NUMERICAL METHODS FOR THE LÉVY LIBOR MODEL

ANTONIS PAPAPANTOLEON AND DAVID SKOVMAND

Abstract. The aim of this work is to provide fast and accurate approximation schemes for the Monte-Carlo pricing of derivatives in the Lévy LIBOR model of Eberlein and Özkan (2005). Standard methods can be applied to solve the stochastic differential equations of the successive LIBOR rates but the methods are generally slow. Our contribution is twofold. Firstly, we propose an alternative approximation scheme based on Picard iterations. This approach is similar in accuracy to the Euler discretization, but with the feature that each rate is evolved independently of the other rates in the term structure. This enables simultaneous calculation of derivative prices of different maturities using parallel computing. Secondly, the product terms occuring in the drift of a LIBOR Market model driven by a jump process grow exponentially as a function of the number of rates, quickly rendering the model intractable. We reduce this growth from exponential to quadratic in an approximation using truncated expansions of the product terms. We include numerical illustrations of the accuracy and speed of our method pricing caplets, swaptions and forward rate agreements.

1. Introduction

The LIBOR market model has become a standard model for the pricing of interest rate derivatives in recent years. The main advantage of the LIBOR model in comparison to other approaches, is that the evolution of discretely compounded, market-observable forward rates is modeled directly and not deduced from the evolution of unobservable factors. Moreover, the log-normal LIBOR model is consistent with the market practice of pricing caps according to Black's formula (cf. Black 1976). However, despite its apparent popularity, the LIBOR market model has certain well-known pitfalls.

On the one hand, the log-normal LIBOR model is driven by a Brownian motion, hence it cannot be calibrated adequately to the observed market data. An interest rate model is typically calibrated to the implied volatility surface from the cap market and the correlation structure of at-the-money swaptions. Several extensions of the LIBOR model have been proposed in the literature using jump-diffusions, Lévy processes or general semimartingales as the driving motion (cf. e.g. Glasserman and Kou 2003, Eberlein and Özkan 2005, Jamshidian 1999), or incorporating stochastic volatility effects (cf. e.g. Andersen and Brotherton-Ratcliffe 2005).

On the other hand, the dynamics of LIBOR rates are not tractable under every forward measure due to the random terms that enter the dynamics of LIBOR rates during the construction of the model. In particular, when the driving process has continuous paths the dynamics of LIBOR rates

Date: July 16, 2010.

Key words and phrases. LIBOR models, Lévy processes, Lévy LIBOR model, Picard approximation, drift expansion, parallel computing.

are tractable under their corresponding forward measure, but they are not tractable under any other forward measure. When the driving process is a general semimartingale, then the dynamics of LIBOR rates are not even tractable under their very own forward measure. Consequently: if the driving process is a *continuous* semimartingale caplets can be priced in closed form, but *not* swaptions or other multi-LIBOR derivatives. However, if the driving process is a *general* semimartingale, then even caplets *cannot* be priced in closed form. The standard remedy to this problem is the so-called "frozen drift" approximation, where one replaces the random terms in the dynamics of LIBOR rates by their deterministic initial values; it was first proposed by Brace et al. (1997) for the pricing of swaptions and has been used by several authors ever since. Brace et al. (2001) among others argue that freezing the drift is justified, since the deviation from the original equation is small in several measures.

Although the frozen drift approximation is the simplest and most popular solution, it is well-known that it does not yield acceptable results, especially for exotic derivatives and longer horizons. Therefore, several other approximations have been developed in the literature. We refer the reader to Joshi and Stacey (2008) for a detailed overview of that literature, and for some new approximation schemes and numerical experiments.

Although most of this literature focuses on the lognormal LIBOR market model, Glasserman and Merener (2003b, 2003a) have developed approximation schemes for the pricing of caps and swaptions in jump-diffusion LIBOR market models.

In this article we develop a general method for the approximation of the random terms that enter into the drift of LIBOR models. In particular, by applying Picard iterations we develop a generic approximation scheme. The method we develop yields more accurate results than the frozen drift approximation, while having the added feature that the individual rates can be evolved independently in a Monte Carlo simulation. This enables the use of parallel computing in the maturity dimension. Moreover, our method is universal and can be applied to any LIBOR model driven by a general semimartingale. We illustrate the accuracy and speed of our method in a case where LIBOR rates are driven by a normal inverse Gaussian process.

2. The Lévy LIBOR model

The Lévy LIBOR model was developed by Eberlein and Özkan (2005), following the seminal articles of Sandmann et al. (1995), Miltersen et al. (1997) and Brace et al. (1997) on LIBOR market models driven by Brownian motion; see also Glasserman and Kou (2003) and Jamshidian (1999) for LIBOR models driven by jump processes and general semimartingales respectively. The Lévy LIBOR model is a *market model* where the forward LIBOR rate is modeled directly, and is driven by a time-inhomogeneous Lévy process.

Let $0 = T_0 < T_1 < \cdots < T_N < T_{N+1} = T_*$ denote a discrete tenor structure where $\delta_i = T_{i+1} - T_i$, $i \in \{0, 1, \ldots, N\}$. Consider a complete stochastic basis $(\Omega, \mathcal{F}, \mathbf{F}, \mathbb{P}_{T_*})$ and a time-inhomogeneous Lévy process $H =$

$(H_t)_{0\le t\le T_*}$ satisfying standard assumptions such as the existence of exponential moments and absolutely continuous characteristics. The law of H is described by the Lévy–Khintchine formula:

$$\mathbb{E}_{\mathbb{P}_{T_*}}\left[\mathrm{e}^{iuH_t}\right] = \exp\left(\int_0^t \kappa_s(iu)\mathrm{d}s\right). \tag{2.1}$$

Here κ_s is the *cumulant generating function* associated to the infinitely divisible distribution with Lévy triplet $(0, c, F^{T_*})$, i.e. for $u \in \mathbb{R}$ and $s \in [0, T_*]$

$$\kappa_s(iu) = -\frac{c_s}{2}u^2 + \int_{\mathbb{R}} (\mathrm{e}^{iux} - 1 - iux)F_s^{T_*}(\mathrm{d}x). \tag{2.2}$$

The canonical decomposition of H is:

$$H = \int_0^{\cdot} \sqrt{c_s}\mathrm{d}W_s^{T_*} + \int_0^{\cdot}\int_{\mathbb{R}} x(\mu^H - \nu^{T_*})(\mathrm{d}s, \mathrm{d}x), \tag{2.3}$$

where W^{T_*} is a $\mathbb{P}_{T_*}$-standard Brownian motion, μ^H is the random measure associated with the jumps of H and ν^{T_*} is the $\mathbb{P}_{T_*}$-compensator of μ^H. We further assume that the following conditions are in force.

(LR1): For any maturity T_i there exists a bounded, continuous, deterministic function $\lambda(\cdot, T_i) : [0, T_i] \to \mathbb{R}$, which represents the volatility of the forward LIBOR rate process $L(\cdot, T_i)$. Moreover, we assume that (i) for all $s \in [0, T_*]$, there exist $M, \epsilon > 0$ such that $\int_0^{T_*}\int_{\{|x|>1\}} \mathrm{e}^{ux}F_t(\mathrm{d}x)\mathrm{d}t < \infty$, for $u \in [-(1+\varepsilon)M, (1+\varepsilon)M]$, and (ii) for all $s < T_i$

$$\sum_{i=1}^{N} \left|\lambda(s, T_i)\right| \le M.$$

(LR2): The initial term structure $B(0, T_i)$, $1 \le i \le N + 1$, is strictly positive and strictly decreasing. Consequently, the initial term structure of forward LIBOR rates is given, for $1 \le i \le N$, by

$$L(0, T_i) = \frac{1}{\delta_i}\left(\frac{B(0, T_i)}{B(0, T_i + \delta_i)} - 1\right) > 0. \tag{2.4}$$

The construction of the model starts by postulating that the dynamics of the forward LIBOR rate with the longest maturity $L(\cdot, T_N)$ is driven by the time-inhomogeneous Lévy process H and evolve as a martingale under the terminal forward measure $\mathbb{P}_{T_*}$. Then, the dynamics of the LIBOR rates for the preceding maturities are constructed by backward induction; they are driven by the same process H and evolve as martingales under their associated forward measures. For the full mathematical construction we refer to Eberlein and Özkan (2005).

We will now proceed to introduce the stochastic differential equation that the dynamics of log-LIBOR rates satisfy under the terminal measure $\mathbb{P}_{T_*}$. This will be the starting point for the approximation method that will be developed in the next section.

In the Lévy LIBOR model the dynamics of the LIBOR rate $L(\cdot, T_i)$ under the terminal forward measure $\mathbb{P}_{T_*}$ are given by

$$L(t, T_i) = L(0, T_i) \exp\left(\int_0^t b(s, T_i) \mathrm{d}s + \int_0^t \lambda(s, T_i) \mathrm{d}H_s \right), \tag{2.5}$$

where $H = (H_t)_{0 \le t \le T_*}$ is the $\mathbb{P}_{T_*}$-time-inhomogeneous Lévy process. The drift term $b(\cdot, T_i)$ is determined by no-arbitrage conditions and has the form

$$\begin{aligned} b(s, T_i) = -\frac{1}{2}\lambda^2(s, T_i)c_s - c_s\lambda(s, T_i) \sum_{l=i+1}^{N} \frac{\delta_l L(s-, T_l)}{1 + \delta_l L(s-, T_l)} \lambda(s, T_l) \\ - \int_{\mathbb{R}} \left(\left(\mathrm{e}^{\lambda(s,T_i)x} - 1 \right) \prod_{l=i+1}^{N} \beta(s, x, T_l) - \lambda(s, T_i)x \right) F_s^{T_*}(\mathrm{d}x), \end{aligned} \tag{2.6}$$

where

$$\beta(t, x, T_l,) = \frac{\delta_l L(t-, T_l)}{1 + \delta_l L(t-, T_l)} \left(\mathrm{e}^{\lambda(t,T_l)x} - 1 \right) + 1. \tag{2.7}$$

Note that the drift term in (2.5) is random, therefore we are dealing with a general semimartingale, and not with a Lévy process. Of course, $L(\cdot, T_i)$ is not a $\mathbb{P}_{T_*}$-martingale, unless $i = N$ (where we use the conventions $\sum_{l=1}^{0} = 0$ and $\prod_{l=1}^{0} = 1$).

Let us denote by Z the log-LIBOR rates, that is

$$\begin{aligned} Z(t, T_i) &:= \log L(t, T_i) \\ &= Z(0, T_i) + \int_0^t b(s, T_i) \mathrm{d}s + \int_0^t \lambda(s, T_i) \mathrm{d}H_s, \end{aligned} \tag{2.8}$$

where $Z(0, T_i) = \log L(0, T_i)$ for all $i \in \{1, \dots, N\}$.

Remark 2.1. Note that the martingale part of $Z(\cdot, T_i)$, i.e. the stochastic integral $\int_0^{\cdot} \lambda(s, T_i) \mathrm{d}H_s$, is a time-inhomogeneous Lévy process. However, the random drift term destroys the Lévy property of $Z(\cdot, T_i)$, as the increments are no longer independent.

3. Picard approximation for LIBOR models

The log-LIBOR can be alternatively described as a solution to the following linear SDE

$$\mathrm{d}Z(t, T_i) = b(t, T_i)\mathrm{d}t + \lambda(t, T_i)\mathrm{d}H_t, \tag{3.1}$$

with initial condition $Z(0, T_i) = \log L(0, T_i)$. Let us look further into the above SDE for the log-LIBOR rates. We introduce the term $Z(\cdot)$ in the drift term $b(\cdot, T_i; Z(\cdot))$ to make explicit that the log-LIBOR rates depend on all subsequent rates on the tenor.

The idea behind the Picard approximation scheme is to approximate the drift term in the dynamics of the LIBOR rates; this approximation is achieved by the Picard iterations for (3.1). The first Picard iteration for (3.1) is simply the initial value, i.e.

$$Z^{(0)}(t, T_i) = Z(0, T_i), \tag{3.2}$$

while the second Picard iteration is

$$\begin{aligned} Z^{(1)}(t, T_i) &= Z(0, T_i) + \int_0^t b(s, T_i; Z^{(0)}(s)) \mathrm{d}s + \int_0^t \lambda(s, T_i) \mathrm{d}H_s \\ &= Z(0, T_i) + \int_0^t b(s, T_i; Z(0)) \mathrm{d}s + \int_0^t \lambda(s, T_i) \mathrm{d}H_s. \end{aligned} \tag{3.3}$$

Since the drift term $b(\cdot, T_i; Z(0))$ is deterministic, as the random terms have been replaced with their initial values, we can easily deduce that the second Picard iterate $Z^{(1)}(\cdot, T_i)$ is a Lévy process.

Comparing (3.3) with (2.8) it becomes evident that we are approximating the semimartingale $Z(\cdot, T_i)$ with the time-inhomogeneous *Lévy process* $Z^{(1)}(\cdot, T_i)$.

3.1. **Application to LIBOR models.** In this section, we will apply the Picard approximation of the log-LIBOR rates $Z(\cdot, T_i)$ by $Z^{(1)}(\cdot, T_i)$ in order to derive a *strong*, i.e. pathwise, approximation for the dynamics of log-LIBOR rates. That is, we replace the random terms in the drift $b(\cdot, T_i; Z(\cdot))$ by the Lévy process $Z^{(1)}(\cdot, T_i)$ instead of the semimartingale $Z(\cdot, T_i)$. Therefore, the dynamics of the *approximate* log-LIBOR rates are given by

$$\widehat{Z}(t, T_i) = Z(0, T_i) + \int_0^t b(s, T_i; Z^{(1)}(s)) \mathrm{d}s + \int_0^t \lambda(s, T_i) \mathrm{d}H_s, \tag{3.4}$$

where the drift term is provided by

$$\begin{aligned} b(s, T_i; Z^{(1)}(s)) = &-\frac{1}{2}\lambda^2(s, T_i) c_s - c_s \lambda(s, T_i) \sum_{l=i+1}^{N} \frac{\delta_l \mathrm{e}^{Z^{(1)}(s-, T_l)}}{1 + \delta_l \mathrm{e}^{Z^{(1)}(s-, T_l)}} \lambda(s, T_l) \\ &- \int_{\mathbb{R}} \left(\left(\mathrm{e}^{\lambda(s, T_i) x} - 1 \right) \prod_{l=i+1}^{N} \widehat{\beta}(s, x, T_l) - \lambda(s, T_i) x \right) F_s^{T_*}(\mathrm{d}x), \end{aligned} \tag{3.5}$$

with

$$\widehat{\beta}(t, x, T_l,) = \frac{\delta_l \exp\left(Z^{(1)}(t-, T_l)\right)}{1 + \delta_l \exp\left(Z^{(1)}(t-, T_l)\right)} \left(\mathrm{e}^{\lambda(t, T_l) x} - 1 \right) + 1. \tag{3.6}$$

The main advantage of the Picard approximation is that the resulting SDE for $\widehat{Z}(\cdot, T_i)$ can be simulated more easily than the equation for $Z(\cdot, T_i)$. Indeed, looking at (3.1) and (2.6) again, we can observe that each LIBOR rate $L(\cdot, T_i)$ depends on all subsequent rates $L(\cdot, T_l)$, $i + 1 \leq l \leq N$. Hence, in order to simulate $L(\cdot, T_i)$, we should start by simulating the furthest rate in the tenor and proceed iteratively from the end. On the contrary, the dynamics of $\widehat{Z}(\cdot, T_i)$ depend only on the Lévy processes $Z^{(1)}(\cdot, T_l)$, $i + 1 \leq l \leq N$, which are independent of each other. Hence, we can use *parallel* *computing* to simulate all approximate LIBOR rates simultaneously. This significantly increases the speed of the Monte Carlo simulations which will be demonstrated in the numerical example.

3.2. **Drift expansion.** Let us look now at the drift term in (2.6) more carefully. Observe that there is a product of the form $\prod_{k=1}^{N}(1+a_k)$ appearing; the expansion of this product has the following form

$$\prod_{k=1}^{N}(1+a_k) = 1 + \sum_{k=1}^{N} a_k + \underbrace{\sum_{1\le i<j\le N} a_i a_j}_{\#\binom{N}{2}} + \underbrace{\sum_{1\le i<j<k\le N} a_i a_j a_k}_{\#\binom{N}{3}} + \underbrace{\sum_{i\neq j\neq k\neq l} a_i a_j a_k a_l}_{\#\binom{N}{4}} + \cdots + \prod_{k=1}^{N} a_k, \tag{3.7}$$

where the number of terms on the right hand side is 2^N. Therefore, we need to perform 2^N computations in order to calculate the drift of the LIBOR rates. Since N is the length of the tenor, it becomes apparent that this calculation is feasible for a short tenor, but not for long tenors; e.g. for $N = 40$ this amounts to more than 1 trillion computations.

In order to deal with this computational problem, we will approximate the LHS of (3.7) with the first or second order terms. Let us introduce the following shorthand notation for convenience:

$$\lambda_l := \lambda(s, T_l) \quad \text{and} \quad L_l := L(s, T_l). \tag{3.8}$$

We denote by $\mathbb{A}$ the part of the drift term that is stemming from the jumps, i.e.

$$\mathbb{A} = \int_{\mathbb{R}} \left(\left(\mathrm{e}^{\lambda_i x} - 1\right) \prod_{l=i+1}^{N} \left(1 + \frac{\delta_l L_l}{1+\delta_l L_l}\left(\mathrm{e}^{\lambda_l x} - 1\right)\right) - \lambda_i x \right) F_s^{T_*}(\mathrm{d}x). \tag{3.9}$$

The first order approximation of the product term is

$$\begin{aligned}
\mathbb{A}' &= \int_{\mathbb{R}} \left(\left(\mathrm{e}^{\lambda_i x} - 1\right) \left(1 + \sum_{l=i+1}^{N} \frac{\delta_l L_l}{1+\delta_l L_l}\left(\mathrm{e}^{\lambda_l x} - 1\right)\right) - \lambda_i x \right) F_s^{T_*}(\mathrm{d}x) \\
&= \int_{\mathbb{R}} \left(\mathrm{e}^{\lambda_i x} - 1 - \lambda_i x\right) F_s^{T_*}(\mathrm{d}x) \\
&\quad + \sum_{l=i+1}^{N} \frac{\delta_l L_l}{1+\delta_l L_l} \int_{\mathbb{R}} \left(\mathrm{e}^{\lambda_i x} - 1\right)\left(\mathrm{e}^{\lambda_l x} - 1\right) F_s^{T_*}(\mathrm{d}x) \\
&= \kappa(\lambda_i) + \sum_{l=i+1}^{N} \frac{\delta_l L_l}{1+\delta_l L_l} \Big(\kappa(\lambda_i + \lambda_l) - \kappa(\lambda_i) - \kappa(\lambda_l)\Big),
\end{aligned} \tag{3.10}$$

and the order of the error is

$$\mathbb{A} = \mathbb{A}' + O(\|L\|^2). \tag{3.11}$$

Similarly the second order approximation is provided by

$$\begin{aligned}\mathbb{A}'' = \kappa\big(\lambda_i\big) + \sum_{l=i+1}^{N} \frac{\delta_l L_l}{1+\delta_l L_l}\Big(\kappa\big(\lambda_i+\lambda_l\big) + \kappa\big(\lambda_i\big) + \kappa\big(\lambda_l\big)\Big) \\ + \sum_{i+1\le k<l\le N} \frac{\delta_l L_l}{1+\delta_l L_l}\frac{\delta_k L_k}{1+\delta_k L_k} \\ \times \Big(\kappa\big(\lambda_i+\lambda_l+\lambda_k\big) - \kappa\big(\lambda_i+\lambda_l\big) - \kappa\big(\lambda_i+\lambda_k\big) \\ - \kappa\big(\lambda_k+\lambda_l\big) + \kappa\big(\lambda_i\big) + \kappa\big(\lambda_l\big) + \kappa\big(\lambda_k\big)\Big),\end{aligned} \tag{3.12}$$

and the order of the error is

$$\mathbb{A} = \mathbb{A}'' + O(\|L\|^3). \tag{3.13}$$

3.3. **Caplets.** The price of a caplet with strike K maturing at time T_i, using the relationship between the terminal and the forward measures can be expressed as

$$\begin{aligned}\mathbb{C}_0(K,T_i) &= \delta_i B(0,T_{i+1})\, \mathbb{E}_{\mathbf{P}_{T_{i+1}}}[(L(T_i,T_i)-K)^+] \\ &= \delta B(0,T_{i+1})\, \mathbb{E}_{\mathbf{P}_{T_*}}\Big[\frac{\mathrm{d}\mathbf{P}_{T_{i+1}}}{\mathrm{d}\mathbf{P}_{T_*}}\big|_{\mathcal{F}_{T_i}}(L(T_i,T_i)-K)^+\Big] \\ &= \delta B(0,T_*)\, \mathbb{E}_{\mathbf{P}_{T_*}}\Big[\prod_{l=i+1}^{N}\big(1+\delta L(T_i,T_l)\big)(L(T_i,T_i)-K)^+\Big].\end{aligned} \tag{3.14}$$

This equation will provide the actual prices of caplets corresponding to simulating the full SDE for the LIBOR rates. In order to calculate the Picard approximation prices for a caplet we have to replace $L(\cdot,T_\cdot)$ in (3.18) with $\widehat{L}(\cdot,T_\cdot)$. Similarly, for the frozen drift approximation prices we must use $\widehat{L}^0(\cdot,T_\cdot)$ instead of $L(\cdot,T_\cdot)$.

3.4. **Swaptions.** Next, we will consider the pricing of swaptions. Recall that a payer (resp. receiver) swaption can be viewed as a put (resp. call) option on a coupon bond with exercise price 1; cf. section 16.2.3 and 16.3.2 in Musiela and Rutkowski (1997). Consider a payer swaption with strike rate K, where the underlying swap starts at time T_i and matures at T_m ($i < m \le N$). The time-T_i value is

$$\begin{aligned}\mathbb{S}_{T_i}(K,T_i,T_m) &= \left(1 - \sum_{k=i+1}^{m} c_k B(T_i,T_k)\right)^+ \\ &= \left(1 - \sum_{k=i+1}^{m}\left(c_k \prod_{l=i}^{k-1}\frac{1}{1+\delta L(T_i,T_l)}\right)\right)^+,\end{aligned} \tag{3.15}$$

where

$$c_k = \begin{cases} \delta_k K, & i+1 \leq k \leq m-1, \\ 1+\delta_k K, & k = m. \end{cases} \tag{3.16}$$

Then, the time-0 value of the swaption is obtained by taking the $\mathrm{I\!P}_{T_i}$-expectation of its time-T_i value, that is

$$\begin{aligned} \mathbb{S}_0 &= \mathbb{S}_0(K, T_i, T_m) \\ &= B(0,T_i)\,\mathrm{I\!E}_{\mathbf{P}_{T_i}}\left[\left(1 - \sum_{k=i+1}^{m}\left(c_k \prod_{l=i}^{k-1} \frac{1}{1+\delta L(T_i,T_l)}\right)\right)^+\right] \\ &= B(0,T_*) \\ &\quad \times \mathrm{I\!E}_{\mathbf{P}_{T_*}}\left[\prod_{l=i}^{N}(1+\delta L(T_i,T_l))\left(1 - \sum_{k=i+1}^{m}\left(c_k \prod_{l=i}^{k-1} \frac{1}{1+\delta L(T_i,T_l)}\right)\right)^+\right], \end{aligned}$$

hence

$$\mathbb{S}_0 = B(0,T_*)\,\mathrm{I\!E}_{\mathbf{P}_{T_*}}\left[\left(-\sum_{k=i}^{m}\left(c_k \prod_{l=k}^{N}(1+\delta L(T_i,T_l))\right)\right)^+\right], \tag{3.17}$$

where $c_i := -1$.

3.5. **Forward Rate Agreements.** A forward rate agreement(FRA) with strike K and notional value of 1 with expiry at time T_i, has the value of $\delta_i(K - L(T_i,T_i))$ at expiry. The time zero value of the the contract has the model independent value of $\delta_i B(0,T_{i+1})[K - L(0,T_i)]$ or zero if the contract is struck at-the-money ($K = L(0,T_i)$). In the following section we will compare the known true values with simulated prices generated from the terminal measure expectation of the payoff:

$$\mathbb{F}_0(K,T_i) = \delta B(0,T_*)\,\mathrm{I\!E}_{\mathbf{P}_{T_*}}\Big[\prod_{l=i+1}^{N}\big(1+\delta L(T_i,T_l)\big)(K - L(T_i,T_i))\Big]. \tag{3.18}$$

4. Numerical illustration

The aim of this section is to demonstrate the accuracy and efficiency of the Picard approximation scheme and the drift expansions in the valuation of options in the Lévy LIBOR model. In the first section we demonstrate the accuracy of our methods compared to a standard Euler discretization approximation in pricing caplets and swaptions. The second section we estimate the speed of our method and finally we compare with alternative approximations.

We will will examine a simple example with a flat volatility structure of $\lambda(\cdot,T_i) = 0.18$ and zero coupon rates generated from a flat term structure of interest rates $B(0,T) = \exp(-0.04T)$. We consider a tenor structure with 6 month increments (δ_i=0.5).

The driving Lévy process H is a normal inverse Gaussian (NIG) process with parameters $\alpha = \bar{\delta} = 12$ and $\mu = \beta = 0$ resulting in process with mean zero and variance 1. We denote by μ^H the random measure of jumps of H and by $\nu(\mathrm{d}t,\mathrm{d}x) = F(\mathrm{d}x)\mathrm{d}t$ the $\mathrm{I\!P}_{T_*}$-compensator of μ^H, where F is the

Lévy measure of the NIG process. The necessary conditions are then satisfied for term structures up to 30 years of length because $M = \alpha$, hence $\sum_{i=1}^{60} |\lambda(\cdot, T_i)| = 10.8 < \alpha$ and $\lambda(\cdot, T_i) < \frac{\alpha}{2}$, for all $i \in \{1, \dots, 9\}$.

The NIG Lévy process is a pure-jump Lévy process with canonical decomposition $H = \int_0^\cdot \int_{\mathbb{R}} x(\mu^H - \nu)(\mathrm{d}s, \mathrm{d}x)$. The cumulant generating function of the NIG distribution, for all $u \in \mathbb{C}$ with $|\Re u| \leq \alpha$, is

$$\kappa(u) = \bar{\delta}\alpha - \bar{\delta}\sqrt{\alpha^2 - u^2}. \tag{4.1}$$

4.1. **Accuracy of the Methods.** The Picard approximation should be considered primarily as a parallelizable alternative to the standard Euler discretization of the model. This will therefore be the benchmark to which we compare. In order to avoid Monte Carlo error we use the same discretization grid (5 steps per tenor increment) and the same pseudo random numbers (50000 paths) for each method. The pseudo random numbers are generated from the NIG distribution using the standard methodology described in Glasserman 2003.

Starting with caplets we can see in figure 4.1 the difference between the Euler discretization and the Picard approximation expressed in price (left) and implied volatility (right). The difference in price is small with max. errors around half of a percentage of a basis point. On the right we see somewhat larger errors with a maximum slightly below 1 basis point of implied volatility for low strike mid maturity caplets. Implied volatility is normally quoted in units of 1 basis point while bid-ask spreads are usually around at least 5 bp of implied volatility. The errors are therefore deemed to be at acceptable levels. Note also that in experiments not shown we found that the levels and patterns of the errors are insensitive to the number of discretization points as well the number of paths.

The errors display the non-monotonic behavior as a function of maturity with peaks around mid maturity. The non-monotonicity can be explained by the fact that volatility dominates the price of options in the short end, making the drift, and any error in it, less relevant. As maturity is increased the importance the drift grows relative to volatility but the state dependence becomes less critical as the number of "live" rates decreases. These two opposing effects result in the mid-maturity peak that we observe. This pattern is also noted in the study by Joshi and Stacey (2008).

As we established in (3.7) the number of terms needed to calculate the drift grows with a rate 2^N. In market applications N is often as high as 60 reflecting a 30 year term structure with a 6 month tenor increment. At this level even the calculation of one drift term becomes infeasible and this necessitates the use of the approximations introduced in (3.10) and (3.12). We investigate the errors introduced by the drift expansion by comparing them with the full numerical solution obtained as before using the true drift from (2.6). The results are plotted in figure 4.2. Here we can see that doing the first order drift expansion adds errors of a fairly low magnitude, however the second order drift expansion performs significantly better with errors so small they can be disregarded. We also plot the Picard approximation alone as well as combined with the second order drift expansion and these are similarly indistinguishable.

Continuing on with swaptions in figure 4.3 we plot again difference in implied volatility between our methods and the Euler discretization and we see smaller errors than for caplets, most likely because swaptions span a broad range maturities which smooth out the higher errors in the mid-maturity region.

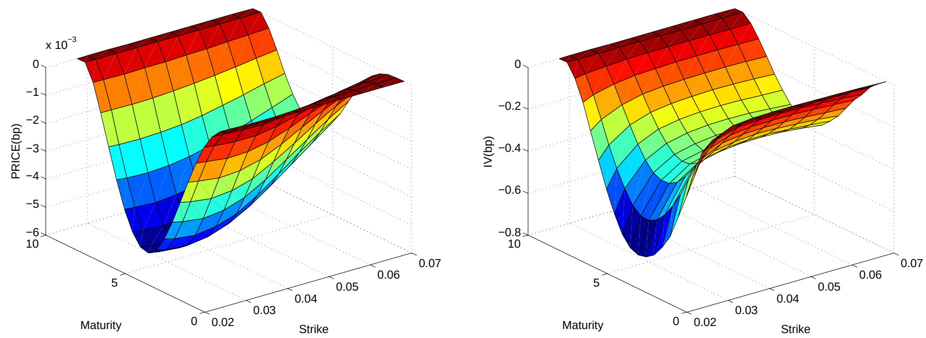

FIGURE 4.1. Difference in price (left) and implied caplet volatility in (right) between the Euler discretization and the the Picard approximation both in basis points.

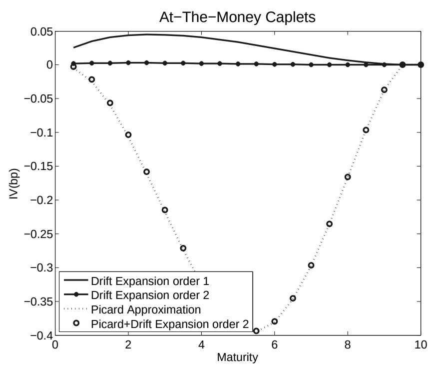

FIGURE 4.2. Difference in implied caplet volatility (in basis points) between the Euler discretization and 4 other methods

4.2. **Computational Speed.** In terms of computational time the largest gain by far is realized when using the approximations in (3.10) and (3.12). In the example above the CPU time for the full numerical solution is 1.5 Hours seconds but after applying the first order and second order drift approximation it drops to 1.3 and 27.2 seconds respectively. The Picard approximation in itself does not contribute to the computational speed unless parallelization is employed, in fact it is slightly but insignificantly slower as the auxiliary processes have to be evolved along with the rates. On the left in figure 4.4, CPU time as function of the number of paths for the Picard approximation and the full Euler discretization is plotted. In both cases the second

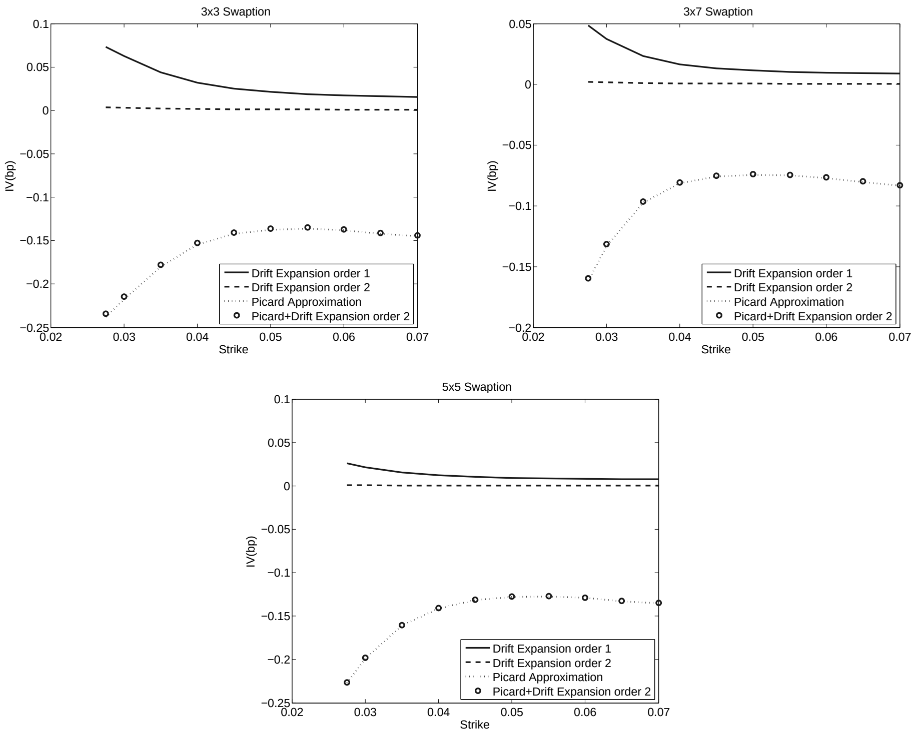

FIGURE 4.3. Difference in implied swaption volatility (in basis points) between Euler discretization and 4 other methods

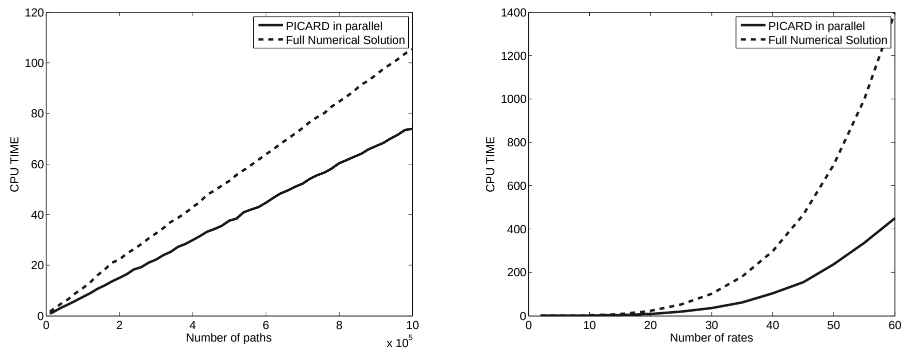

FIGURE 4.4. CPU time as function of the number of paths (left) and CPU time as a function of N, the number of rates (right).

order drift approximation scheme in (3.12) is employed. The computations are done in Matlab running on an Intel i7 processor with the capability of running 8 processes simultaneously. Here we see the typical linear behavior as the number of paths are increased but it can be seen that the Picard approximation has a significantly lower slope. Furthermore, on the right when we plot CPU time as a function of rates one can see CPU time exponentially increasing, revealing that large gains in computational time are realizable when using the Picard approximation scheme and the drift expansion. Needless to say, the speed advantages of the Picard approximation are only hinted at in the graphs since we are running on a simple desktop computer, and further speed increases can of course be realized as the access to more and more CPUs (or clusters of PCs) become more commonplace. Indeed, it already is a part of the infrastructure of many larger financial institutions.

4.3. **Comparison with other Methods.** Unfortunately, very little work has been done done in the area of approximation in a general semi-martingale driven LIBOR model leaving us without any standard method to compare with other than the frozen drift approximation. As mentioned in the introduction the existing work has mainly focused on the log-normal case and the case of finite intensity models. However, some of the simpler techniques applied to the log-normal case can be applied to the our setup as well. Suppose we want to simulate from time t to time $t+\Delta t$ where $t+\Delta t \leq T_i$. We have

$$L(t+\Delta t, T_i) = L(t, T_i) \exp\left(\int_t^{t+\Delta t} b(s, T_i)\mathrm{d}s + \int_t^{t+\Delta t} \lambda(s, T_i)\mathrm{d}H_s \right), \quad (4.2)$$

where $b(\cdot, T_i)$ is the state dependent drift function defined in (2.6). The standard (log)-Euler approximation assumes

$$\int_t^{t+\Delta t} b(s, T_i)\mathrm{d}s \approx \Delta t b(t, T_i) \quad (4.3)$$

But this can be refined as noted first in Kloeden and Platen 1992 (see also Hunter, Jäckel, and Joshi 2001, and Joshi and Stacey 2008 and the references therein) by instead using

$$\int_t^{t+\Delta t} b(s, T_i)\mathrm{d}s \approx \Delta t \left(\frac{1}{2} b(t, T_i) + \frac{1}{2} b(t+\Delta t, T_i) \right) \quad (4.4)$$

The second term in the parenthesis requires the knowledge of the LIBOR rates $L(t+\Delta t, T_{i+1}), \dots, L(t+\Delta t, T_N)$. One therefore has to simulate these rates. This can be done in a separate Euler step and the procedure is known in the LIBOR market model literature as *predictor-corrector* (PC). One can also note that when rate i is evolved under the terminal measure it only depends on rates $k > i$. Furthermore, if we start with $i = N$ we have no state-dependence in the drift. We can then generate realizations of $L(t+\Delta t, N)$ without discretization error and these can be used in the drift of rate $N-1$ as described above. Realizations of rate $N-1$ can then be generated with the corrected drift from (3.5), which can subsequently be used in the drift of rate $N-2$, and so forth. This latter method is referred to as *iterative predictor-corrector* (IPC).

IPC has been found to often outperform PC in the log-normal case studied in Joshi and Stacey 2008. It is also slightly more efficient since it does not require a separate simulation step for the simulation of the rates at time $t+\Delta t$.

One can also apply PC and IPC to the Picard approximation by merely replacing the drift in (4.4) with the one from (3.5). Furthermore, PC and IPC will actually be equivalent when applied to the Picard case since the drift does not involve the rates themselves, but the auxiliary processes denoted $Z^(1)(\cdot)$, making the order in which the rates are evolved irrelevant (See section XXX). We compare PC and IPC with our methods in 3 different cases. Since we do not have the true price of caplets or swaptions we instead compare prices of at-the-money forward rate agreements (FRA) since these all have a known model independent price of zero. Here we keep Monte-Carlo error sufficiently low by simulating 5 million paths, employing antithetic sampling as the only variance reducing measure. Looking at the top left of figure we have kept the discretization grid dense at 5 steps per tenor period. Here we see prices which for all methods are sufficiently small at max levels of 2.5 pct of a basis point. Furthermore the different methods are more or less indistinguishable – certainly in statistical terms with 95% confidence limit halfwidths ranging from about 0.004 bp in the short end to to 0.03 basis points in the long end. The frozen drift case is left out in this graph and the next, since the errors/prices are so big they dominate the other methods.
The second graph in the top right of figure shows prices for a discretization grid of 1 step per tenor period. Here the prices for Picard and Euler clearly reflect the higher discretization error and the PC methods and IPC are indistinguishable and significantly lower than Picard and Euler. In fact the prices are around the same level as in the 5 step/tenor period discretization. Finally, we price each FRA using a single long discretization step from time zero to the maturity of each contract also referred to as "long-stepping". In this case the Picard and Euler methods are analogous to the to the frozen drift case and PC and Picard+PC are equivalent to each other, hence we draw only PC, IPC, and frozen drift. We see, somewhat surprisingly that the frozen drift slightly outperforms PC and IPC which is in opposition the log-normal case previously studied in the literature. Nevertheless the errors are quite high and beyond an acceptable level for all methods.
The general conclusion one can draw from these graphs is that Picard+PC with 1 discretization step per tenor period would be the preferable choice. The errors are indistinguishable from regular PC and IPC, but the method has the advantage that prices can be parallelized in the maturity dimension and the speed increases shown in the previous section can be realized.

5. Conclusion

This paper derives new approximation methods for Monte Carlo derivative pricing in LIBOR models. The methods adress problem the slow Monte Carlo simulation by allowing parallel pricing through Picard approximations. In particular, the method decouples the interdependence of the rates when moving them forward in time in a simulation, meaning that the computations can be parallelized in the maturity dimension. Furthermore, the largest computational load in the model stems from the exponential growth of the drift terms. We reduce this growth from exponential to quadratic through truncated expansions of the product terms that appear in the drift.

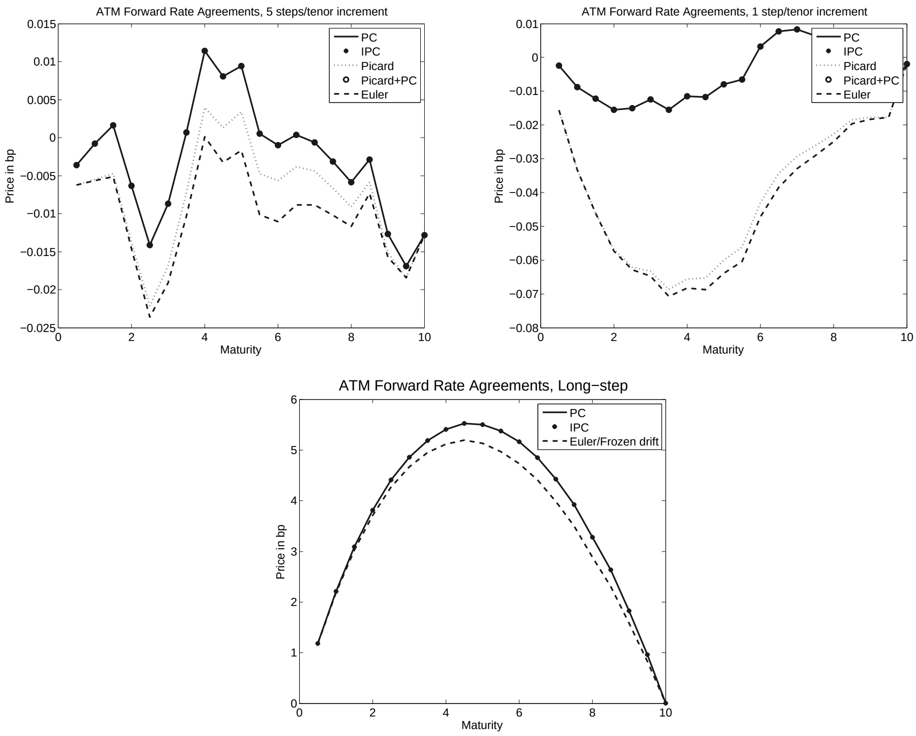

FIGURE 4.5. Price in Basis Points for at-the-money forward rate agreements

The accuracy is very high if a second order expansion is employed and we show that it reduces the computation load immensely.
Numerical methods for the general semi-martingale LIBOR model are still in their infancy. As we have demonstrated there is still work to be done in this area, in particular in the area of developing an algorithm for pricing of derivatives using long-step simulation. Predictor-corrector methods do not appear to work very well in this case, and a finer discretization grids need to be employed.

REFERENCES

Andersen, L. and R. Brotherton-Ratcliffe (2005). Extended LIBOR market models with stochastic volatility. *J. Comput. Finance 9*, 1–40.

Black, F. (1976). The pricing of commodity contracts. *J. Financ. Econ. 3*, 167–179.

Brace, A., T. Dun, and G. Barton (2001). Towards a central interest rate model. In E. Jouini, J. Cvitanić, and M. Musiela (Eds.), *Option pricing, interest rates and risk management*, pp. 278–313. Cambridge University Press.

Brace, A., D. Gątarek, and M. Musiela (1997). The market model of interest rate dynamics. *Math. Finance 7*, 127–155.

Eberlein, E. and F. Özkan (2005). The Lévy LIBOR model. *Finance Stoch. 9*, 327–348.

Glasserman, P. (2003). *Monte Carlo methods in financial engineering.* Springer-Verlag.

Glasserman, P. and S. G. Kou (2003). The term structure of simple forward rates with jump risk. *Math. Finance 13*, 383–410.

Glasserman, P. and N. Merener (2003a). Cap and swaption approximations in LIBOR market models with jumps. *J. Comput. Finance 7*, 1–36.

Glasserman, P. and N. Merener (2003b). Numerical solution of jump-diffusion LIBOR market models. *Finance Stoch. 7*, 1–27.

Hunter, C., P. Jäckel, and M. Joshi (2001). Getting the drift. *Risk 14*, 81–84.

Jamshidian, F. (1999). LIBOR market model with semimartingales. Working Paper, NetAnalytic Ltd.

Joshi, M. and A. Stacey (2008). New and robust drift approximations for the LIBOR market model. *Quant. Finance 8*, 427–434.

Kloeden, P. and E. Platen (1992). *Numerical solution of stochastic differential equations. 1992.* Springer-Verlag (Berlin).

Miltersen, K. R., K. Sandmann, and D. Sondermann (1997). Closed form solutions for term structure derivatives with log-normal interest rates. *J. Finance 52*, 409–430.

Musiela, M. and M. Rutkowski (1997). *Martingale Methods in Financial Modelling.* Springer.

Sandmann, K., D. Sondermann, and K. R. Miltersen (1995). Closed form term structure derivatives in a Heath–Jarrow–Morton model with log-normal annually compounded interest rates. In *Proceedings of the Seventh Annual European Futures Research Symposium Bonn*, pp. 145–165. Chicago Board of Trade.

Institute of Mathematics, TU Berlin, Strasse des 17. Juni 136, 10623 Berlin, Germany & Quantitative Products Laboratory, Deutsche Bank AG, Alexanderstr. 5, 10178 Berlin, Germany

E-mail address: papapan@math.tu-berlin.de

Aarhus School of Business, Aarhus University, Fuglesangs Allé 4, 8210 Aarhus V, Denmark

E-mail address: davids@asb.dk