MIXING MONTE-CARLO & PARTIAL DIFFERENTIAL EQUATIONS FOR PRICING OPTIONS

Tobias Lipp
LJLL-UPMC, Boite 187, Place Jussieu, 75252 PARIS cedex 5, France
Grégoire Loeper
Banque BNP-Parisbas, 20 Boulevard des Italiens 75009 PARIS, France
Olivier Pironneau
LJLL-UPMC, Boite 187, Place Jussieu, 75252 PARIS cedex 5, France

Abstract. There is a need for very fast option pricers when the financial objects are modeled by complex systems of stochastic differential equations. Here we 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 price the underlying asset by a PDE with random coefficients, derived by Ito calculus. This strategy is investigated for vanilla options, barrier and American options with stochastic volatilities and jumps optionally.

1. Introduction. Since the pioneering work of Phelim Boyle [6] Monte Carlo methods have entered and shaped mathematical finance like barely any other method. They are often appreciated due to their flexibility and applicability in high dimensions although they go hand in hand with 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 PDE methods. The key idea has been studied by two of the authors a few years ago in [12]: it tries to uncouples the underlying system of stochastic differential equations (SDEs) and then applies the last-mentioned methods appropriately.

In Section 2 we begin by 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.

Mixing MC and PDE for stochastic volatility models is formulated in section 3; a numerical evaluation of the method is made using closed form solutions of the

2000 Mathematics Subject Classification. Primary: 91B28, 65L60; Secondary: 82B31.
Key words and phrases. Monte-Carlo, Partial Differential Equations, Heston Model, financial mathematics, option pricing.

This article is dedicated to the memory of Jacques-Louis Lions.
PDE. In sections 4 and 5 the method is extended to American option and to the case where the underlying asset is model with jump-diffusion processes.

In section 6 a method to reduce the number of samples is given based on the weak dependence of the option price on the initial vol-vol of the stochastic volatility models.

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 [11]: under a risk neutral probability, the risky asset \( S_t \), the volatility \( \sigma_t \) and the option price \( P_t \) follow the diffusion process:

\[
\begin{align*}
    dS_t &= S_t(rdt + \sigma_t dW_t^1), \\
    dv_t &= k(\theta - v_t)dt + \delta \sqrt{v_t}dW_t^2, \\
    P_t &= e^{-r(T-t)}E[(K - S_T)^+|S_t, v_t]
\end{align*}
\]

with \( v_t = \sigma_t^2 \), and \( E(dW_t^1 \cdot dW_t^2) = \rho dt \), \( E(\cdot) \) denoting expectation with respect to the risk neutral measure and \( r \) the interest rate on a risk less commodity. The pair \( W^1, W^2 \) is a two-dimensional correlated Brownian motion, the correlation between the two components being equal to \( \rho \). As it is usually observed in equity option markets, options with low strikes have a higher implied volatility than at the money or high strikes options, 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 \( \delta t \), and denoting by \( T \) the maturity of the option, we have \( T = N\delta t \). Full Monte-Carlo simulation (see [9]) consists in a time loop starting at \( S_0, v_0 = \sigma_0^2 \) of

\[
\begin{align*}
    v_{i+1} &= v_i + k(\theta - v_i)\delta t + \sigma_i\sqrt{\delta t}N_{i,1}\delta \\
    S_{i+1} &= S_i \left(1 + r\delta t + \sigma_i\sqrt{\delta t}(N_{i,1}\rho + N_{i,1}^2\sqrt{1-\rho^2})\right)
\end{align*}
\]

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

The method is slow, at least 300 000 samples are necessary for a precision of 0.1%. Of course acceleration methods exists (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 (see [1]), then \( u(S, y, \tau) \) is given by

\[
\begin{align*}
    \partial_\tau u - \frac{yS^2}{2}\partial_{SS}u - \rho \lambda S y \partial_{Sv}u - \frac{\lambda^2 y}{2} \partial_{yy}u - r S \partial_S u - k(\theta - y) \partial_y u + ru = 0 \\
    u(S, y, 0) = (K - S)^+
\end{align*}
\]

Now instead of integrating (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 (4), (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 (MC) 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 on figure 1 validates the methodology and it may be attractive to use it to obtain more precision on a small domain.

3. Monte-Carlo Mixed with a one dimensional PDE. Let us rewrite (1) as

$$dS_t = S_t \left[ \mu_t dt + \sigma_t \sqrt{1 - \rho^2} \tilde{W}_t^{(1)} + \sigma_t \rho \tilde{W}_t^{(2)} \right],$$

(7)

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

Having drawn a trajectory of $v_t$ by (4), consider, with the same $\delta t$:

$$dS_t = S_t \left[ \mu_t dt + \sigma_t \sqrt{1 - \rho^2} \tilde{W}_t^{(1)} \right],$$

(8)

$$\mu_t = r + \rho \sigma_t \frac{W^{(2)}(t + \delta t) - W^{(2)}(t)}{\delta t} - \frac{1}{2} \rho^2 \sigma_t^2.$$  

(9)

**Proposition 1.** As $\delta t \to 0$, $S_t$ given by (4), (8),(9) converges to the solution of Heston’s model (1)(2). Moreover the put $P = e^{-T} \mathbb{E}(K - S_T)^+$ is also the expected value of $u(S_0, 0)$, $u$ given by

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

(10)

with $\sigma_t$ given by (4) and $\mu_t$ given by (9).

**Proof:** By Ito’s formula we have

$$d \log(S_t) = \frac{dS_t}{S_t} + \frac{1}{2} \left( \log S \right)'' (S_t^2 \sigma_t^2 (1 - \rho^2) dt) = \frac{dS_t}{S_t} - \frac{1}{2} \sigma_t^2 (1 - \rho^2) dt$$

$$= \mu_t dt + \sqrt{1 - \rho^2} \sigma_t d\tilde{W}_t^{(1)} - (1 - \rho^2) \frac{\sigma_t^2}{2} dt$$

$$= r dt + \rho \sigma_t dW_t^{(2)} - \frac{\rho^2 \sigma_t^2}{2} dt + \sqrt{1 - \rho^2} \sigma_t d\tilde{W}_t^{(1)} - (1 - \rho^2) \frac{\sigma_t^2}{2} dt$$

$$= r dt + \rho \sigma_t dW_t^{(2)} + \sqrt{1 - \rho^2} \sigma_t d\tilde{W}_t^{(1)} - \frac{\sigma_t^2}{2} dt$$

(11)

Consequently

$$S_t = S_0 \exp \left( \int_0^t \mu_t dt + \int_0^t \sqrt{1 - \rho^2} \sigma_t d\tilde{W}_t^{(1)} - \int_0^t \frac{1}{2} (1 - \rho^2) \sigma_t^2 dt \right)$$

(12)

**Proposition 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 (10), (4), (9) are well posed.

**Proof:** Let

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

(13)

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

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

(14)
Figure 1. Put option with Heston’s model computed by solving the PDE by implicit Euler + FEM using the public domain package freefem++[10]. Top: the computational domain is $(0, y_{max}) \times (0, v_{max})$ with Neumann condition at $v = v_{max}$. Middle: the computational domain is the half circle shown on the top figure; the Dirichlet boundary condition on the circle is obtained by a spline approximation (shown at the bottom) of the solution of Heston’s model solved by MC on a few points on the circle.
If \( 0 < \sigma_m \leq \sigma_t \leq \sigma_M \) almost surely and for all \( t \), then the solution exists in the sense of A. Barth et al [3].

**Remark 1.** Note that (12), is also

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

\[
S_T(x) = S_0 \exp\left(mT + \bar{\sigma}Tx\right)
\]

Therefore

\[
\mathbb{E}[u(S_0, 0)] = e^{-rT} \int_{\mathbb{R}^+} \left(K - S_0 e^{mT + \bar{\sigma}Tx}\right) + e^{-\frac{\bar{\sigma}^2}{2T}} dx
\]

There is a closed form for this integral, namely the Black-Scholes formula with interest rate \( r \), dividend \( m \) and 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 (4). The Black-Scholes formula is used as indicated in Remark 1. To observe the precision with respect to \( \rho \) we have taken a large number of Monte-Carlo samples: \( M = 3 \times 10^5 \) and \( M' = 10^4 \). Similarly the number of time steps is 300 with 400 mesh points and \( S_{max} = 600 \), (i.e. \( \delta S = 1.5 \)).

**Table 1.** Precision versus \( \rho \)

<table>
<thead>
<tr>
<th>( \rho )</th>
<th>-0.5</th>
<th>0</th>
<th>0.5</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heston MC</td>
<td>11.135</td>
<td>10.399</td>
<td>9.587</td>
<td>8.960</td>
</tr>
<tr>
<td>Heston MC+BS</td>
<td>11.102</td>
<td>10.391</td>
<td>9.718</td>
<td>8.977</td>
</tr>
<tr>
<td>Speed-up</td>
<td>42</td>
<td>44</td>
<td>42</td>
<td>42</td>
</tr>
</tbody>
</table>

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 for the mixed strategy MC+BS. This variance reduction explains why MC+BS is much faster.

**Table 2.** Precision study with respect to \( M \) and \( M' \). Five realizations of pure MC and MC+EDP for various \( M' \) and \( M \)

<table>
<thead>
<tr>
<th>( P )</th>
<th>MC+BS: ( M' = 100 )</th>
<th>MC+BS: ( M' = 1000 )</th>
<th>MC+BS: ( M' = 10000 )</th>
<th>MC: ( M = 3000 )</th>
<th>MC: ( M = 30000 )</th>
<th>MC: ( M = 300000 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P^1 )</td>
<td>10.475</td>
<td>11.129</td>
<td>11.100</td>
<td>11.564</td>
<td>11.481</td>
<td>11.169</td>
</tr>
<tr>
<td>( P^2 )</td>
<td>10.436</td>
<td>11.377</td>
<td>11.120</td>
<td>11.6978</td>
<td>11.409</td>
<td>11.249</td>
</tr>
<tr>
<td>( P^3 )</td>
<td>11.025</td>
<td>11.528</td>
<td>11.113</td>
<td>11.734</td>
<td>11.383</td>
<td>11.143</td>
</tr>
<tr>
<td>( P^4 )</td>
<td>11.205</td>
<td>11.002</td>
<td>11.113</td>
<td>11.565</td>
<td>11.482</td>
<td>11.169</td>
</tr>
<tr>
<td>( P^5 )</td>
<td>11.527</td>
<td>11.360</td>
<td>11.150</td>
<td>11.085</td>
<td>11.519</td>
<td>11.208</td>
</tr>
<tr>
<td>( P = \frac{1}{5} \sum_i P_i )</td>
<td>10.934</td>
<td>11.279</td>
<td>11.119</td>
<td>11.529</td>
<td>11.454</td>
<td>11.187</td>
</tr>
<tr>
<td>( \sqrt{\frac{1}{5} \sum (P^i - P)^2} )</td>
<td>0.422</td>
<td>0.188</td>
<td>0.0168</td>
<td>0.232</td>
<td>0.0507</td>
<td>0.0370</td>
</tr>
</tbody>
</table>
4. Extension to American Options. For American options we must proceed step by step backward in time as in dynamic programming for binary trees for American options [2]. Consider \( M' \) realizations \( \{\sigma^m_t\}_{t \in (0,T)}^{M'} \), giving \( \{\mu^m_t\}_{t \in (0,T)}^{M'} \) by (9). 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 at \( t_{n-1} \) and \((K - S)^+\), i.e.

\[
 u_{n-1}(S) = \max\left\{\frac{1}{M'} \sum_m u^m_{n-1}(S), (K - S)^+\right\}
\]  

(18)

where \( u^m_{n-1} \) is the solution at \( t_{n-1} \) of

\[
\begin{align*}
\partial_t u + (1 - \rho^2) \frac{(S\sigma^m_t)^2}{2} \partial_{SS} u + S\mu^m_t \partial_S u - ru &= 0, \ t \in (t_{n-1}, t_n); \\
u_n := u(S, t_n) &\text{ known}
\end{align*}
\]

(19)

As with American options with binary trees, convergence with optimal order will hold only if \( \delta t \) is small enough. A numerical result with this procedure is shown on figure 2.

![Figure 2](image-url)

Figure 2. An American option at \( T = 1 \) with Heston’s model compared with the same European contract. The broken line \( S \rightarrow (K - S)^+ \) is drawn too for reference.

5. Levy Processes. Consider Bates model [4], i.e. an asset modeled with Stochastic Volatility and a jump process:

\[
\begin{align*}
dv_t &= k(\theta - v_t)dt + \xi \sqrt{v_t}dW_t^{(2)}, \ \sigma_t = \sqrt{v_t} \\
dX_t &= (r - \frac{\sigma^2_t}{2})dt + \sigma_t(\sqrt{1 - \rho^2}d\tilde{W}_t^{(1)} + \rho d\tilde{W}_t^{(2)}) + \eta dN_t
\end{align*}
\]

(20)  

(21)

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

\[
\begin{align*}
dX_t &= \tilde{\mu}_t dt + \sigma_t \sqrt{1 - \rho^2}d\tilde{W}_t^{(1)} + \eta dN_t \\
\tilde{\mu}_t &= r - \frac{\sigma^2_t}{2} + \rho \sigma_t \frac{\delta W^{(2)}}{\delta t}
\end{align*}
\]

(22)  

(23)
Remark 3. Let \( \bar{\mu} \) be Itô, a put on \( S_t \) with \( u(T) = (K - e^z)^+ \) satisfies

\[
\partial_t u - ru + \frac{1}{2}(1 - \rho^2)\sigma^2 \partial_{xx} u + \bar{\mu} \partial_x u = -\int_{\mathbb{R}} \left( (u(x + z) - u(x))J(z) - \partial_x u(x)(e^z - 1)J(z) \right) dz
\]

(24)

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

\[
v(y, \tau) = e^{(r + f_h)(J(z)dz)\tau} u(y + \int_{T-\tau}^T \bar{\mu}_t dt, T - \tau)
\]

(25)

Proposition 3.

\[
\partial_\tau v - \frac{1}{2}(1 - \rho^2)\sigma^2 \partial_{y y} v - \int_{\mathbb{R}} v(y + z)J(z)dz = 0, \quad v(y, 0) = (K - e^y)^+
\]

(26)

Proof

Let \( \bar{r} = r + \int_{\mathbb{R}} J(z)dz \), then

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

(27)

Therefore

\[
e^{-\bar{r}\tau} \left[ \partial_\tau v - \frac{1}{2}(1 - \rho^2)\sigma^2 \partial_{yy} v - \int_{\mathbb{R}} v(y + z)J(z)dz \right]
\]

\[
= (r + \int_{\mathbb{R}} J(z)dz)u + \bar{\mu}_T \partial_x u - \partial_t u - (1 - \rho^2)\sigma^2 \partial_{xx} u - \int_{\mathbb{R}} u(x + z)J(z)dz
\]

which is zero by (24).

Remark 2. Once more we notice that the PDE depends on time integrals of \( \bar{\mu}_t \) and integrals of \( \sigma_t \) and integrals damp the randomness and make PDE (26) easier to solve. Table 3 displays 9 realizations of \( \sqrt{\frac{1}{T} \int_0^T \sigma^2_t} \) for \( M' = 100 \) and 500.

<table>
<thead>
<tr>
<th>( M' )</th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>T5</th>
<th>T6</th>
<th>T7</th>
<th>T8</th>
<th>T9</th>
<th>mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>3470</td>
<td>.3482</td>
<td>.3496</td>
<td>.3484</td>
<td>.3474</td>
<td>.3548</td>
<td>.3492</td>
<td>.3502</td>
<td>.3493±.002</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>.3490</td>
<td>.3481</td>
<td>.3488</td>
<td>.3493</td>
<td>.3502</td>
<td>.3501</td>
<td>.3499</td>
<td>.3488</td>
<td>.3493±.0007</td>
<td></td>
</tr>
</tbody>
</table>

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

Remark 3. Let \( \bar{f}_\tau = \frac{1}{\tau} \int_{T-\tau}^T f(t)dt \). From (??) we see that the option price is recovered by

\[
u(S, t) = e^{-(r + f_h)(J(z)dz)(T-t)}v(\ln S - (r - \frac{\sigma^2}{2} - \int_{\mathbb{R}} (e^z - 1)J(z)dz)T-t, T-t)
\]

where \( v \) is solution of (26). For a European put option with the standard diffusion-Lévy process model and dividend \( q \) the formula is

\[
u(S, t) = e^{-(r + f_h)(J(z)dz)(T-t)}v(\ln S - (r - q - \frac{\sigma^2}{2} - \int_{\mathbb{R}} (e^z - 1)J(z)dz)(T-t), T-t)
\]
\[ \partial_x v - \frac{1}{2} \sigma^2 \partial_{yy} v - \int_R v(y + z) J(z) dz = 0, \quad v(y, 0) = (K - e^y)^+ \] \tag{28}

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

1. In the solver change \( \sigma^2 \) into \( (1 - \rho^2) \sigma^2 |t| \)
2. Change \( q \) into \( q + \rho^2 \sigma^2 |t| - \rho \sigma_t \Delta W(t) |t| \)

5.1. Numerical Solution of the PIDE by Spectral Method. Let the Fourier transform operators be

\[ F(u) = \int_R e^{-i \omega x} u(x) dx \quad \text{and} \quad F^{-1}(\hat{u}) = \frac{1}{2\pi} \int_R e^{i \omega x} \hat{u}(\omega) d\omega \] \tag{29}

Applying the operator \( F \) to the PIDE (26) for a call gives

\[ \partial_x \hat{v} - \Psi \hat{v} = 0 \quad \text{in} \quad \mathbb{R}, \quad \hat{v}(\omega, 0) = F(e^x - K)^+ \] \tag{30}

where \( \Psi(\omega) = -(1 - \rho^2) \sigma^2 \omega^2 - \varphi(\omega), \quad \varphi(\omega) = \int \hat{F} e^{i \omega x} J(y) dy \] \tag{31}

So, \( m \) indicating a realization, the solution is

\[ u(x - \int_{T-\tau}^{T} \tilde{\mu}_t dt) = \frac{1}{M^2} \sum \frac{e^{-\tau T}}{m} \left( K - F^{-1}[\{F \hat{u}(\omega) = e^{-(\omega^2 + \kappa^2) T}\} e^{-\varphi(\omega) T} \right) \] \tag{32}

with \( \tilde{\mu}_t \) given by (9) and \( \tilde{\mu}_t = \tilde{\mu}_t + \int_{\mathbb{R}} (e^z - 1) J(z) dz \).

Remark 4. The Car-Madan trick [?1] must be used and \( u^0 \) must be replaced by \( e^{-\eta S}(S - K)^+ \) which has a Fourier transform, in case of a call option. Then in (??) \( F^{-1} \hat{\chi} \) must be changed into

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

Remark 5. As an alternative to FFT methods, following Lewis [?], for a call option, when \( 3 \omega > 1 \)

\[ F u^0 = F(e^y - K)^+ = -\frac{e^{\ln K (i \omega + 1)}}{\omega^2 - i \omega} \] \tag{33}

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

\[ u(S, T) = S - \sqrt{\frac{KS}{\pi}} \int_0^\infty \Re[e^{i u k} \phi_T(u - \frac{i}{2})] \frac{du}{u^2 + \frac{1}{4}} \] \tag{34}

with \( k = \ln \frac{K}{S} \) and where \( \phi_T \) is the characteristic function of the process, which, in the case of (26) with Merton Kernel [?]

\[ J(x) = \lambda e^{-\frac{(x - \mu)^2}{2\sigma^2}} \sqrt{2\pi \delta^2} \] is

\[ \phi_T(u) = \exp \left( i u \delta T - \frac{1}{2} u^2 \Sigma^2 T + T \lambda (e^{-\delta^2 u^2 / 2 + i u} - 1) \right) \]

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

\[ T = 1, \quad \mu = -0.5, \quad \lambda = 0.1, \quad \delta = 0.4, \quad K = 1, \quad r = 0.03, \quad \sigma_0 = 0.4, \quad \theta = 0.4, \quad \kappa = 2, \]
\[ \rho = -0.5, \xi = 0.25, \, M' = 10000, \, \delta t = 0.001. \]  \hspace{1cm} (35)

Results for a put are reported on figure ???. The method is not precise out of the money, i.e. \( S > K \). CPU is 0.8" per points on the curve.

\begin{figure}[!h]
\centering
\includegraphics[width=\textwidth]{fig3.png}
\caption{Put calculated with Bates' model by mixing MC with Lewis' formula. (??)}
\end{figure}

\begin{figure}[!h]
\centering
\includegraphics[width=\textwidth]{fig4.png}
\caption{Call calculated by a Heston+Merton-Lévy by mixed MC-Fourier (blue) and compared with the solution of the 2D PIDE Black-Scholes+Levy (red) and a pure Black-Scholes (green).}
\end{figure}
5.2. 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 \( \sim 0.27 \). For the Heston and for 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 on figure 3.

6. Reduction of the Number of MC Samples. If the full surface \( \sigma_0, S_0 \to u(\sigma_0, S_0, 0) \) is required, MC+PDE becomes prohibitively expensive, much like MC is too expensive if \( S_0 \to u(S_0, 0) \) is required for all \( S \).

However notice that after some time \( t_1 \) the SDE for \( \sigma_t \) will have generated a large number of sample values \( \sigma_1 \). Let us take advantage of this to compute \( u(\sigma_1, S_1, t_1) \).

6.1. Polynomial Fits. Let \( \tau = T - t_1 \), for some fixed \( t_1 \).

Instead of gathering all \( u(\cdot, \tau) \) corresponding to the samples \( \sigma_t^{m} \) with the same initial value \( \sigma_0 \) at \( t = 0 \), we focus on the time interval \( (t_1, T) \) and consider that \( \sigma_t^{m} \) is a stochastic volatility initiated by \( \sigma_t^{(m)} \) and we 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

\[
\min_{\alpha} J(\alpha) := \frac{1}{M} \sum_{m} \frac{1}{L} \int_{0}^{L} \| \sum_{k} \alpha_k(S) \phi_k^{(m)}(\sigma_T) - u(\sigma_T^{(m)}; S, \tau) \|^2 dS
\]

It leads to solve for each \( S_i = i\delta S \)

\[
\left( \frac{1}{M} \sum_{m} \phi_k(\sigma_T^{(m)}) \phi_l(\sigma_T^{(m)}) \right) \alpha_k^l = \frac{1}{M} \sum_{m} u(\sigma_T^{(m)}; S_i, \tau) \phi_l(\sigma_T^{(m)})
\]

(36)

6.2. 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 0 otherwise.

Algorithm 1

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

6.3. Polynomial Projection. Now we chose \( \phi_k(\sigma) = \sigma^k \).

Algorithm 2

1. Choose \( \sigma_m \), \( \sigma_M \), \( \delta \sigma \), \( \sigma_0 \)
2. Set \( A[::] = 0 \); set \( b[::] = 0 \).
3. Compute \( M \) realizations \( \{\sigma_t^{(m)}\} \) by MC on the vol equation and for each realization

- compute \( u(\cdot, \tau) \) by solving the PDE
- do \( A[j][k] + = \frac{1}{M} \sum_{m} (\sigma_T^{(m)})^{j+k} \), \( j, k = 1..K \);
- and do \( b[i][k] + = \frac{1}{M} u(i\delta S, \tau)(\sigma_T^{(m)})^k \), \( k = 1..K \)
4. The answer is found by solving (32) for each \( i = 1..N \).
6.4. **Numerical Test.** A Vanilla put with the same characteristics as in subsection 3.1 has been computed by algorithm 2, for a maturity of 3 years. The surface $S_{t_1}, \sigma_{t_1} \to u$ is shown after $t_1 = 1.5$ years in figure 4. The implied volatility is also shown.

![Put at 1.5 with Heston by mixed H-F Monte Carlo algorithm](image1)

**Figure 5.** Top: 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. Bottom: comparison on the price of the put computed with full MC Heston. Both surface are on top of each other, indistinguishable.

7. **Systems of Dimension Greater than 2.** Stochastic volatility models with several SDE 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: for $i = 1, 2, 3$

$$dS_{i,t} = S_{i,t}(rdt + dW_{i,t}), \quad t > 0$$

(37)

with initial condition $S_{i,t=0} = S_{i,0}$, $S_{i,0} \in \mathbb{R}^+$. 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

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

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

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

$$\text{Cov}[W_{i,t},W_{j,t}] = E[W_{i,t}W_{j,t}]$$

(39)

or equivalently such that

$$AA^T = \Xi$$

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

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

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

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

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

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

(38)

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 SDEs in (33). We start with a change of variable to logarithmic prices. Let $s_{i,t} := \log(S_{i,t})$, $i = 1, 2, 3$, then Itô’s lemma shows that

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

(39)
with initial condition $s_{i,t=0} = s_{i,0} := \log(S_{i,0})$. The parameters $r_i$, $i = 1, 2, 3$, have
been defined as $r_i = r - \frac{a_i^2}{2} - \frac{\sigma_i^2}{2} - \frac{\sigma_i^2}{2} = r - \frac{\sigma_i^2}{2}$. In the rest of the section the time
index of any object is omitted to simplify the notation.

We note that equation (35) 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

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

we write

$$F_2F_1(ds + rd) = \text{Diag}(a_{11}, a_{22}, a_{33})dB$$

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 $L^{-1} := F_2F_1$ and define

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

Remark 6. i) The processes $\tilde{s}_1$, $\tilde{s}_2$ and $\tilde{s}_3$ are independent of each other; analog
$\tilde{S}_1$, $\tilde{S}_2$ and $\tilde{S}_3$.
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 $L = (L_{ij})^3_{i,j=1}$
and $L^{-1} = ((L^{-1})_{ij})^3_{i,j=1}$.

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

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

7.3. Mixed methods. We describe nine combinations of Monte Carlo, quadrature
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

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

ii) QUAD3 method: In order to use a quadrature formula we replace the risk
neutral measure in

$$P_0 = e^{-rT}E^* \left[ (K - e^{(L\tilde{s})_1} - e^{(L\tilde{s})_2} - e^{(L\tilde{s})_3})^+ \right]$$
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
\[ P_0 = e^{-rT} \int_{\mathbb{R}^3} \left( K - e^{(Lx)_1} - e^{(Lx)_2} - e^{(Lx)_3} \right)^+ f_{1,T}(x_1)f_{2,T}(x_2)f_{3,T}(x_3) \, dx. \]

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

The methods which are based on a combination of quadrature and some other
method 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 method(s).

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
\[ x_{i,0} = \mu_{i,T} - \kappa a_{ii}^2 \]
\[ x_{i,n} = x_{i,0} + n \delta x_i, \quad n = 1, \ldots, N_Q, \]
\[ 1 \leq i \leq 3, \quad \delta x_i = \frac{2\kappa}{N_Q}, \quad N_Q. \]

The option price \( P_0 \) is then approximated by
\[ P_0^a := e^{-rT} \sum_{n_1,n_2,n_3=1}^{N_Q} \left( \prod_{i=1}^{3} \chi_{n_i} \delta x_i f_{i,T}(x_{i,n_i}) \right) \left( K - e^{(Lx_n)_1} - e^{(Lx_n)_2} - e^{(Lx_n)_3} \right)^+ \]
where \( x_n := (x_{1,n_1}, x_{2,n_2}, x_{3,n_3})^T \) and
\[ \chi_n = \begin{cases} 1 & \text{otherwise.} \\
0.5 & \text{if } n = 0 \text{ or } n = N_Q \end{cases} \]

**Combination of two methods.** iii) **MC2-PDE1 method:** Note,
\[ P_0 = e^{-rT} E^* \left[ \left( K - S_{1,T} - S_{2,T} - S_{3,T}^2 \tilde{s}_{3,T} \right)^+ \right] \]
\[ = e^{-rT} E^* \left[ \left( \tilde{K} - \tilde{s}_{3,T} \right)^+ \right] \]
where \( \tilde{K} := K - S_{1,T} - S_{2,T} \).
and $\tilde{S}_3$ is the solution of the stochastic initial value problem

$$d\tilde{S}_{3,t} = \tilde{S}_{3,t} \left( \tilde{r}_3 dt + a_{33} dB_{3,t} \right)$$

$$\tilde{S}_{3,t=0} = \alpha \tilde{S}_{3,0}$$

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

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

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

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

$$\frac{\partial u}{\partial t} - \frac{(a_{33} x_3)^2}{2} \frac{\partial^2 u}{\partial x_3^2} - \tilde{r}_3 x_3 \frac{\partial u}{\partial x_3} + \tilde{r}_3 u = 0 \quad \text{in } \Omega \times (0, T) \quad (40a)$$

$$u(t = 0) = u_0 \quad \text{in } \Omega \quad (40b)$$

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

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

iv) QUAD2-PDE1 method: Note,

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

The option price $P_0$ is approximated by

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

where

$$\tilde{K}_{n_1, n_2} := K - e^{L_{11} x_1, n_1} - e^{L_{21} x_1, n_1 + L_{22} x_2, n_2},$$

$$\alpha_{n_1, n_2} := e^{L_{31} x_1, n_1 + L_{32} x_2, n_2}$$

and $u$ denotes the solution of (36).

v) MC1-PDE2 method: Note,

$$P_0 = e^{-rT} E^* \left[ (K - S_{1,T} - S_2 - S_{1,T}^{-2(L-1)} S_{2,T}^{-2(L-1)}) \tilde{S}_{3,T} \right].$$

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

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

where $u$ denotes the solution of the initial value problem for the two dimensional Black-Scholes PDE with parametrized ($\beta$) initial condition

$$u_0(x_1, x_2; \beta) = (K - x_1 - x_2 - x_1^{-2(L-1)} x_2^{-2(L-1)} \beta)^+, \quad x_1, x_2 > 0.$$
vi) QUAD1-PDE2 method: Note,

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

With the notation from above another approximation of the option price \( P_0 \) is

\[ P_0^a = \sum_{n=1}^{N_Q} \delta x_3 f_{3,T}(x_3,n) e^{-rT} E^* \left[ \left( K - S_{1,T} - S_{2,T} - S_{1,T}^{2(L^{-1})^31} S_{2,T}^{2(L^{-1})^32} e^{x_3,n} \right)^+ \right] \]

\[ = \sum_{n=1}^{N_Q} \delta x_3 f_{3,T}(x_3,n) u(x_1, x_2, t; x_3,n)|_{x_1=S_{1,0}, x_2=S_{2,0}, t=T} \]

where \( u \) is the solution of the initial value problem (37).

vii) MC1-QUAD2 method: Reformulating equation (34) we deduce

\[ P_0 = e^{-rT} E^* \int_{\mathbb{R}^2} \left( K - e^{(Lx_1)1} - e^{(Lx_2)2} - e^{L_1x_1 + L_3x_2 + \delta_3,T} \right)^+ f_{1,T}(x_1)f_{2,T}(x_2)dx_1dx_2 \]

and obtain the following method:

Compute \( M' \) realizations of \( \delta_3,T \) and approximate \( P_0 \) by

\[ P_0^a = e^{-rT} \frac{1}{M'} \sum_{n_1,n_2=1}^{N_Q} \sum_{m=1}^{M'} \left( \prod_{i=1}^{2} \chi_{n_i} \delta x_i f_{i,T}(x_i,n_i) \right) \]

\[ \left( K - e^{x_{1,n_1} - e^{L_2 x_{1,n_1} + x_{2,n_2}} - e^{L_1 x_{1,n_1} + L_3 x_{2,n_2} + \delta_3,T}} \right)^+. \]

viii) MC2-QUAD1 method: Note,

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

The method is: simulate \( M' \) realizations of \( (S_1, S_2) \) and compute

\[ P_0^a = e^{-rT} \frac{1}{M'} \sum_{m=1}^{M'} \sum_{n=1}^{N_Q} \chi_n \delta x_3 f_{3,T}(x_3,n) \]

\[ \left( K - S_{1,T}^m - S_{2,T}^m - S_{1,T}^{m-2(L^{-1})^31} S_{2,T}^{m-2(L^{-1})^32} e^{x_3,n} \right)^+. \]
Combination of three methods. ix) MC1-QUAD1-PDE1 method: Note,

\[
P_0 = \int_\mathbb{R} f_{2,T}(x_2)e^{-rT}E^\ast\left[(K - e^{\tilde{s}_1,T} - e^{L_{21}\tilde{s}_1,T+x_2}
- e^{-(2(L^{-1})_{31}-(L^{-1})_{32}L_{21})\tilde{s}_1,T-(L^{-1})_{32}x_2}\tilde{s}_{3,T}^m}\right]dx_2.
\]

An approximation to \(P_0\) is then

\[
P_0^a := \frac{1}{M} \sum_{m=1}^{M'} \sum_{n=1}^{N_Q} \chi(x_2) f_{2,T}(x_2,n)u(x_3,t; \tilde{K}^m_n) |_{x_3=\alpha^m_n} \tilde{s}_{3,0,t=T}
\]

where

\[
\tilde{K}^m_n := K - e^{\tilde{s}_1,T} - e^{L_{21}\tilde{s}_1,T+x_2,n}
\]

\[
\alpha^m_n := e^{-(2(L^{-1})_{31}-(L^{-1})_{32}L_{21})\tilde{s}_1,T-(L^{-1})_{32}x_2,n}
\]

and \(u\) denotes the solution of (36).

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 price. If the method is stochastic, i.e. if 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 solving the one-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 asset in the framework outlined in §7.1. We have chosen the parameters as follows: \(K = 150, T = 1, r = 0.05, S_0 = (55, 50, 45), \)

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

We have used various (mixed) methods to compute approximations to \(P_0\) (see (34)). We have used \texttt{freefem++}; the rest is programmed in \texttt{C++}. The implementation in \texttt{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 is done by \texttt{freefem++}.

A reference result for \(P_0\) has been computed 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 (MC1-PDE2, QUAD1-PDE2) is much larger than for the other methods. Furthermore the results seem to be less precise than in the other cases. The results have been obtained very fast if just quadrature (QUAD3) or quadrature in combination with the Black-Scholes formula (QUAD2-PDE1) has
been used. In these cases the results seem to be very precise although the discretization has been coarse ($N_Q = 12$). 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 is for MC2-PDE1 much less than 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 $K = 250$, $T = 1$, $r = 0.05$, $S_0 = (40,45,50,55,60)^T$, $\sigma = (0.3, 0.275, 0.25, 0.225, 0.2)^T$

$$\rho = \begin{pmatrix} 1 & -0.37 & -0.40 & -0.44 & -0.50 \\ -0.37 & 1 & -0.50 & -0.46 & -0.05 \\ -0.40 & -0.50 & 1 & 0.51 & 0.29 \\ -0.44 & -0.46 & 0.51 & 1 & 0.20 \\ -0.50 & -0.05 & 0.29 & 0.20 & 1 \end{pmatrix},$$

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

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

Table 4. Pricing a European put option on a basket of three assets: Estimates of the option price at time 0. Columns 1-3: the method used to approximate $P_0$; column 4-7: the discretization parameters: $M'$ the number of Monte Carlo samples, $N_Q$ the number of quadrature points, $N_S$ the number of vertices of the triangulation for the finite element method, $N_S$ the number of samples used to compute the mean (m) and the standard deviation (s); in column 10 the computing time.
to be enough the computational effort has been very low for QUAD5 and QUAD4-PDE1. In the case the method is stochastic, deterministic methods allowed to reduce the variance, such as in MC4-QUAD1 and MC4-PDE1-QUAD1.

8. Conclusion. Mixing Monte-Carlo methods with partial differential equation 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 non constant coefficient models with and without jump processes and also for American contracts, although proofs of convergence have not been given here.

For multidimensional 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 use PDE methods on one equation only.

Speed-up technique by polynomial fit has been discussed also but we plan to elaborate on such ideas in the future in particular 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


E-mail address: Olivier.Pironneau@upmc.fr