

No. 6

**ACCELERATING THE CALIBRATION OF  
STOCHASTIC VOLATILITY MODELS**

**FIODAR KILIN**

MAY 2007

AUTHOR: *FIODAR KILIN*  
FRANKFURT SCHOOL OF  
FINANCE & MANAGEMENT  
FRANKFURT/MAIN  
FIODAR.KILIN@QUANTEAM.DE

PUBLISHER: FRANKFURT SCHOOL OF FINANCE & MANAGEMENT  
PHONE: +49 (0) 69 154 008-0 ■ FAX: +49 (0) 69 154 008-728  
SONNEMANNSTR. 9-11 ■ D-60314 FRANKFURT/M. ■ GERMANY

## Abstract

This paper compares the performance of three methods for pricing vanilla options in models with known characteristic function: (1) Direct integration, (2) Fast Fourier Transform (FFT), (3) Fractional FFT. The most important application of this comparison is the choice of the fastest method for the calibration of stochastic volatility models, e.g. Heston, Bates, Barndorff-Nielsen-Shephard models or Levy models with stochastic time. We show that using additional cache technique makes the calibration with the direct integration method at least seven times faster than the calibration with the fractional FFT method.

**JEL Classification:** G13

**Keywords:** Stochastic Volatility Models, Calibration, Numerical Integration, Fast Fourier Transform

**Acknowledgments.** The author thanks Uwe Wystup and Bernd Engelmann for useful comments.

# 1 Introduction

When implementing a calibration algorithm for an option pricing model with known characteristic function of the asset's return, one has to choose a method for pricing vanilla options. In this paper we compare the following methods: (1) Direct integration, (2) Fast Fourier Transform (FFT), (3) Fractional FFT. Before choosing one of these techniques, it is important to consider all possible ways of improving accuracy and calculation speed of each of these methods. These improvements can include mathematical modifications as well as implementation techniques. In this paper we compare optimized implementations of the calibration algorithm based on each of the above mentioned valuation methods. It helps to identify the factors which are most important for accuracy and speed of calibration.

The literature provides some critique on an unoptimized version of the direct integration method. Carr and Madan (1999) criticize the inability of the direct integration method to harness the computational power of FFT. Lee (2004) and Carr and Madan (1999) point out the numerical instability of the direct integration method in case of using a decomposition of an option price into probability elements. However the modification of the direct integration method described in Attari (2004) is free from this instability. In the present paper we compare systematically all advantages and disadvantages of the three valuation methods. A special attention is paid to the possibility of a simultaneous valuation of a set of options and to the efficiency of the applied numerical integration methods.

The outline of this paper is as follows. In Section 2, we review some popular stochastic volatility models with known characteristic function. Section 3 describes the pricing methods which are compared in this paper. In Section 4, we describe the caching technique that accelerates the calibration with the direct integration method. Section 5 elaborates the details of a numerical experiment that compares the speed of calibration for the compared methods. Section 6 concludes.

## 2 The Models

The methods discussed in this paper can be applied to calibrate a bundle of models. An overview of the most promising stochastic volatility models can be found in Schoutens et al.

(2004) and in Cont and Tankov (2004). In this section we briefly describe examples of such models. The characteristic functions of these models are listed in the appendix.

**Heston model.** The risk-neutral dynamics in the model of Heston (1993) is

$$\frac{dS_t}{S_t} = rdt + \sigma_t dW_t, S_0 \geq 0, \quad (1)$$

where

$$d\sigma_t^2 = \kappa(\eta - \sigma_t^2)dt + \theta\sigma_t d\tilde{W}_t, \sigma_0 \geq 0, \quad (2)$$

$$\text{Cov}[dW_t, d\tilde{W}_t] = \rho dt, \quad (3)$$

$\kappa$  is the mean-reversion speed,  $\eta$  the long-run variance,  $\theta$  the volatility of variance,  $\rho$  the correlation between the underlying and volatility, and  $\sigma_0$  is the initial value of the volatility.

**Bates model.** This extension of the Heston model introduces jumps in the underlying process.

$$\frac{dS_t}{S_t} = (r - \lambda\mu_J)dt + \sigma_t dW_t + J_t dN_t, S_0 \geq 0, \quad (4)$$

where  $N_t$  is a Poisson process with intensity  $\lambda > 0$ . The process  $N_t$  is independent of  $W_t$  and  $\tilde{W}_t$ .  $J_t$  denotes the percentage jump size. It is lognormally, identically and independently distributed over time

$$\log(1 + J_t) \sim N(\log(1 + \mu_J) - \frac{\sigma_J^2}{2}, \sigma_J^2). \quad (5)$$

The volatility process  $\sigma_t$  follows the SDE (2). Condition (3) is fulfilled.

**Barndorff-Nielsen-Shephard model.** This model introduces simultaneous up-jumps in the volatility and down-jumps in the underlying price. The risk-neutral dynamics of the log-spot is

$$d(\log S_t) = (r - \lambda k(-\rho) - \sigma_t^2/2)dt + \sigma_t dW_t + \rho dz_{\lambda t}. \quad (6)$$

with the squared volatility following the process

$$d\sigma_t^2 = -\lambda\sigma_t^2 dt + dz_{\lambda t}, \quad (7)$$

where

$$z_{\lambda t} = \sum_{n=1}^{N_t} x_n. \quad (8)$$

$N_t$  is a Poisson process with intensity  $a$ ,  $x_n$  is an iid sequence, each  $x_n$  follows an exponential law with mean  $1/b$ . The cumulant function of  $z_1$  is

$$k(u) = \ln E(\exp(-uz_1)) = -au(b+u)^{-1}. \quad (9)$$

**Levy models with stochastic time.** This class of models introduces stochastic volatility effects by making the time stochastic. The risk-neutral underlying process is modeled as

$$S_t = S_0 \frac{\exp(rt)}{E[\exp(X_{Y_t})|y_0]} \exp(X_{Y_t}), \quad (10)$$

where  $X_t$  is a Levy process,  $Y_t$  is a business time

$$Y_t = \int_0^t y_s ds, \quad (11)$$

and  $y_t$  is the rate of time change. A possible choice for  $X_t$  is the Variance Gamma (VG) process or the Normal Inverse Gaussian process (NIG). Typical examples of the rate of time change  $y_t$  are the CIR stochastic clock

$$dy_t = \kappa(\eta - y_t)dt + \lambda y_t^{1/2} dW_t, \quad (12)$$

or the Gamma-Ornstein-Uhlenbeck process

$$dy_t = -\lambda y_t dt + dz_{\lambda t}, \quad (13)$$

where  $z_t$  is defined as in (8).

### 3 Pricing methods

In this section we describe the pricing methods and point out their limitations.

**Direct integration.** The direct integration method implies computing vanilla call option values using one-dimensional numerical quadrature, for example Gaussian quadrature.<sup>1</sup> For this method Attari (2004) obtains an efficient formula<sup>2</sup>

---

<sup>1</sup>The first analytical formula for pricing vanilla options in stochastic volatility models with known characteristic function was obtained in Heston (1993). Bakshi and Madan (2000) extend this approach and point out its theoretical advantages.

<sup>2</sup>An equivalent formula has been obtained by Lewis (2001). This formula can also be used with the direct integration method.

$$C(S_0, T, K) = S_0 - \frac{1}{2}e^{-rT}K \tag{14}$$

$$-e^{-rT}K \left( \frac{1}{\pi} \int_0^{+\infty} \frac{(\operatorname{Re}(\phi(\omega)) + \frac{\operatorname{Im}(\phi(\omega))}{\omega}) \cos(\omega l(K)) + (\operatorname{Im}(\phi(\omega)) - \frac{\operatorname{Re}(\phi(\omega))}{\omega}) \sin(\omega l(K))}{1 + \omega^2} d\omega \right)$$

where  $S_t$  is the underlying price,  $K$  is the strike price,  $T$  is the maturity of the option,  $r$  is the risk-free interest rate, the dividend yield is assumed to be zero,  $\mathbb{Q}$  is the risk-neutral measure and

$$\phi(\omega) = E_{\mathbb{Q}}(e^{i\omega x})$$

is the characteristic function of

$$x = \ln \left( \frac{S_T}{S_0} \right) - rT.$$

The advantages of this formula in comparison with the classical formula of Heston (1993) are:

- 1.) Formula (14) contains only one integral instead of two.
- 2.) The integrand in (14) has a quadratic term in the denominator. This gives a faster rate of decay.

The implementation of this method should control the branches of the complex logarithm that appears in the characteristic function. It slightly complicates the implementation of this method, but does not affect the accuracy and the speed of the calculations. One possible solution of this problem is a reimplementaion of the complex logarithm routine with storing the returned value and the branch number at the previous step of the algorithm. An alternative solution is described in Kahl and Jäckel (2005). Lord and Kahl (2006,2) and Albrecher et al. (2007) show that for some models this problem can be solved by using an appropriate representation of the characteristic function.

**Fast Fourier Transform.** Carr and Madan (1999) suggest a transformation of the vanilla pricing formula that allows to use the FFT technique. The value of a call option can be expressed as

$$C(S_0, T, k) = \frac{e^{-\gamma k}}{\pi} \int_0^{+\infty} e^{-iku} \psi(u) du \quad (15)$$

where  $k$  denotes the log of the strike price,  $\gamma$  is a dampening parameter and

$$\psi(u) = \frac{e^{-rT} \widehat{\phi}(u - (\gamma + 1)i)}{\gamma^2 + \gamma - u^2 + (2\gamma + 1)ui}, \quad (16)$$

where

$$\widehat{\phi}(\omega) = E_Q(e^{i\omega \widehat{x}}) \quad (17)$$

is the characteristic function of the log price  $\widehat{x} = \ln(S_T)$ . The integral in (15) is approximated using an integration rule

$$\int_0^{+\infty} e^{-iku} \psi(u) du \approx \sum_{j=0}^{N_{FFT}-1} e^{-iku_j} \psi(u_j) w_j \delta, \quad (18)$$

$$u_j = j\delta, \quad (19)$$

where  $N_{FFT}$  is the number of grid points and the weights  $w_j$  implement the integration rule. The crucial limitation of the FFT method is that the grid points  $u_j$  must be chosen equidistantly. This limitation prohibits the use of the most effective integration rules such as the Gaussian quadrature.

The FFT pricing method simultaneously computes the values of the integral approximations (18) for the set of log-strikes  $\{k_m = -(\frac{N_{FFT}\lambda}{2}) + m\lambda, m = 0, \dots, N_{FFT} - 1\}$ . The simultaneous calculation for all strikes is not an exclusive advantage of the FFT-based methods, because a slightly modified direct integration method also has this advantage. This simple modification is described in the next section.

The second important restriction is that the grid spacings must satisfy the condition

$$\lambda\delta = \frac{2\pi}{N_{FFT}}. \quad (20)$$

If this condition is satisfied, the sums in (18) can be expressed in the form

$$\sum_{j=0}^{N_{FFT}-1} e^{-ik u_j} \psi(u_j) w_j \delta = \sum_{j=0}^{N_{FFT}-1} e^{-i\lambda \delta j m} h_j = \sum_{j=0}^{N_{FFT}-1} e^{-i(\frac{2\pi}{N_{FFT}}) j m} h_j, \quad (21)$$

which allows the application of the FFT procedure invoked on the vector  $h = \{h_j = e^{i(\frac{N\lambda}{2} j \delta)} \psi(u_j) w_j \delta, j = 0, \dots, N_{FFT} - 1\}$ .

**Fractional Fast Fourier Transform.** Chourdakis (2005) has shown how the method of Carr and Madan (1999) can be accelerated using the fractional FFT algorithm. This algorithm rapidly computes sums of the form

$$D_k(h, \alpha) = \sum_{j=0}^{N-1} e^{-i2\pi k j \alpha} h_j \quad (22)$$

for any value of  $\alpha$ . The fractional FFT method can be applied without the need to impose the restriction (20). However, the fractional FFT method does not overcome the crucial limitation of the FFT method because the grid points  $u_i$  still must be chosen equidistantly. Fractional FFT is implemented by invoking three FFT procedures, i.e.,

$$D_k(h, \alpha) = (e^{-i\pi k^2 \alpha})_{k=0}^{N-1} \odot D_k^{-1}(D_j(y) \odot D_j(z)), \quad (23)$$

where

$$y = ((h_j e^{-i\pi j^2 \alpha})_{j=0}^{N-1}, (0)_{j=0}^{N-1}), \quad (24)$$

$$z = ((e^{i\pi j^2 \alpha})_{j=0}^{N-1}, (e^{i\pi(N-j)^2 \alpha})_{j=0}^{N-1}), \quad (25)$$

$D_k(h)$  denotes the FFT sum

$$D_k(h) = \sum_{j=0}^{N-1} e^{-i\frac{2\pi}{N} k j} h_j, \quad (26)$$

$D_k^{-1}(h)$  is the inverse FFT sum

$$D_k^{-1}(H) = \frac{1}{N} \sum_{j=0}^{N-1} e^{i\frac{2\pi}{N} k j} H_j, \quad (27)$$

and  $\odot$  denotes element-by-element vector multiplication.

The fractional FFT pricing method is faster than the FFT pricing method, because the absence of the restriction (20) allows the use of sparser grids. This effect is more important in terms of computing time than the disadvantage of using three FFT routines instead of one.<sup>3</sup>

The accuracy of the prices calculated with the FFT or the fractional FFT methods strongly depends on the choice of the dampening parameter  $\gamma$ . Using the same value of the dampening parameter in all pricing situations would be fatal for the calibration procedure. In particular, if we use a reasonable grid size  $N_{FFT} < 4096$  there is no value of  $\gamma$  that leads to an acceptable pricing error for all possible parameter values. Lord and Kahl (2006,1) provide an example of two pricing inputs with non-overlapping sets of acceptable dampening parameters. Only an extremely fine FFT grid will result in overlapping sets of acceptable dampening parameters. But fine FFT grids are impractical because they slow down the calibration. Therefore the recommendations of Lee (2004) and Lord and Kahl (2006,1) for the choice of  $\gamma$  are not just an additional improvement of the FFT-based methods but a necessary requirement for the implementation of these methods.

## 4 Caching technique

The most time-consuming part of the computation is the evaluation of the characteristic function. For example the characteristic function of the Heston model<sup>4</sup>

---

<sup>3</sup>See Chourdakis (2005)

<sup>4</sup>The literature provides two specifications for the characteristic function of the Heston model. The first one is used in Heston (1993). The second one can be found in Schoutens et al. (2004) or in Gatheral (2005). We use the second specification. For justification of this choice see Albrecher et al. (2007)

$$\begin{aligned} \phi(\omega) = & \exp\left\{\eta\kappa\theta^{-2}((\kappa - \rho\theta\omega i - d)T - 2\ln\left(\frac{1 - ge^{-dT}}{1 - g}\right))\right. \\ & \left. + \sigma_0^2\theta^{-2}(\kappa - \rho\theta\omega i - d)\frac{1 - e^{-dT}}{1 - ge^{-dT}}\right\}, \end{aligned} \quad (28)$$

$$d = ((\rho\theta\omega i - \kappa)^2 - \theta^2(-i\omega - \omega^2))^{1/2}, \quad (29)$$

$$g = \frac{\kappa - \rho\theta\omega i - d}{\kappa - \rho\theta\omega i + d}, \quad (30)$$

contains two complex exponents,<sup>5</sup> one complex logarithm and one complex square root. Therefore an extremely important requirement for an effective implementation of the calibration algorithm is the following: The number of evaluations of the characteristic function should be as low as possible. If the calibration algorithm uses the direct integration method to compute the values of vanilla options, a caching technique should be used to avoid unnecessary recalculations of the characteristic function.

If the caching technique is not used, the calculation of the values of vanilla options at each iteration of the optimization algorithm includes the following steps:

1. Loop over expiries of the vanilla options.
2. Loop over strikes of the vanilla options.
3. Loop over the points  $\omega_i, i = 1, \dots, U$  that are used to evaluate the integral in (14) numerically.
4. Evaluate the characteristic function in  $\omega_i$ .
5. Evaluate the integrand in  $\omega_i$ .
6. Calculate the value of the vanilla option.

However, the value of the characteristic function does not depend on the strike. If we use the same grid  $\omega_i, i = 1, \dots, U$  for all options and run the described algorithm, we recalculate the same values of the characteristic function at each step of the strike-loop. We can use the following modification of the algorithm in order to avoid these recalculations:

1. Loop over expiries of the vanilla options.
2. Loop over strikes of the vanilla options.
3. Loop over the points  $\omega_i, i = 1, \dots, U$  that are used to evaluate the integral in (14) nu-

---

<sup>5</sup>We do not count identical repeated terms.

merically.

4. If we are at the first step of the strike-loop, evaluate the characteristic function in  $\omega_i$  and save this value in the cache.
5. If we are not at the first step of the strike-loop, read the value of the characteristic function in  $\omega_i$  from the cache.
6. Evaluate the integrand in  $\omega_i$ .
7. Calculate the price of the vanilla option.

The numerical evaluation of the integral in (14) requires a choice of the numerical upper integration limit. Suppose a maximum tolerable truncation error is given. Then the numerical upper integration limit  $\bar{\omega}$  depends on the maturity  $T$  and the strike  $K$  of the vanilla option:  $\bar{\omega} = \bar{\omega}(T, K)$ . We can still use the same  $\omega$ -grid for all  $T$  and  $K$  - we just define the index of the last integration point as an integer-valued function  $U(T, K)$  that satisfies the condition

$$\omega_{U(T,K)} \leq \bar{\omega}(T, K) < \omega_{U(T,K)+1}. \quad (31)$$

The grid at step 3 of the algorithm can now be defined as  $\omega_i, i = 1, \dots, U(T, K)$ . In most cases the function  $U(T, K)$  is an increasing function of  $K$ . It leads to a different number of loop iteration at step 3 for different  $K$ . Therefore, we have to modify the described algorithm once more in order to take this fact into account. We can use a reverse order of strikes or we can control at each point  $\omega_i$  whether the characteristic function has been already evaluated at this point. We can also combine these two solutions. In this case the algorithm is:

1. Loop over expiries of the vanilla options.
2. Loop over strikes of the vanilla options. Use a reverse order of strikes.
3. Loop over the points  $\omega_i, i = 1, \dots, U(T, K)$  that are used to evaluate the integral in (14) numerically.
4. If the value of the characteristic function in  $\omega_i$  is still not in the cache, evaluate it and save this value in the cache.
5. If the value of the characteristic function in  $\omega_i$  is already in the cache, use this precomputed value.
6. Evaluate the integrand in  $\omega_i$ .

7. Calculate the value of the vanilla option.

There is a further possibility to accelerate this algorithm. Some terms of the characteristic function do not depend on  $T$ . These terms can be precomputed before starting the loop over expiries of the vanilla options. For example, we recommend to compute the term (29) only once and store it, because it contains a time-consuming square root operator.

## 5 Numerical experiment

The FFT-based methods of pricing vanilla options are very popular because they simultaneously give option values for a range of strikes. This simultaneous calculation saves computing time because the characteristic function need not to be recomputed for different strikes. However, the direct integration method also has this useful feature - we just have to use the caching technique described in the previous section. Therefore the possibility of simultaneous pricing for different strikes cannot be considered as a criterium for comparison of pricing methods.<sup>6</sup> We have to define other criteria for the comparison.

The first of these criteria is the speed of the numerical integration method. Obviously, there are a lot of techniques of simple numerical integration in the general case that are both faster and more accurate than integration using FFT. They are designed to minimize the number of integrand evaluations. One of these techniques is the Gaussian quadrature formula. This section shows that the grid for the numerical integration (14) with six-point Gaussian quadrature is at least seven times more economical than the FFT-grid in (15).

The second criterion is the rate of decay of the integrand. The integrand in (14) decays at a quadratic rate. This is the main reason why we use the pricing formula from Attari (2004) rather than the formula from Heston (1993). The rate of decay of the integrand in (15) is also quadratic. Therefore this criterion does not indicate any advantages of the formula (15) relative to the formula (14).

---

<sup>6</sup>The FFT algorithm reduces the number of multiplications in the required  $N_{FFT}$  summations from an order of  $N_{FFT}^2$  to that of  $N_{FFT} \ln_2 N_{FFT}$  (Carr and Madan (1999)). However the computing time required for these multiplications is negligible in comparison with the time required for the evaluations of the characteristic function. Therefore we concentrate on the number of the calculations of the characteristic function only.

Accuracy (implied volatility basis points)	Grid size for the FFT method	Grid size for the fractional FFT method	Numerical integration grid size for the direct integration method
2.0	4096	1024	96
1.0	4096	2048	126
0.2	8192	2048	162
0.02	16384	4096	582

Table 1: Grid sizes that are needed to obtain some benchmark accuracy levels

As we have already pointed out, the number of the evaluations of the characteristic function is the main factor driving the calibration time. We have carried out a numerical experiment to compare the influence of this factor in each pricing method. Then we have conducted a second numerical experiment where we have compared the calibration time directly.

First, we notice that the number of evaluations of the characteristic function during the calibration procedure is equal to  $N_{grid} \cdot N_M$ , where  $N_M$  denotes the number of maturities in the set of vanilla options which are used to calibrate a model,  $N_{grid}$  is equal to  $N_{FFT}$  for the FFT-based pricing methods,<sup>7</sup> and

$$N_{grid} = \frac{\sum_{t=1}^{N_M} \max_j U(T_t, K_j)}{N_M} \quad (32)$$

for the direct integration method. Note that  $N_{grid}$  in (32) is approximately equal to the average size of the numerical integration grid at the last strike of each calibration maturity. In order to compare the performance of the above-mentioned pricing methods, we define some benchmark accuracy levels. We then estimate grid sizes  $N_{grid}$  that lead to the desired accuracy. The results are summarized in Table 1.<sup>8</sup> These results are based on the following numerical experiment. Values of 100 vanilla options (10 maturities from 0.1 to 5.0 years,

<sup>7</sup>Application of the fractional FFT method results in a smaller grid size and in fewer evaluations of the characteristic function.

<sup>8</sup>If the direct integration method is used,  $N_{grid}$  is not necessary an integer value (see (32)). However we report only an integer part of  $N_{grid}$  for a more natural interpretation.

10 strikes for each maturity) have been calculated with 100 random (but reasonable<sup>9</sup>) sets of parameters of the Heston model. These calculations have been performed with different grid sizes. The results of the calculations with extremely fine grids have been used as the benchmark for accuracy estimations.<sup>10</sup> Suppose we have obtained a value  $p_1$  with a reasonable grid size and a price  $p_2$  with extremely fine grid size. To estimate the accuracy of the price  $p_1$  we have computed Black-Scholes implied volatilities  $\sigma(p_1)$  and  $\sigma(p_2)$  for both prices  $p_1$  and  $p_2$ . The accuracy has been defined as the absolute difference  $|\sigma(p_1) - \sigma(p_2)|$  between these implied volatilities. Table 1 reports the minimum grid sizes that have led to the desired accuracy in all 10000 pricing situations.<sup>11</sup> The same experiment has been carried out for the Bates model, the Barndorff-Nielsen-Shephard model and four models based on time-changed Levy processes (NIG-CIR, NIG-GOU, VG-CIR, VG-GOU).<sup>12</sup> The results are very similar to the results reported in the Table 1. For all these models the grid for the fractional FFT method must be at least seven times finer than the grid for the direct integration method to obtain the same accuracy in both methods.

Our second numerical experiment compares the speed of the calibration directly. We have selected 100 random business days from January, 2000 to November, 2006. For each of these days we have used historical market data on DAX vanilla option prices as an input to the calibration routine. Each calibration input contains 80 to 155 options with 8 to 12 different maturities. We have run the calibration procedure with the three different vanilla pricing methods applied to seven different models. Each calibration consists of three runs of the Differential Evolution algorithm and three runs of the Levenberg-Marquardt algorithm.<sup>13</sup>

---

<sup>9</sup>The parameters have been drawn from the following ranges: long-run variance  $\eta \in [0.01, 1.0]$ , mean-reversion rate  $\kappa \in [1.0, 4.0]$ , volatility of variance  $\theta \in [0.01, 10.0]$ , short-term volatility  $\sigma_0 \in [0.1, 1.0]$ , correlation  $\rho \in [-1.0, 1.0]$ .

<sup>10</sup>It has been also checked that these benchmark values are identical for all three pricing methods. We have also tested the benchmark values against Monte-Carlo simulations. Occasionally there were some reference prices in the literature. In these cases we have also compared our prices with these references (Table 3 in Lord, Koekkoek and van Dijk (2006), Figure 6 in Kahl and Jäckel (2005)).

<sup>11</sup>10000 pricing situations correspond to 100 options and 100 parameter sets.

<sup>12</sup>The description of all these models can be found in Schoutens (2004).

<sup>13</sup>The Differential Evolution algorithm has been used to find a good initial guess for the Levenberg-Marquardt optimization routine. The Differential Evolution algorithm is described in Price et. al (2005). An open source code is available from <http://www.icsi.berkeley.edu/~storn/code.html>. The Levenberg-Marquardt algorithm is described in Gill and Murray (1978). An open source code is available from <http://quantlib.org>.

Model	FFT	Fractional FFT	Direct integration
Heston	466	239	15
Bates	620	316	20
Barndorff-Nielsen-Shephard	405	208	13
VG-CIR	540	281	17
VG-GOU	522	269	17
NIG-CIR	546	280	18
NIG-GOU	521	273	17

Table 2: Average calibration time (in seconds).

The grid sizes has been set as in the second line of Table 1 (accuracy = 1.0 basis points). The average calibration time<sup>14</sup> is compared in Table 2.

Table 2 shows that the calibration with the direct integration method is approximately 16 times faster than the calibration with the fractional FFT method. It corresponds to the ratio of grid sizes: 126 points for the direct integration and 2048 for the fractional FFT. However, we should take into account that we are extremely inflexible in the choice of the FFT grid - the number of the FFT grid points must be a power of two. Therefore the difference in calibration speed between the fractional FFT method and the direct integration method highly depends on the desired accuracy. For example, if the desired accuracy is 0.02 basis points, the direct integration grid size is approximately 7 times smaller than the fractional FFT grid (the last line of Table 1). It results in a corresponding ratio of calibration times for these methods.

## 6 Conclusion

We have shown that an efficient implementation of the direct integration method results in a sizable speed up of the calibration of stochastic volatility models. This method even

---

<sup>14</sup>The computations were done on a Centrino Pentium M, 1.5GHz CPU

outperforms the calibration with the fractional FFT. The simultaneous pricing of options with different strikes is not an exclusive advantage of the FFT methods compared to the direct integration method, because an application of a cache technique leads to simultaneous pricing of options with different strikes in the framework of direct integration. Taking this into account we argued that the pricing methods differ in two aspects only: the numerical integration technique and the pricing formula. The combination of these factors results in higher calculation speed of the direct integration method in comparison to the FFT and fractional FFT methods. Specifically: (1) Gaussian quadrature is a much faster numerical integration technique than the FFT, (2) The transformed pricing formula of Attari (2004) provides approximately the same rate of decay of the integrand in comparison with the main formula of the FFT method. As we have pointed out in the introduction, the direct integration method is frequently criticized in the literature. However this critique is valid only if we consider an unoptimized implementation of the general formula. The use of the modified pricing formula and the caching technique makes the direct integration method the best choice for practical applications.

## References

- [1] Albrecher, H., Mayer, P., Schoutens, W., Tistaert, J. (2007). The little Heston trap. *Wilmott Magazine*, to appear.
- [2] Attari, M. (2004). Option pricing using Fourier transforms. *Charles River Associates working paper*, available at <http://ssrn.com/abstract=520042>.
- [3] Bakshi, G., and Madan, D. (2000). Spanning and derivative-security valuation. *Journal of Financial Economics* 55, 205-38.
- [4] Carr, P., and Madan, D. (1999). Option valuation using the fast Fourier transform. *Journal of Computational Finance* 3, 463-520.
- [5] Chourdakis, K. (2005). Option pricing using the fractional FFT. *Journal of Computational Finance* 8(2), 1-18.
- [6] Cont, R., and Tankov, P. (2003). Financial modelling with jump processes. *Chapman & Hall/CRC*.

- [7] Gatheral, J. (2005). The volatility surface: A practioner's guide. *Wiley Finance*.
- [8] Gill, P., and Murray, W. (1978). Algorithms for the solution of the nonlinear least-squares problem. *SIAM Journal on Numerical Analysis* 15(5), 977-992.
- [9] Heston, S.L. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies* 6, 327-44.
- [10] Kahl, C., and Jäckel, P. (2005). Not-so-complex logarithms in the Heston model. *Wilmott Magazine* September, 94-103.
- [11] Lee, R. (2004). Option pricing by transform methods: extensions, unification and error control. *Journal of Computational Finance* 7(3), 51-86.
- [12] Lewis, A. (2001). A simple option formula for general jump-diffusion and other exponential Levy processes. *available from <http://www.optioncity.net>*.
- [13] Lord, R., Koekkoek, R., van Dijk, D. (2005). A comparison of biased simulation schemes for stochastic volatility models. *Tinbergen Institute Discussion Paper* 2006-046/4. *available from <http://ssrn.com/abstract=903116>*.
- [14] Lord, R., and Kahl, C. (2006,1). Optimal Fourier inversion in semi-analytical option pricing. *available from <http://ssrn.com/abstract=921336>*.
- [15] Lord, R., and Kahl, C. (2006,2). Why the rotation count algorithm works. *Tinbergen Institute Discussion Paper* 2006-065/2. *available from <http://ssrn.com/abstract=921335>*.
- [16] Price, K., Storn, R., Lampinen, J. (2005). Differential Evolution. A practical approach to global optimization. *Springer*.
- [17] Schoutens, W., Simons, E. and Tistaert, J. (2004). A perfect calibration! Now what? *Wilmott Magazine* March, 66-78.
- [18] Zhu, J. (2000). Modular pricing of options: An application of Fourier analysis. *Springer*.

## Appendix: Characteristic functions

The derivation of the characteristic functions of stochastic volatility models can be found in Gatheral (2005), Cont and Tankov (2003) and Zhu (2000). This appendix lists the characteristic functions of the models described in Section 2.

**Heston model.** The characteristic function of the Heston model is given by (28).

**Bates model.** The characteristic function of the Bates model is

$$\begin{aligned}\phi(\omega) = & \exp\{\eta\kappa\theta^{-2}((\kappa - \rho\theta\omega i - d)T - 2\ln(\frac{1 - ge^{-dT}}{1 - g})) \\ & + \sigma_0^2\theta^{-2}(\kappa - \rho\theta\omega i - d)\frac{1 - e^{-dT}}{1 - ge^{-dT}} \\ & - \lambda\mu_J i\omega T + \lambda T((1 + \mu_J)^{i\omega} \exp(\sigma_J^2(i\omega/2)(i\omega - 1)) - 1)\},\end{aligned}\quad (33)$$

where  $d$  and  $g$  are given by (29) and (30).

**Barndorff-Nielsen-Shephard model.** The characteristic function of this model is given by

$$\begin{aligned}\phi(\omega) = & \exp\{-a\lambda\rho(b - \rho)^{-1}t - \lambda^{-1}(u^2 + iu)(1 - e^{-\lambda t})\sigma_0^2/2 \\ & + a(b - f_2)^{-1}(b\ln(\frac{b - f_1}{b - iu\rho}) + f_2\lambda t)\},\end{aligned}\quad (34)$$

where

$$f_1 = iu\rho - \lambda^{-1}(u^2 + iu)(1 - e^{-\lambda t})/2, \quad (35)$$

$$f_2 = iu\rho - \lambda^{-1}(u^2 + iu)/2. \quad (36)$$

**Levy models with stochastic time.** The characteristic functions of the Levy models with stochastic time can be composed from the characteristic exponent of the Levy component and the characteristic function of the rate of time change according to the formula

$$\phi(\omega) = \frac{\varphi(-ic(\omega))}{\varphi(-ic(i))^{i\omega}}, \quad (37)$$

where the characteristic exponent of the Levy process  $X_t$  is the logarithm of the characteristic function of the value of the process at time  $t = 1$

$$c(\omega) = \ln E[\exp(i\omega X_1)]. \quad (38)$$

The characteristic function of the CIR stochastic clock is

$$\varphi(\zeta) = \frac{\exp(\kappa^2 \eta t / \lambda^2) \exp(2y_0 i \zeta / (\kappa + \gamma \coth(\gamma t / 2)))}{(\cosh(\gamma t / 2) + \kappa \sinh(\gamma t / 2) / \gamma)^{2\kappa \eta / \lambda^2}}, \quad (39)$$

where

$$\gamma = \sqrt{\kappa^2 - 2\lambda^2 i u}. \quad (40)$$

The characteristic function of the GOU stochastic clock is

$$\varphi(\zeta) = \exp \left[ i \zeta y_0 \lambda^{-1} (1 - e^{-\lambda t}) + \frac{\lambda a}{i \zeta - \lambda b} \left( b \ln \left( \frac{b}{b - i \zeta \lambda^{-1} (1 - e^{-\lambda t})} \right) - i \zeta t \right) \right]. \quad (41)$$

The characteristic exponent of the Variance Gamma process is

$$c(\omega) = \frac{-\ln(1 + \frac{1}{2} \omega^2 \sigma^2 \kappa - i \theta \kappa \omega)}{\kappa}, \quad (42)$$

where  $\theta$  and  $\sigma$  are drift and volatility of the subordinated arithmetic Brownian motion,  $\kappa$  is variance of the Gamma subordinator.

The characteristic exponent of the Normal Inverse Gaussian process is

$$c(\omega) = \frac{1 - \sqrt{1 + \omega^2 \sigma^2 \kappa - 2i \theta \omega \kappa}}{\kappa}, \quad (43)$$

where  $\theta$  and  $\sigma$  are drift and volatility of the subordinated arithmetic Brownian motion,  $\kappa$  is variance of the Inverse Gaussian subordinator.