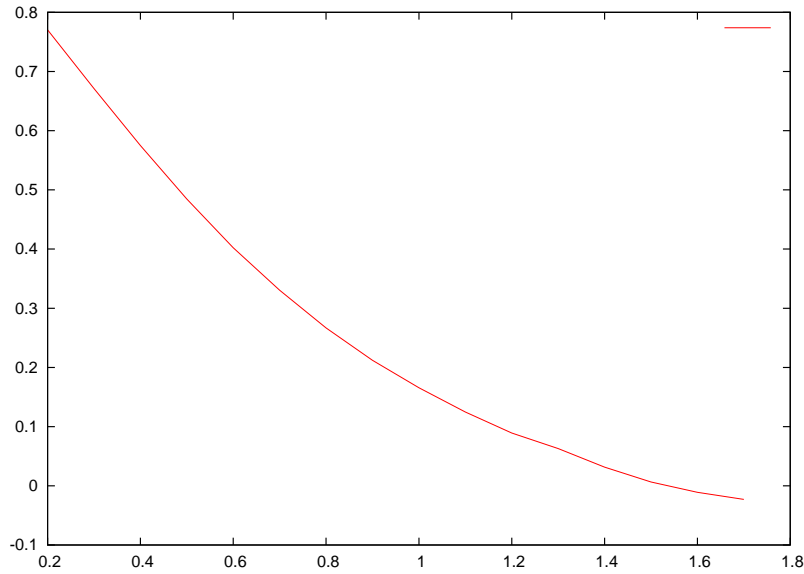


Figure 2 Put calculated with Bates' model by mixing MC with Lewis' formula (see (4.15)).

### 4.2 Numerical results

The method has been tested numerically. The coefficients for the Heston+Merton-Lévy are  $T = 1, r = 0, \xi = 0.3, v_0 = 0.1, \theta = 0.1, k = 2, \lambda = 0.3, \rho = 0.5$ . This gives an average volatility 0.27. For the Heston and the pure Black-Scholes for comparison,  $T = 1, r = 0, \sigma = 0.3, \lambda = 5, m = -0.01, v = 0.01$ .

The results are shown in Figure 3.

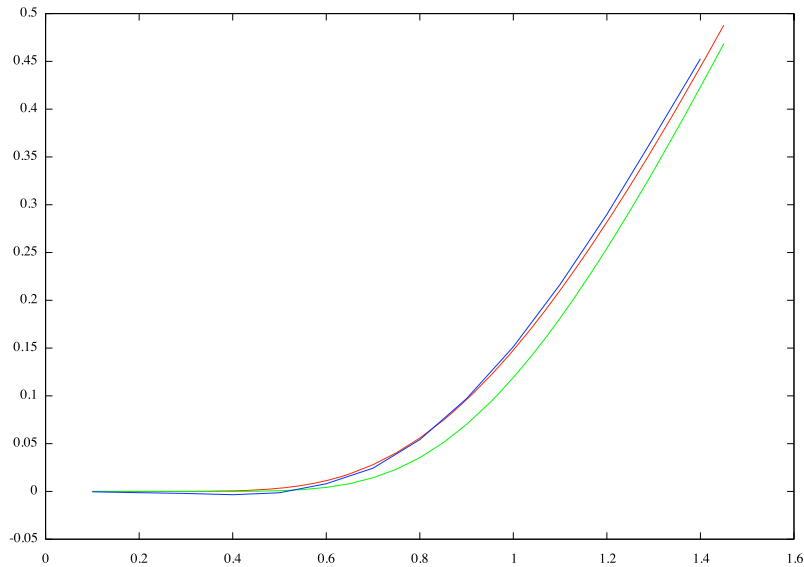


Figure 3 Call calculated by a Heston+Merton-Lévy by mixed MC-Fourier (see the blue curve), and compared with the solution to the 2-dimensional PIDE Black-Scholes+Lévy (see the red curve), and a pure Black-Scholes (see the green curve).

## 5 Conditional Expectation with Spot and Volatility

If the full surface  $\sigma_0, S_0 \rightarrow u(\sigma_0, S_0, 0)$  is required, MC+PDE becomes prohibitively expensive, much like MC is too expensive if  $S_0 \rightarrow u(S_0, 0)$  is required for all  $S$ .

However, notice that after some time  $t_1$  the stochastic differential equation (or SDE for short) for  $\sigma_t$  will generate a large number of sample values  $\sigma_1$ . Let us take advantage of this to compute  $u(\sigma_1, S_1, t_1)$ .

### 5.1 Polynomial fits

Let  $\tau = T - t_1$  for some fixed  $t_1$ .

Instead of gathering all  $u(\cdot, \tau)$  corresponding to the samples  $\sigma_\tau^m$  with the same initial value  $\sigma_0$  at  $t = 0$ , we focus on the time interval  $(t_1, T)$ , consider that  $\sigma_t^m$  is a stochastic volatility initiated by  $\sigma_{t_1}^{(m)}$ , then search for the best polynomial fit in terms of  $\sigma$  for  $u$ , i.e., a projection on the basis  $\phi_k(\sigma)$  of  $\mathbb{R}$ , and solve

$$\min_{\alpha} J(\alpha) := \frac{1}{M} \sum_m \frac{1}{L} \int_0^L \left\| \sum_k \alpha_k(S) \phi^k(\sigma_\tau^{(m)}) - u^{(m)}(S, \tau) \right\|^2 dS.$$

It leads to solving, for each  $S_i = i\delta S$ ,

$$\left( \frac{1}{M} \sum_m \phi_k(\sigma_\tau^{(m)}) \phi_l(\sigma_\tau^{(m)}) \right) \alpha_k^i = \frac{1}{M} \sum_m u^{(m)}(S_i, \tau) \phi_l(\sigma_\tau^{(m)}). \quad (5.1)$$

### 5.2 Piecewise constant approximation on intervals

We begin with a local basis of polynomials, namely,  $\phi_k(\sigma) = 1$  if  $\sigma \in (\sigma_k, \sigma_{k+1})$  and  $\phi_k(\sigma) = 0$  otherwise.

- Algorithm 5.1**
- (1) Choose  $\sigma_m, \sigma_M, \delta\sigma, \sigma_0$ .
  - (2) Initialize an array  $n[j] = 0, j = 0, \dots, J := \frac{\sigma_M - \sigma_m}{\delta\sigma}$ .
  - (3) Compute  $M$  realizations  $\{\sigma_{t_i}^{(m)}\}$  by MC on the volatility equation.
  - (4) For each realization, compute  $u(\cdot, \tau)$  by solving the PDE.
  - (5) Set  $j = \frac{\sigma_\tau^{(m)} - \sigma_m}{\delta\sigma}$  and  $n[j] += 1$ , and store  $u(\cdot, \tau)$  in  $w(\cdot)[j]$ .
  - (6) The answer is  $u(\sigma; S, \tau) = \frac{w(S)[j]}{n[j]}$  with  $j = \frac{\sigma - \sigma_m}{\delta\sigma}$ .

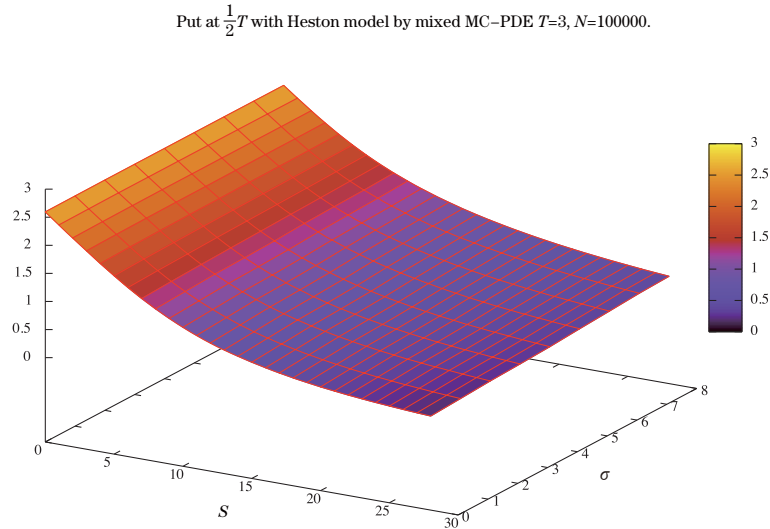
### 5.3 Polynomial projection

Now we choose  $\phi_k(\sigma) = \sigma^k$ .

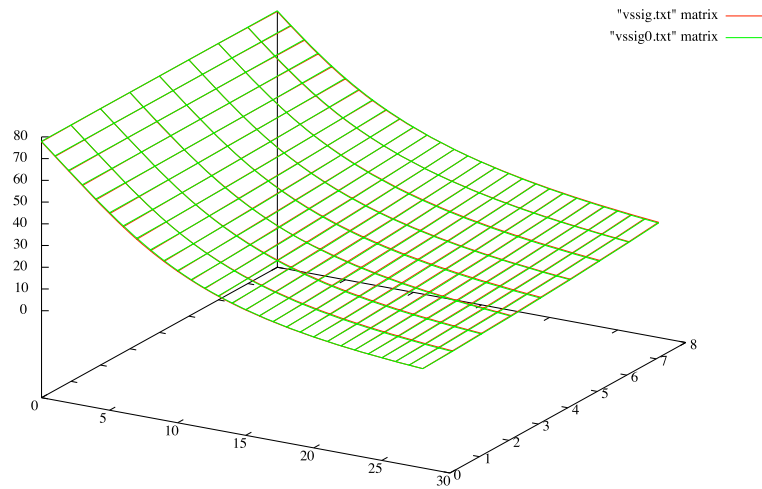
- Algorithm 5.2**
- (1) Choose  $\sigma_m, \sigma_M, \delta\sigma, \sigma_0$ .
  - (2) Set  $A[\cdot][\cdot] = 0, b[\cdot][\cdot] = 0$ .
  - (3) Compute  $M$  realizations  $\{\sigma_{t_i}^{(m)}\}$  by MC on the volatility equation and for each realization.
    - (i) Compute  $u(\cdot, \tau)$  by solving the PDE.
    - (ii) Do  $A[j][k] += \frac{1}{M} \sum_m (\sigma_\tau^{(m)})^{j+k}, j, k = 1, \dots, K$ .
    - (iii) Do  $b[i][k] += \frac{1}{M} u(i\delta S, \tau) (\sigma_\tau^{(m)})^k, k = 1, \dots, K$ .
  - (4) The answer is found by solving (5.1) for each  $i = 1, \dots, N$ .

5.4 The numerical test

A Vanilla put with the same characteristics as in subsection 3.1 has been computed by Algorithm 5.2 for a maturity of 3 years. The surface  $S_{t_1}, \sigma_{t_1} \rightarrow u$  is shown after  $t_1 = 1.5$  years in Figure 4. The implied volatility is also shown.



(a) Local volatility of a vanilla put with 3 years maturity after 1.5 years, computed with a Heston model by the mixed MC-PDE algorithm with polynomial projection.



(b) Comparison on the price of the put computed with full MC Heston.

Figure 4 Both surfaces (a) and (b) are on top of each other, indistinguishable.

6 American and Bermudan Options

For American options, we must proceed step by step backward in time as in the dynamic programming for binary trees (see [2]).

Consider  $M'$  realizations  $[\{\sigma_t^m\}_{t \in (0,T)}]_{m=1}^{M'}$ , giving  $[\{\mu_t^m\}_{t \in (0,T)}]_{m=1}^{M'}$  by (3.3). At time  $t_n = T$ , the price of the contract is  $(K - S)^+$ . At time  $t_{n-1} = T - \delta t$ , it is given by the maximum of the European contract, knowing  $S$  and  $\sigma$  at  $t_{n-1}$  and  $(K - S)^+$ , i.e.,

$$u_{n-1}(S) = \max \left\{ \frac{1}{|M_\sigma|} \sum_{m \in M_\sigma} u_{n-1}^m(S), (K - S)^+ \right\}, \tag{6.1}$$

where  $u_{n-1}^m$  is the solution at  $t_{n-1}$  to

$$\begin{aligned} \partial_t u + (1 - \rho^2) \frac{(S\sigma_t^m)^2}{2} \partial_{SS} u + S\mu_t^m \partial_S u - ru &= 0, \quad t \in (t_{n-1}, t_n), \\ u_n &:= u(S, t_n), \end{aligned} \tag{6.2}$$

where  $u_n$  is known, and  $M_\sigma$  is the set of trajectories which give a volatility equal to  $\sigma$  at time  $t$ .

Here we have used the piecewise constant approximation intervals to compute the European premium. Alternatively, one could use any projection method, and the backward algorithm follows the same lines.

As with American options with binary trees, convergence with optimal order will hold only if  $\delta t$  is small enough.  $M_\sigma$  is built as in the previous section.

To prove the concept, we computed a Bermudan contract at  $\frac{1}{2}T$  by the above method, using the polynomial basis for the projection. The parameters are the same as above except  $K = 100$ . The results are displayed in Figure 5. To obtain the price of the option at time zero, the surface of Figure 5, i.e., (6.1), must be used as time-boundary conditions for the MC-PDE mixed solver for  $t \in (0, \frac{1}{2}T)$ , while for Americans, this strategy is applied at every time step, but here it is done once only at  $\frac{1}{2}T$ .

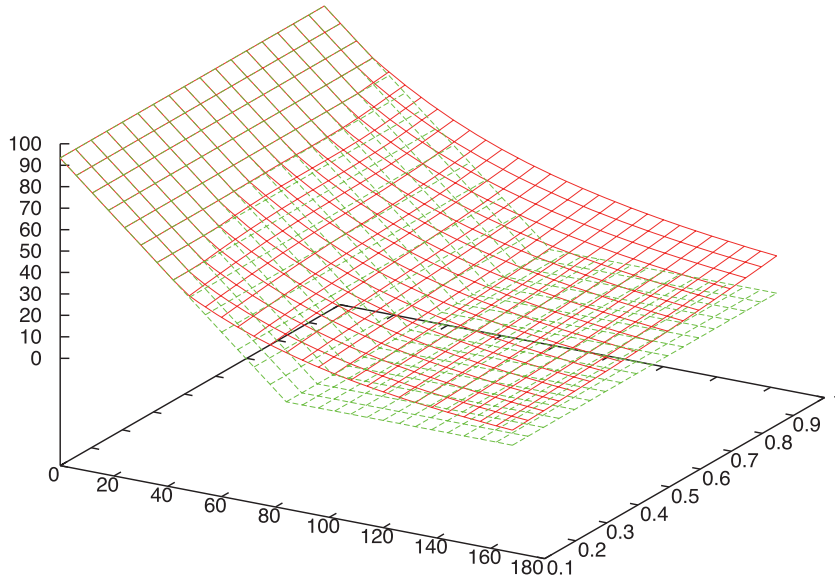


Figure 5 A Bermuda option at  $\frac{1}{2}T$  with Heston's model compared with  $(K - S)^+$ .

## 7 Systems of Dimension Greater than 2

Stochastic volatility models with several SDEs for the volatilities are now in use. However, in order to assess the mixed MC-PDE method, we need to work on a systems for which an exact or precise solution is easily available. Therefore, we will investigate basket options instead.

### 7.1 Problem formulation

We consider an option  $P$  on three assets whose dynamics are determined by the following system of stochastic differential equations:

$$dS_{i,t} = S_{i,t}(rdt + dW_{i,t}), \quad t > 0, \quad i = 1, 2, 3 \tag{7.1}$$

with initial conditions  $S_{i,t=0} = S_{i,0}$ ,  $S_{i,0} \in \mathbb{R}^+$ . The parameter  $r$  ( $r \in \mathbb{R}_{\geq 0}$ ) is constant, and  $W_i := \sum_{j=1}^3 a_{ij}B_j$  are linear combinations of standard Brownian motions  $B_j$ , such that

$$\text{Cov}[W_{i,t}, W_{j,t}] = \rho_{ij}\sigma_i\sigma_jt, \quad t > 0.$$

We further assume that  $\Xi := (\rho_{ij}\sigma_i\sigma_j)_{i,j=1}^3$  is symmetric positive definite with

$$\rho_{ij} = 1 \quad (i = j) \quad \text{or} \quad \rho_{ij} \in (-1, 1) \text{ otherwise.}$$

The coefficients  $a_{ij}$  ( $a_{ij} \in \mathbb{R}$ ) have to be chosen, such that

$$\begin{aligned} \text{Cov}[W_{i,t}, W_{j,t}] &= E[W_{i,t}W_{j,t}] \\ &= E[(a_{i1}B_{1,t} + a_{i2}B_{2,t} + a_{i3}B_{3,t})(a_{j1}B_{1,t} + a_{j2}B_{2,t} + a_{j3}B_{3,t})] \\ &= a_{i1}a_{j1}E[B_{1,t}^2] + a_{i2}a_{j2}E[B_{2,t}^2] + a_{i3}a_{j3}E[B_{3,t}^2] \\ &= (a_{i1}a_{j1} + a_{i2}a_{j2} + a_{i3}a_{j3})t, \quad t > 0, \end{aligned}$$

or equivalently,

$$AA^T = \Xi,$$

where  $A := (a_{ij})_{i,j=1}^3$ . Without loss of generality, we may set the strict upper triangular components of  $A$  to zero and find

$$A = \begin{pmatrix} \sigma_1 & 0 & 0 \\ \sigma_2\rho_{21} & \sigma_2\sqrt{1 - \rho_{12}^2} & 0 \\ \sigma_3\rho_{31} & \sigma_3\frac{\rho_{32} - \rho_{21}\rho_{31}}{\sqrt{1 - \rho_{12}^2}} & \sigma_3\sqrt{1 - \rho_{31}^2 - \left(\frac{\rho_{32} - \rho_{21}\rho_{31}}{\sqrt{1 - \rho_{12}^2}}\right)^2} \end{pmatrix}.$$

The option  $P$  has the maturity  $T$  ( $T \in \mathbb{R}^+$ ), the strike  $K$  ( $K \in \mathbb{R}^+$ ) and the payoff function  $\varphi : \mathbb{R}^{+3} \rightarrow \mathbb{R}$ ,

$$\varphi(x) = \left(K - \sum_{i=1}^3 x_i\right)^+, \quad x = (x_1, x_2, x_3)^T \in \mathbb{R}^{+3}.$$

The Black-Scholes price of  $P$  at time 0 is

$$P_0 = e^{-rT} E^* \left[ \left( K - \sum_{i=1}^3 S_{i,T} \right)^+ \right], \tag{7.2}$$

where  $E^*$  denotes the expectation with respect to the risk-neutral measure.

### 7.2 The uncoupled system

In order to combine different types of methods (Monte-Carlo, quadrature and/or PDE methods), we will uncouple the SDE in (7.1), we start with a change of variable to logarithmic prices. Let  $s_{i,t} := \log(S_{i,t})$ ,  $i = 1, 2, 3$ , and then Itô's lemma shows that

$$ds_{i,t} = r_i dt + dW_{i,t}, \quad t > 0 \tag{7.3}$$

with initial conditions  $s_{i,t=0} = s_{i,0} := \log(S_{i,0})$ . The parameters  $r_i$  ( $i = 1, 2, 3$ ) have been defined as  $r_i = r - \frac{a_{i1}^2}{2} - \frac{a_{i2}^2}{2} - \frac{a_{i3}^2}{2} = r - \frac{\sigma_i^2}{2}$ . In the rest of the section, the time index of any object is omitted to simplify the notation.

We note that (7.3) can be written as

$$\begin{pmatrix} ds_1 - r_1 dt \\ ds_2 - r_2 dt \\ ds_3 - r_3 dt \end{pmatrix} = \begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} dB_1 \\ dB_2 \\ dB_3 \end{pmatrix}.$$

Then, uncoupling reduces to Gaussian elimination. Using the Frobenius matrices

$$F_1 := \begin{pmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ -\frac{a_{31}}{a_{11}} & 0 & 1 \end{pmatrix}, \quad F_2 := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{a_{32}}{a_{22}} & 1 \end{pmatrix},$$

we write

$$F_2 F_1 (ds + r dt) = \text{Diag}(a_{11}, a_{22}, a_{33}) dB,$$

where  $s = (s_1, s_2, s_3)^T$ ,  $r = (r_1, r_2, r_3)^T$  and  $B = (B_1, B_2, B_3)^T$ . We set  $L^{-1} := F_2 F_1$ , and define

$$\tilde{s} := L^{-1} s \quad \text{and} \quad \tilde{S} := e^{L^{-1} s}.$$

**Remark 7.1** (i) The processes  $\tilde{s}_1, \tilde{s}_2$  and  $\tilde{s}_3$  are independent of each other, and are analogous with  $\tilde{S}_1, \tilde{S}_2$  and  $\tilde{S}_3$ , respectively.

(ii) Let  $\tilde{r} := L^{-1} r$ . Then

$$d\tilde{s} = \tilde{r} dt + \text{Diag}(a_{11}, a_{22}, a_{33}) dB.$$

(iii) The coupled system expressed in terms of the uncoupled system is  $s = L\tilde{s}$ .

(iv) In the next section, we will make use of the triangular structure of  $L = (L_{ij})_{i,j=1}^3$  and  $L^{-1} = ((L^{-1})_{ij})_{i,j=1}^3$ ,

$$L = \begin{pmatrix} 1 & 0 & 0 \\ \frac{a_{21}}{a_{11}} & 1 & 0 \\ \frac{a_{31}}{a_{11}} & \frac{a_{32}}{a_{22}} & 1 \end{pmatrix} \quad \text{and} \quad L^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ \frac{a_{21}a_{32}}{a_{11}a_{22}} - \frac{a_{31}}{a_{11}} & \frac{a_{32}}{a_{22}} & 1 \end{pmatrix}.$$

(v) The notation has been symbolic and the derivation heuristic.

### 7.3 Mixed methods

We describe nine combinations of Monte-Carlo, quadrature (or QUAD for short) and/or PDE methods.

**Convention** If  $Z$  is a stochastic process, we denote by  $Z^m$  a realization of the process. Let  $M'$  stand for a fixed number of Monte-Carlo samples.

#### Basic methods

(i) MC3 method

Simulate  $M'$  trajectories of  $(S_1, S_2, S_3)$ . An approximation of the option price  $P_0$  is

$$P_0^a := e^{-rT} \frac{1}{M'} \sum_{m=1}^{M'} \varphi(S_{1,T}^m, S_{2,T}^m, S_{3,T}^m).$$

(ii) QUAD3 method

In order to use a quadrature formula, we replace the risk neutral measure in

$$P_0 = e^{-rT} E^*[(K - e^{(L\tilde{s}_T)_1} - e^{(L\tilde{s}_T)_2} - e^{(L\tilde{s}_T)_3})^+]$$

by the Lebesgue-measure. Note

$$\tilde{s}_{i,t} \sim N(\mu_{i,t}, a_{ii}^2 t), \quad 1 \leq i \leq 3,$$

where  $\mu_{i,t} = \tilde{s}_{i,0} + \tilde{r}_i t$ . Let  $f_{i,t}$  be the density of  $\tilde{s}_{i,t}$ , i.e.,

$$f_{i,t}(x_i) = \frac{1}{\sqrt{2\pi a_{ii}} \sqrt{t}} e^{-\frac{1}{2} \left(\frac{x_i - \mu_{i,t}}{a_{ii} \sqrt{t}}\right)^2}, \quad x_i \in \mathbb{R}, \quad 1 \leq i \leq 3.$$

Due to the independence of  $\tilde{s}_{1,t}$ ,  $\tilde{s}_{2,t}$  and  $\tilde{s}_{3,t}$ , the density of

$$(K - e^{(L\tilde{s}_T)_1} - e^{(L\tilde{s}_T)_2} - e^{(L\tilde{s}_T)_3})^+$$

is

$$(x_1, x_2, x_3) \mapsto f_{1,T}(x_1) f_{2,T}(x_2) f_{3,T}(x_3), \quad (x_1, x_2, x_3) \in \mathbb{R}^3.$$

The formula for the option price becomes

$$P_0 = e^{-rT} \int_{\mathbb{R}^3} (K - e^{(Lx)_1} - e^{(Lx)_2} - e^{(Lx)_3})^+ f_{1,T}(x_1) f_{2,T}(x_2) f_{3,T}(x_3) dx.$$



Now, a quadrature formula can be used to compute the integral.

The methods, which are based on a combination of quadrature and some other methods, will be presented for the case, where the trapezoidal rule is used. Next we show how the trapezoidal rule can be used to compute the integral. This allows us to introduce the notation for the description of methods, which are combinations of quadrature and some other methods.

To compute the integral, we truncate the domain of integration to  $\kappa$  standard deviations around the means  $\mu_{1,T}$ ,  $\mu_{2,T}$  and  $\mu_{3,T}$ . Let

$$\begin{aligned} x_{i,0} &= \mu_{i,T} - \kappa a_{i,i}^2, \\ x_{i,n} &= x_{i,0} + n\delta x_i, \quad n = 1, \dots, N_Q, \end{aligned}$$

$1 \leq i \leq 3$ , where  $\delta x_i = \frac{2\kappa}{N_Q}$ ,  $N_Q$ .

The option price  $P_0$  is then approximated by

$$P_0^a := e^{-rT} \sum_{n_1, n_2, n_3=1}^N \left( \prod_{i=1}^3 \chi_{n_i} \delta x_i f_{i,T}(x_{i,n_i}) \right) (K - e^{(Lx_n)_1} - e^{(Lx_n)_2} - e^{(Lx_n)_3})^+,$$

where  $x_n := (x_{1,n_1}, x_{2,n_2}, x_{3,n_3})^T$  and

$$\chi_n = \begin{cases} 0.5, & \text{if } n = 0 \text{ or } n = N_Q, \\ 1, & \text{otherwise.} \end{cases}$$

(iii) MC2-PDE1 method (combination of two methods)

Note

$$\begin{aligned} P_0 &= e^{-rT} E^*[(K - S_{1,T} - S_{2,T} - S_{1,T}^{-2(L^{-1})_{31}} S_{2,T}^{-(L^{-1})_{32}} \tilde{S}_{3,T})^+] \\ &= e^{-rT} E^*[(\bar{K} - \tilde{S}_{3,T})^+], \end{aligned}$$

where

$$\bar{K} := K - S_{1,T} - S_{2,T},$$

and  $\tilde{S}_3$  is the solution to the stochastic initial value problem

$$\begin{aligned} d\tilde{S}_{3,t} &= \tilde{S}_{3,t}(\tilde{r}_3 dt + a_{33} dB_{3,t}), \\ \tilde{S}_{3,t=0} &= \alpha \tilde{S}_{3,0} \end{aligned}$$

with parameters  $\tilde{r}_3 := \tilde{r}_3 + \frac{a_{33}^2}{2}$  and  $\alpha = S_{1,T}^{-2(L^{-1})_{31}} S_{2,T}^{-(L^{-1})_{32}}$ .

The method is then as follows. Simulate  $M'$  realizations of  $(S_1, S_2)$  and set  $\bar{K}^m = K - S_{1,T}^m - S_{2,T}^m$  and  $\alpha^m = S_{1,T}^{m-2(L^{-1})_{31}} S_{2,T}^{m-(L^{-1})_{32}}$ . Compute an approximation of  $P_0$  by

$$P_0^a := \frac{1}{M'} \sum_{m=1}^{M'} u(x_3, t; \bar{K}^m) |_{x_3 = \alpha^m \tilde{S}_{3,0}, t=T},$$

where  $u$  is the solution to the initial value problem for the one-dimensional Black-Scholes PDE with the parametrized  $(\beta)$  initial condition

$$\frac{\partial u}{\partial t} - \frac{(a_{33}x_3)^2}{2} \frac{\partial^2 u}{\partial x_3^2} - \tilde{r}_3 x_3 \frac{\partial u}{\partial x_3} + \tilde{r}_3 u = 0 \quad \text{in } \Omega \times (0, T), \tag{7.4a}$$

$$u(t = 0) = u_0 \quad \text{in } \Omega, \tag{7.4b}$$

where  $\Omega = \mathbb{R}^+$  and

$$u_0(x_3; \beta) := (\beta - x_3)^+, \quad x_3 > 0.$$

(iv) QUAD2-PDE1 method

Note

$$P_0 = e^{-rT} \int_{\mathbb{R}^2} E^* [(K - e^{L_{11}x_1} - e^{L_{21}x_1 + L_{22}x_2} - e^{L_{31}x_1 + L_{32}x_2} e^{L_{33}\tilde{s}_{3,T}})^+] f_{1,T}(x_1) f_{2,T}(x_2) dx_1 dx_2.$$

The option price  $P_0$  is approximated by

$$P_0^a := \sum_{n_1, n_2=1}^{N_Q} \left( \prod_{i=1}^2 \chi_{n_i} \delta x_i f_{i,T}(x_{i,n_i}) \right) u(x_3, t; \bar{K}_{n_1 n_2})|_{x_3 = \alpha_{n_1 n_2} \tilde{S}_{3,0}, t=T},$$

where

$$\begin{aligned} \bar{K}_{n_1 n_2} &:= K - e^{L_{11}x_{1,n_1}} - e^{L_{21}x_{1,n_1} + L_{22}x_{2,n_2}}, \\ \alpha_{n_1 n_2} &:= e^{L_{31}x_{1,n_1} + L_{32}x_{2,n_2}}, \end{aligned}$$

and  $u$  denotes the solution to (7.4).

(v) MC1-PDE2 method

Note

$$P_0 = e^{-rT} E^* [(K - S_{1,T} - S_{2,T} - S_{1,T}^{-2(L^{-1})_{31}} S_{2,T}^{-(L^{-1})_{32}} \tilde{S}_{3,T})^+].$$

Simulate  $M'$  realizations of  $\tilde{S}_3$ . The option price  $P_0$  is then approximated by

$$P_0^a := \frac{1}{M'} \sum_{m=1}^{M'} u(x_1, x_2, t; \tilde{S}_{3,T}^m)|_{x_1=S_{1,0}, x_2=S_{2,0}, t=T},$$

where  $u$  denotes the solution to the initial value problem for the 2-dimensional Black-Scholes PDE with the parameterized  $(\beta)$  initial condition

$$u_0(x_1, x_2, 0; \beta) = (K - x_1 - x_2 - x_1^{-2(L^{-1})_{31}} x_2^{-(L^{-1})_{32}} \beta)^+, \quad x_1, x_2 > 0.$$

The problem is

$$\frac{\partial u}{\partial t} - \sum_{i,j=1}^2 x_i x_j \varrho_{ij} \frac{\partial u}{\partial x_i} \frac{\partial u}{\partial x_j} - r \sum_{i=1}^2 x_i \frac{\partial u}{\partial x_i} + ru = 0 \quad \text{in } \Omega \times (0, T), \tag{7.5a}$$

$$u(t = 0) = u_0 \quad \text{in } \Omega, \tag{7.5b}$$

where  $\Omega = \mathbb{R}^+ \times \mathbb{R}^+$  and

$$\varrho = (\varrho_{ij})_{i,j=1,\dots,3} = \frac{1}{2} \begin{pmatrix} a_{11}^2 & a_{11}a_{21} \\ a_{11}a_{21} & a_{21}^2 + a_{22}^2 \end{pmatrix}. \tag{7.6}$$

(vi) QUAD1-PDE2 method

Note

$$P_0 = e^{-rT} \int_{\mathbb{R}} E^*[(K - S_{1,T} - S_{2,T} - S_{1,T}^{-2(L^{-1})_{31}} S_{2,T}^{-(L^{-1})_{32}} e^{x_3})^+] f_{3,T}(x_3) dx_3.$$

With the notation above, another approximation of the option price  $P_0$  is

$$\begin{aligned} P_0^a &:= \sum_{n=1}^{N_Q} \delta x_3 f_{3,T}(x_{3,n}) e^{-rT} E^*[(K - S_{1,T} - S_{2,T} - S_{1,T}^{2(L^{-1})_{31}} S_{2,T}^{-(L^{-1})_{32}} e^{x_{3,n}})^+] \\ &= \sum_{n=1}^{N_Q} \delta x_3 f_{3,T}(x_{3,n}) u(x_1, x_2, t; x_{3,n})|_{x_1=S_{1,0}, x_2=S_{2,0}, t=T}, \end{aligned}$$

where  $u$  is the solution to the initial value problem (7.5).

(vii) MC1-QUAD2 method

Reformulating (7.2), we deduce

$$P_0 = e^{-rT} E^* \int_{\mathbb{R}^2} (K - e^{(Lx)_1} - e^{(Lx)_2} - e^{L_{31}x_1 + L_{32}x_2 + \tilde{s}_{3,T}})^+ f_{1,T}(x_1) f_{2,T}(x_2) dx_1 dx_2,$$

and obtain the following method.

Compute  $M'$  realizations of  $\tilde{s}_{3,T}$ , and approximate  $P_0$  by

$$\begin{aligned} P_0^a &:= e^{-rT} \frac{1}{M'} \sum_{n_1, n_2=1}^{N_Q} \sum_{m=1}^{M'} \left( \prod_{i=1}^2 \chi_{n_i} \delta x_i f_{i,T}(x_{i,n_i}) \right) \\ &\quad \cdot (K - e^{x_{1,n_1}} - e^{L_{21}x_{1,n_1} + x_{2,n_2}} - e^{L_{31}x_{1,n_1} + L_{32}x_{2,n_2} + \tilde{s}_{3,T}^m})^+. \end{aligned}$$

(viii) MC2-QUAD1 method

Note

$$P_0 = e^{-rT} \int_{\mathbb{R}} E^*[(K - S_{1,T} - S_{2,T} - S_{1,T}^{-2(L^{-1})_{31}} S_{2,T}^{-(L^{-1})_{32}} e^{x_3})^+] f_{3,T}(x_3) dx_3.$$

The method is as follows. Simulate  $M'$  realizations of  $(S_1, S_2)$ , and compute

$$\begin{aligned} P_0^a &:= e^{-rT} \frac{1}{M'} \sum_{m=1}^{M'} \sum_{n=1}^{N_Q} \chi_n \delta x_3 f_{3,T}(x_{3,n}) (K - S_{1,T}^m - S_{2,T}^m \\ &\quad - S_{1,T}^m S_{2,T}^{-2(L^{-1})_{31}} S_{2,T}^{-(L^{-1})_{32}} e^{x_{3,n}})^+. \end{aligned}$$

(ix) MC1-QUAD1-PDE1 method (combination of three methods)

Note

$$\begin{aligned} P_0 &= \int_{\mathbb{R}} f_{2,T}(x_2) e^{-rT} E^*[(K - e^{\tilde{s}_{1,T}} - e^{L_{21}\tilde{s}_{1,T} + x_2} \\ &\quad - e^{(-2(L^{-1})_{31} - (L^{-1})_{32}L_{21})\tilde{s}_{1,T} - (L^{-1})_{32}x_2} \tilde{S}_{3,T}^+)] dx_2. \end{aligned}$$

Then an approximation to  $P_0$  is

$$P_0^a := \frac{1}{M'} \sum_{m=1}^{M'} \sum_{n=1}^{N_Q} \chi_2 \delta x_2 f_{2,T}(x_{2,n}) u(x_3, t; \overline{K}_n^m) |_{x_3 = \alpha_n^m \tilde{S}_{3,0}, t=T},$$

where

$$\begin{aligned} \overline{K}_n^m &:= K - e^{\tilde{s}_{1,T}^m} - e^{L_{21} \tilde{s}_{1,T}^m + x_{2,n_2}}, \\ \alpha_n^m &:= e^{(-2(L^{-1})_{31} - (L^{-1})_{32} L_{21}) \tilde{s}_{1,T}^m - (L^{-1})_{32} x_{2,n}}, \end{aligned}$$

and  $u$  denotes the solution to (7.4).

### 7.4 Numerical results

This section provides a documentation of numerical results. We have considered European put options on baskets of three and five assets, and used mixed methods to compute their prices. If the method is stochastic, i.e., if a part of it is Monte-Carlo simulation, then we have run the method with different seed values several times ( $N_S$ ) and computed mean (m) and standard deviation (s) of the price estimates. If the method is deterministic, we have chosen the discretization parameters, such that the first three digits of  $P_0^a$  remained fix, while the discretization parameters have been further refined. Instead of solving the 1-dimensional Black-Scholes PDE, we have used the Black-Scholes formula.

(i) European put on three assets

The problem is to compute the price of a European put option on a basket of three assets in the framework outlined in Subsection 7.1.

We have chosen the parameters as follows:  $K = 150$ ,  $T = 1$ ,  $r = 0.05$ ,  $S_0 = (55, 50, 45)$ ,

$$\rho = \begin{pmatrix} 1 & -0.1 & -0.2 \\ -0.1 & 1 & -0.3 \\ -0.2 & -0.3 & 1 \end{pmatrix}, \quad \sigma = (0.3 \quad 0.2 \quad 0.25)^T.$$

We have used various (mixed) methods to compute approximations to  $P_0$  (see (7.2)).

We have used freefem++, and the rest is programmed in C++. The implementation in freefem++ requires a localization and the weak formulation of the Black-Scholes PDE. The triangulation of the computational domain and the discretization of the Black-Scholes PDE by conforming P1 finite elements are done by freefem++.

A reference result for  $P_0$  has been computed by using the Monte-Carlo method with  $10^7$  samples.

The numerical results are displayed in Table 4. One can see that the computational load for the PDE2 methods (i.e., MC1-PDE2, QUAD1-PDE2) is much larger than that for the other methods. Furthermore, the results seem to be less precise than those in the other cases. The results have been obtained very fast if just quadrature (i.e., QUAD3) or quadrature in combination with the Black-Scholes formula (i.e., QUAD2-PDE1) was used. In these cases, the results seem to be very precise although the discretization has been coarse ( $N_Q = 12$ ).

Table 4 Pricing a European put option on a basket of three assets, i.e., estimates of the option price at time 0. Columns 1–3: the method used to approximate  $P_0$ . Columns 4–6: the discretization parameters.  $M'$  is the number of Monte-Carlo samples,  $N_Q$  is the number of quadrature points,  $N_S$  is the number of samples used to compute the mean (m) and the standard deviation (s). Column 9: the computing time.

MC	PDE	QUAD	$M'$	$N_Q$	$N_S$	m	s	CPU
3	-	-	$10^7$	-	10	3.988	0.002	22.46
3	-	-	25000	-	100	3.994	0.046	0.147
2	1	-	25000	-	100	3.989	0.029	0.162
1	2	-	100	2601	10	3.886	0.195	372.5
-	-	3	-	-	-	3.984	-	0.005
-	1	2	-	-	-	3.987	-	0.005
-	2	1	-	2601	-	4.016	-	42.24
1	-	2	25000	-	100	3.991	0.022	2.723
2	-	1	25000	-	100	3.987	0.032	0.369
1	1	1	25000	-	100	3.990	0.023	0.514

Comparison of the results obtained by the MC3 method with the results obtained by the MC2-PDE1 method shows that the last mentioned seems to be superior. The computing time is about equal, but the standard deviation for MC2-PDE1 is much less than that for MC3.

(ii) European put on five assets

Let  $P$  be a European put option on a basket of five assets, with payoff

$$\varphi(x) = \left( K - \sum_{i=1}^5 x_i \right)^+.$$

The system of stochastic differential equations, which describes the dynamics of the underlying assets, has the usual form. We have set  $K = 250$ ,  $T = 1$ ,  $r = 0.05$ ,

$$\begin{aligned} S_0 &= (40, 45, 50, 55, 60)^T, \\ \sigma &= (0.3, 0.275, 0.25, 0.225, 0.2)^T, \\ \rho &= \begin{pmatrix} 1 & -0.37 & -0.40 & -0.44 & -0.50 \\ -0.37 & 1 & -0.50 & -0.46 & -0.05 \\ -0.40 & -0.50 & 1 & 0.51 & 0.29 \\ -0.44 & -0.46 & 0.51 & 1 & 0.20 \\ -0.50 & -0.05 & 0.29 & 0.20 & 1 \end{pmatrix}. \end{aligned}$$

We approximated the price of  $P$  at time 0 by various (mixed) methods. The results are displayed in Table 5. One can see that for all tested methods the (mean) price has been close ( $\pm 0.003$ ) to the reference price (1.159). Since  $N_Q = 10$  turned out to be enough, the computational effort has been very low for QUAD5 and QUAD4-PDE1. In the case, the method is stochastic, and deterministic methods allow to reduce the variance, such as in MC4-QUAD1 and MC4-PDE1-QUAD1.

Table 5 Pricing a European put option on a basket of five assets, i.e., estimates of the option price at time 0. Columns 1–3: the method used to approximate  $P_0$ . Columns 4–6: the discretization parameters.  $M'$  is the number of Monte-Carlo samples,  $N_Q$  is the number of quadrature points,  $N_S$  is the number of samples used to compute the mean (m) and the standard deviation (s). Columns 7–9: the numerical results. Column 7: the mean of  $P_0$ . Column 8: the standard deviation of  $P_0$ . Column 9: the computing time.

MC	PDE	QUAD	$M'$	$N_Q$	$N_S$	m	s	$\tau$
5	-	-	$10^7$	-	10	1.159	0.001	27.67
5	-	-	25000	-	100	1.161	0.019	0.162
4	-	1	25000	-	100	1.156	0.015	0.174
-	-	5	-	10	-	1.161	-	0.082
-	1	4	-	10	-	1.159	-	0.036
3	1	1	25000	10	100	1.158	0.013	0.442

## 8 Conclusion

Mixing Monte-Carlo methods with partial differential equations allows the use of closed formula on problems which do not have any otherwise. In these cases, the numerical methods are much faster than full MC or full PDE. The method works also for nonconstant coefficient models with and without jump processes and also for American contracts, although proofs of convergence have not been given here.

For multi-dimensional problems, we tested all possibilities of mixing MC and PDE and also quadrature on semi-analytic formula, and we found that the best is to apply PDE methods to one equation only.

The speed-up technique by polynomial fit has been discussed also, but we plan to elaborate on such ideas in the future particularly in the context of reduced basis, such as POD (proper orthogonal decomposition), ideally suited to the subproblems arising from MC+PDE, because the same PDE has to be solved many times for different time dependent coefficients.

## References

- [1] Achdou, Y. and Pironneau, O., Numerical Methods for Option Pricing, SIAM series, Philadelphia, USA, 2005.
- [2] Amin, K. and Khanna, A., Convergence of American option values from discrete- to continuous-time financial models, *Mathematical Finance*, **4**, 1994, 289–304.
- [3] Barth, A., Schwab, C. and Zollinger, N., Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients, *Num. Math.*, **119**(1), 2011, 123–161.
- [4] Bates, D. S., Jumps and stochastic volatility: Exchange rate processes implicit Deutsche mark options, *Review Financial Stud.*, **9**(1), 1996, 69–107.
- [5] Black, F. and Scholes, M., The pricing of options and corporate liabilities, *J. Political Econ.*, **81**, 1973, 637–659.
- [6] Boyle, P., Options: A Monte Carlo approach, *Journal of Financial Economics*, **4**, 1977, 323–338.
- [7] Carr, P. and Madan, D., Option valuation using the fast Fourier transform, *Journal of Computational Finance*, **2**(4), 1999, 61–73.
- [8] Dupire, B., Pricing with a smile, *Risk*, 1994, 18–20.
- [9] George, P. L. and Bouchouk, H., Delaunay triangulation and meshing, Hermès, Editions Paris, 1998; Application to Finite Elements, Translated from the original, P. J. Frey and S. A. Canann (eds.), French, 1997.

- [10] Glasserman, P., Monte-Carlo Methods in Financial Engineering, Stochastic Modeling and Applied Probability, Vol. 53, Springer-Verlag, New York, 2004.
- [11] Hecht, F., Pironneau, O., Le Yeric, A., et al., freefem++ documentation. <http://www.freefem.org>
- [12] Heston, S., A closed form solution for options with stochastic volatility with application to bond and currency options, *Review with Financial Studies*, **6**(2), 1993, 327–343.
- [13] Lewis, A., A simple option formula for general jump-diffusion and other exponential Lévy processes, 2001. <http://www.optioncity.net>
- [14] Loeper, G. and Pironneau, O., A mixed PDE/Monte-Carlo method for stochastic volatility models, *C. R. Acad. Sci. Paris, Ser. I*, **347**, 2009, 559–563.
- [15] Merton, R. C., Option pricing when underlying stock returns are discontinuous, *J. Financ. Econ.*, **3**, 1976, 125–144.
- [16] Pironneau, O., Dupire Identities for Complex Options, *Comptes Rendus de l'Académie des Sciences*, to appear.
- [17] Li, X. S., Demmel, J. W. and Gilbert, J. R., The superLU library. <http://crd.lbl.gov/xiaoye/SuperLU>
- [18] Wilmott, P., Howison, S. and Dewynne, J., *The Mathematics of Financial Derivatives*, Cambridge University Press, Cambridge, 1995.