Speed and biases of Fourier-based pricing choices: Analysis of the Bates and Asymmetric Variance Gamma models

Ricardo Crisóstomo

Documentos de Trabajo
Nº 64

Speed and biases of Fourier-based pricing choices: Analysis of the Bates and Asymmetric Variance Gamma models

Ricardo Crisóstomo

Documentos de Trabajo

Nº 64

September 2016

Ricardo Crisóstomo is at the Department of Research and Statistics, Spanish Securities Market Commission – CNMV, c/ Edison 4, 28006 Madrid; and National Distance Education University – UNED, c/ Bravo Murillo 38, 28015 Madrid. The opinions in this article are the sole responsibility of the author and they do not necessarily coincide with those of the CNMV. Email: rcayala@cnmv.es

Ricardo Crisóstomo is a member of the Research, Statistics and Publications Department, CNMV.

Ricardo Crisóstomo pertenece al Departamento de Estudios, Estadísticas y Publicaciones de la CNMV.

ISSN (edición electrónica): 2172-7147

Maqueta: Composiciones Rali, S.A.

Abstract

This paper compares the CPU effort and numerical biases of six Fourier-based implementations. Our analyses focus on two jump models that can consistently price options with different strikes and maturities: (i) the Bates jump-diffusion model, which combines jumps with stochastic volatility and (ii) the Asymmetric Variance Gamma (AVG) model, a pure-jump process where an infinite number of jumps can occur in any interval of time. We show that both truncation and discretization errors significantly increase as we move away from the diffusive Black-Scholes-Merton dynamics. While most pricing choices converge to the Bates reference values, Attari's formula is the only Fourier-based method that does not completely blow up in any AVG problematic region. In terms of CPU speed, the strike vector computations proposed by Zhu (2010) significantly improve the computational burden, rendering the use of fast Fourier transforms and plain delta-probability decompositions inefficient.

Keywords: Jump processes, Bates model, Variance Gamma, Fourier transforms, pricing errors, speed comparisons.

JEL Classification: G13, C52, C63.

Table of Contents

List of Tables

1. Introduction

Since the seminal papers of Black and Scholes (1973) and Merton (1973), processes where the asset price diffuses continuously have been extensively used in risk management and option pricing. Diffusion models may exhibit a variety of forms, including stochastic volatility, mean-reversion or seasonality, and their widespread use highlights the success that these models have achieved in financial modelling. Yet casual observation reveals that the prices of traded assets routinely undergo jumps. Discontinuities can occur, for instance, due to unexpected news, due to trading restrictions or simply because there is a substantial imbalance between buy and sell orders.

The importance of jump modelling becomes evident if we analyze the prices of short dated out-of-the-money (OTM) options. The value of these contracts critically depends on the occurrence of extreme underlying movements. However, empirical studies have shown that diffusion-only models cannot consistently generate the asymmetry and fat-tails that are routinely implied by OTM option prices (see, Bakshi, Cao, and Chen, 1997 or Cont and Tankov, 2004)[1].

This paper contributes to the option pricing literature by examining the CPU speed and accuracy of six Fourier-based pricing choices. Specifically, our analyses focus on two jump-related models that have been proposed as a framework to consistently price options with different strikes and maturities. First, the Bates (1996) jump-diffusion model, which blends the Heston (1993) dynamics with lognormally distributed price jumps. Second, the Asymmetric Variance Gamma of Madan, Carr, and Chang (1998), a purely discontinuous process where the underlying assets evolve through a combination of many small jumps and rare big moves.

Both models are implemented by means of characteristic functions. Fourier transforms are rapidly gaining traction in finance and most of the option pricing models developed in the last the last decade have relied on characteristic functions to obtain option prices. Thus, a better understanding of the different Fourier implementations is paramount in order to avoid pricing errors. Specifically, we investigate the speed and biases of a broad range of Fourier pricing choices: Delta-probability decompositions, fast Fourier transforms and Carr-Madan's and Attari's formulas, while also considering strike vector computations for all methods where vectorization is possible.

---

1 Foresi and Wu (2005) also attribute the relatively higher prices of OTM puts to a combination of higher demand stemming from concerns about market crashes and limited supply due to hedging difficulties.

The rest of the paper is organized as follows. Section 2 reviews the use of characteristic functions and explains the numerical setup. Section 3 present the Bates model and compares the accuracy and speed of the different implementation choices. Section 4 describes the AVG model and considers three regions where Fourier methods can lead to notably different accuracies. Finally, section 5 summarizes our conclusions.

2. Characteristic functions for option pricing

Under the no-arbitrage paradigm, option prices can be calculated as the present value of the expected option payoff under the risk-neutral measure

$$V_0 = e^{-rT} E_Q \left[H(S_t) \right] \tag{2.1}$$

where V_0 is the option value at time $t = 0$, S_t the underlying price, r the risk-free rate, T the time to maturity, $H(S_t)$ is the option payoff and $E_Q[\bullet]$ denotes the expectation operator under the risk-neutral measure.

For many pricing process, the expected option payoff can be computed in terms of the underlying asset's density function. For instance, the payoff of a European call with strike K and expiration T is given by $H(S_t) = (S_T - K)^+$. Thus, its present value at time $t = 0$ can be obtained as

$$C(T,K) = e^{-rT} \int_0^\infty (S_T - K)^+ q(S_T) dS_T \tag{2.2}$$

where $q(S_T)$ is the risk-neutral density of the underlying asset S_t at the terminal date T. However, there are numerous asset processes that do not exhibit a tractable density. For these cases, pricing models generally rely on characteristic functions in order to obtain option prices. Characteristic functions are defined as the Fourier transform of the probability density functions. Thus, both functions exhibit a one-to-one correspondence and all the probabilistic evaluations that can be performed through a tractable density can be also obtained with characteristic functions. Furthermore, the characteristic functions of many asset specifications, particularly in connection to stochastic volatility and jumps, exhibit simpler analytical forms and are more tractable than their corresponding density functions.

The existing literature considers several alternatives to compute options prices using characteristic functions. In this paper we analyze six choices that can be broadly categorized into four approaches: the delta-probability decomposition, the Carr-Madan formula, the Attari formula and the fast Fourier transform.

2.1 The Delta-Probability Decomposition (DPD)

The DPD was initially developed by Heston (1993). By expanding (2.2), it is straightforward to show that the price of a European call can be expressed as

$$C(T,K) = S_0 \ \Pi_1 - e^{-rT} K \ \Pi_2 \tag{2.3}$$

where Π_1 and Π_2 are two probability-related quantities. Specifically, Π_1 is the option delta while Π_2 is the risk-neutral probability of exercise $P(S_T > K)$.

In the Black-Scholes-Merton (BSM) model and other simple pocesses, these probabilities can be directly computed in terms of the underlying asset density function. However, for processes that do not exhibit a tractable density, Bakshi and Madan (2000) show that these probabilities can be computed as

$$\Pi_1 = \frac{1}{2} + \frac{1}{\pi}\int_0^{\infty} \mathrm{Re}\left[\frac{e^{-iw\ln(K)}\psi_{\ln S_T}(w-i)}{iw\psi_{\ln S_T}(-i)}\right] dw \qquad (2.4)$$

$$\Pi_2 = \frac{1}{2} + \frac{1}{\pi}\int_0^{\infty} \mathrm{Re}\left[\frac{e^{-iw\ln(K)}\psi_{\ln S_T}(w)}{iw}\right] dw \qquad (2.5)$$

Where $\psi_{\ln S_T}$ is the characteristic function of the log-asset price and $\mathrm{Re}[\bullet]$ denotes the real operator. European call prices can be obtained by first computing Π_1 and Π_2, and then substituting these values into (2.3), whereas European puts can be determined through the put-call parity. We refer to Crisostomo (2014) for a mathematical derivation and an implementation of the DPD method in MATLAB.

In a comprehensive survey, Schmelzle (2010) concludes that the integrands in (2.4) and (2.5) decay rapidly and can be approximated through numerical integration. However, the DPD implementation still faces three potential shortcomings:

1. **Discontinuities in the integrand functions**. The characteristic function of many stochastic volatility and jump-related processes contains a complex logarithm that may generate numerical instability. For instance, Schobel and Zhu (1999) give several examples where Heston's original characteristic function shows discontinuities and numerical integration may lead to incorrect option prices. This problem, however, can be circumvented in many models by an appropriate reformulation of the underlying characteristic function (Albrecher et al., 2007 and Lord and Kahl, 2010).

2. **Singularity at w= 0.** The integrands in (2.4) and (2.5) are not defined at the lower integration limit. Lewis (2001) analyzes this singularity and concludes that the integrands are finite as w tends to zero. Nevertheless, this divergence should be treated with caution, since inappropriate handling can lead to pricing errors.

3. **Number of evaluations:** To obtain option prices through the DPD, three characteristic function evaluations are required per integration point (two for Π_1 and another one for Π_2). Thus, if the integration grid is divided into N points, $3N$ evaluations are needed per option priced or $3NM$ for a set of M options. While this may not be a problem for occasional pricing, the CPU effort can become burdensome when calculating many option prices simultaneously or in real-time contexts.

2.2 Strike Vector Computations

Zhu (2010) proposes a simple yet effective trick to reduce the computational effort of the DPD and other Fourier methods. The key insight is that the required characteristic function evaluations, both in Π_1 and Π_2, differ for each expiry, but are independent of the strike. Therefore, for a given T, characteristic function values can be computed once and re-used to price options with different strikes. This idea can be implemented through the use of vectorization or by a catching technique, as suggested by Kilin (2011).

Specifically, if we introduce a vector of strikes $\mathbf{K}$ in the calculation of (2.4) and (2.5), the probability vectors, Π_1 and Π_2, are given by

$$\mathbf{\Pi_1} = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left[\frac{e^{-iw\ln(K)}\psi_{\ln S_T}(w-i)}{iw\psi_{\ln S_T}(-i)}\right] dw \tag{2.6}$$

$$\mathbf{\Pi_2} = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left[\frac{e^{-iw\ln(\mathbf{K})}\psi_{\ln S_T}(w)}{iw}\right] dw \tag{2.7}$$

And, thus, the vector of call prices can be computed as

$$\mathbf{C}(T,\mathbf{K}) = S_0\ \mathbf{\Pi_1} - e^{-rT}\mathbf{K}\ \mathbf{\Pi_2} \tag{2.8}$$

Since the characteristic function evaluations are typically the most burdensome part of the calculations, vectorization significantly reduces the CPU effort while preserving two distinct advantages of the DPD: (i) the flexibility to choose any strikes and any integration method and (ii) the intuitive probabilistic pricing á la Black-Scholes.

2.3 Combining Π_1 and Π_2 into a single integral

Attari (2004) proposes a DPD reformulation that calculates option prices through a single integral. Specifically, by exploiting the similarities in Π_1 and Π_2, Attari's formula merges the integrands in (2.4) and (2.5) into a single pricing expression of the form

$$C(T,K) = S_0 - e^{-rT}K\left(\frac{1}{2} + \frac{1}{\pi}\int_0^\infty I_A(w)dw\right) \tag{2.9}$$

where

$$I_A(w) = \frac{(\mathrm{Re}(\psi_{\ln S_T}(w)) + \frac{\mathrm{Im}(\psi_{\ln S_T}(w))}{w})\cos(w\ln(K)) + (\mathrm{Im}(\psi_{\ln S_T}(w)) - \frac{\mathrm{Re}(\psi_{\ln S_T}(w))}{w})\sin(w\ln(K))}{1+w^2} \tag{2.10}$$

Compared to the integrands in the DPD, $I_A(w)$ includes a quadratic term in the denominator, ensuring a faster decay rate. Furthermore, strike vectorizations can also be employed to speed up the computations, since the characteristic function evaluations are independent of the strike. On the downside, this method does not provide the risk-neutral probabilities or the option's delta, and therefore these figures must be calculated separately, if needed.

2.4 The Carr-Madan formula and the fast Fourier transform (FFT)

The FFT is an algorithm designed to compute Fourier transforms in an efficient way. The first application for option pricing was developed by Carr and Madan (1999). The algorithm exploits periodicities and symmetries in the characteristic function evaluations to reduce the number of operations. For a given maturity, the FFT allows the simultaneous calculation of option prices for a variety of strikes.

The Modified Call Price

Since the FFT can only be used in square-integrable functions, Carr-Madan's approach considers a modified call price where a dampening factor $e^{\alpha \ln(K)}$ is introduced to avoid the divergence at $w = 0$

$$C_{\text{mod}}(T, K) = e^{\alpha \ln(K)} C(T, K) \tag{2.11}$$

where $C_{\text{mod}}(T, K)$ is the modified call price and $\alpha > 0$ is the dampening parameter. Using the Fourier inversions, Carr-Madan's paper shows that the original call price can be recovered as:

$$C(T, K) = \frac{e^{-\alpha \ln(K) - rt}}{\pi} \int_0^{\infty} \operatorname{Re}\left[\frac{e^{-iw\ln(K)} \psi_{\ln S_T}(w - (\alpha + 1)i)}{\alpha^2 + \alpha - w^2 + i(2\alpha + 1)w} \right] dw \tag{2.12}$$

where $\psi_{\ln S_T}$ is the characteristic function of the log-asset price.

Integration with the Fast Fourier Transform

Although (2.12) can be directly used to compute call prices, it is common to evaluate it through the FFT. The FFT specifically computes sums of the form:

$$y(m) = \sum_{n=1}^{N} e^{-i\frac{2\pi}{N}(m-1)(n-1)} x(n) \quad \textit{for}\ m = 1, \ldots, N \tag{2.13}$$

Therefore, before applying the algorithm, the call prices in (2.12) should be expressed in the required summation form. The first step is to approximate the integral by a grid of N equidistant points, thus establishing an upper integration limit of $N\Delta w$. Next, by setting the grid points as $w_n = (n-1)\Delta w$ and using the trapezoidal rule, the price of a *single* European call can be computed as

$$\widehat{C}(K) \approx \sum_{n=1}^{N} e^{-iw_n \ln(K)} f(w_n) \Delta w \tag{2.14}$$

where

$$f(w_n) = e^{\alpha \ln(K) - rT} \frac{\psi_{\ln S_T}(w_n - (\alpha + 1)i)}{\alpha^2 + \alpha - w_n^2 + i(2\alpha + 1)w} \tag{2.15}$$

However, the FFT algorithm takes an N-sized vector $x(n)$as an input and returns another N-sized vector $y(m)$ as output. Consequently, the choice of N simultaneously determines the number of strikes and the integration grid size. In addition, two other FFT constraints must be respected. First, the strikes must be placed at an equal distance in the log space[2]. Second, the Nyquist relation $\Delta k \Delta w = 2\pi / N$ must also be obeyed, effectively imposing an inverse relationship between the integration step width Δw and the sparseness of the output prices.

Putting all together, the prices of N call options can be simultaneously obtained as

$$\widehat{C}(k_m) \approx \sum_{n=1}^{N} e^{-i\frac{2\pi}{N}(n-1)(m-1)} g(w_n) \quad for\, m = 1, \ldots., N \tag{2.16}$$

where

$$g(w_n) = e^{ibw_n + \alpha k_m - rT} \frac{\psi_{\ln S_T}(w_n - (\alpha+1)i)}{\alpha^2 + \alpha - w_n^2 + i(2\alpha+1)w} \Delta w \tag{2.17}$$

Finally, to harness the speed advantages of the FFT, the sums in (2.17) should be divided in two sequences: one with the odd terms and another with the even ones. The key computational insight is that the characteristic function evaluations required in the odd sequence are repeated for the even one. Thus, previously computed values can be used to reduce the number of operations. This strategy can be reinforced by decomposing the odd and even sequences into two additional subsequences. And continuing this decimation until we obtain $N/2$ subsequences of length 1, the FFT algorithm is able to reduce the computational effort from an order of N^2 to an order of $N \log_2(N)$.

FFT limits and alternatives

The main FFT drawbacks stem from the restrictions imposed in the strike and integration grids:

1. **Strike grid**. A fully efficient decimation strategy requires the number of strikes to be a power of 2. Moreover, those 2^N strikes must be equidistantly placed in the log space. This means that the number and location of the resulting FFT prices will rarely match our needs. Prices closer to our strike requirements can be computed by increasing N or by interpolating across the prevailing strikes, but both strategies will impact the merits of the FFT: a higher N implies calculating more option prices than needed, whereas interpolation affects pricing accuracy.
2. **Relationship between the strike and integration grid.** The constraint $\Delta k \Delta w = 2\pi / N$ imposes an inverse relationship between the integration step width and the output FFT prices. Specifically, a finer integration grid will lead to coarser strikes, and if we try to improve the pricing accuracy by reducing Δw the output prices will be more dispersed, thus increasing the need for interpolation.
3. **Integration methods**: Since the FFT requires equidistant integration points, only the most simple quadrature rules can be used to recover option prices.

2 To this end, we define the strike grid as $k_m = -k_{\max} + (m-1)\Delta k + \ln(S_0)$ with $m = 1, \ldots., N$ and $k = \ln K$. This choice entails setting the FFT strikes symmetrically centered around $K = S_0$.

This compares unfavorably to other Fourier-based methods, where more efficient integration techniques can be used to speed up the calculations.

Summing up, the FFT exhibits a tradeoff between: (i) the accuracy of integration, (ii) the need for interpolation and (iii) the speed of computation. Carr and Madan (1999) suggest using an N= 4096. However, for most equity underlyings, there are rarely more than 20 or 30 actively traded strikes per maturity. Therefore, if we employ a high N, only a small fraction of the final FFT call prices will fall within the usual trading ranges, which in turn means that many of the resulting prices might be left unused[3].

To address these constraints, Chourdakis (2005) introduces a Fractional FFT method (FRFT). By relaxing the restriction $\Delta k \Delta w = 2\pi / N$, the FRFT provides greater flexibility in the construction of the strike and integration grids, thus avoiding unnecessary function evaluations. Similarly, Fang and Oosterlee (2009) propose Fourier-cosine expansions as an alternative to calculate the Fourier integrals, showing that the so-called COS method achieves better convergence rates that the quadrature required in the FFT. However, neither of these methods relax the requirement to place all the strike and integration points equidistantly, which is a fundamental FFT constraint.

Alternatively, the Carr-Madan formula in (2.12) can be also used to price call options without further manipulation. For instance, Matsuda (2004) uses a slightly modified version of (2.12) and reports accurate prices and negligible approximation errors for a variety of option models. In addition, Carr-Madan's formula can be optimized through the use of strike vector computations, since the characteristic function evaluations are independent of the strike.

2.5 Numerical exercises: Setup and error analyses

We investigate the pricing biases and computational speed of six pricing choices:

- **DPD**: Delta Probability Decomposition. Call values are individually computed through equations (2.3) to (2.5).
- **DPD-OPT:** Optimized DPD. Strike vector computations are used to simultaneously compute call values for a variety of strikes. Equations (2.6) to (2.8) are used.
- **AT-OPT:** Optimized Attari approach. Call values are computed with equations (2.9) and (2.10). CPU speed is optimized through strike vectorizations.
- **FFT:** Standard FFT. A single fast Fourier transform is used to obtain option prices. Vector operations (instead of loops) are used to improve the performance. After experimenting with different values, we settle for an α= 1.75, which delivers a 10^{-10} accuracy for all the models tested. Options that do not fall within the FFT strike grid are exponentially interpolated.

3 For example, out of the 4096 FFT prices calculated by Carr and Madan (1999) only about 67 fall within the ±20% log-strike interval (Chourdakis, 2005).

- **FFT-SA:** Strike-adjusted FFT. Call values are determined by successive FFT runs. Strike grids are adjusted to cover all the required options in at least one FFT run, thus avoiding interpolation.
- **CM-OPT:** Optimized Carr-Madan formula. Call values are computed using equation (2.12) and strike vector computations.

These Fourier implementations can be subject to three forms of error:

1. **Truncation error:** All methods require evaluating integrals in $w = [0, \infty)$. To numerically approximate such integrals, the integration range must be truncated by choosing an appropriate upper limit, hence introducing a truncation error. For a given integration endpoint, the order of truncation errors will be different depending on (i) the mathematical model used to describe the underlying asset dynamics and (ii) the particular implementation employed to obtain option prices. The rationale is that characteristic functions for different underlying models exhibit different decay rates, whereas the Fourier integrands described in subsections 2.1 to 2.4 also portray varying decay speeds. Lee (2004) provides a comprehensive analysis of truncation errors in Fourier pricing methods.
2. **Discretization error:** The truncated integral is evaluated by using a finite integration grid, thus introducing a sampling error. Different characteristic functions and Fourier implementations also affect the smoothness of the integrands, thus impacting discretization errors. To facilitate comparisons, in our analyses all the option prices are computed through the trapezoidal rule. However, most pricing choices specifically support non-equidistant integration. Therefore, the results for the DPD, DPD-OPT, AT-OPT and CM-OPT should be interpreted as a lower-limit estimate on the potential improvement that such methods offer over the FFT and FFT-SA.
3. **Interpolation error:** This error arises when a pricing method does not provide the price for a desired strike. Consequently, in our setting, this error is specific to the FFT, since all the other variants can evaluate any required strike.

To discriminate between these three errors, option prices in our *accuracy comparisons* are calculated (i) with a high precision of 10^{-10}, (ii) using a common integration domain and (iii) using comparable integration grids of size 2^N. Conversely, for our *speed comparison,* the accuracy is set at a more realistic 10^{-4} and we relax the integration domain and 2^N sampling constraints, thus allowing each method to optimize its integration requirements. Specifically, we compare how fast each method is able to price a variable number of options, covering a wide range of needs from 1 to 2500 options. Calculations are performed using an Intel Core i7-3770 CPU @ 3.40GHz and 16 GB RAM.

2.6 A first test with the BSM model

To form a baseline, we first apply all Fourier implementations to the BSM model, whose characteristic function is given by

$$\psi_{\ln(S_t)}^{BSM}(w) = e^{iw[\ln(S_0)+(r-0.5\sigma^2)t]-0.5w^2\sigma^2 t} \tag{2.18}$$

2.6.1 Pricing accuracy in the BSM model

We employ the parameters: S_0= 50, σ= 0.25 and r= 0.05. Accuracy is evaluated at six option configurations, spanning three different strikes K= [30, 50, 70] and two maturities T= [0.1, 1]. The integration range is set at w= (0, 100] and reference values are computed through the BSM closed-form solution. Table 1 shows the results.

BSM pricing results for different implementations and grid sizes. Shaded areas indicate an accuracy of 10^{-10} TABLE 1

		$T=0.1$			$T=1$		
Method	N	$K=30$	$K=50$	$K=70$	$K=30$	$K=50$	$K=70$
DPD / DPD-OPT	16	20.1496269249	**1.7004462835**	**0.0000139309**	24.1223640619	6.1788617825	1.1638295106
	32	**20.1496256242**	1.7004462835	0.0000139309	21.5036300968	**6.1679994652**	**0.8986170065**
	64	20.1496256242	1.7004462835	0.0000139309	**21.5036288308**	6.1679994652	0.8986170045
	128	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	512	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	1024	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	4096	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
AT-OPT	16	-8.7061837657	-27.155363106	-28.855795459	-7.2665016957	-22.687643605	-27.943228414
	64	19.2367194274	0.7875400867	-0.9128922660	20.5907226339	5.2550932683	-0.0142891923
	128	20.1335448783	1.6843655375	-0.0182208247	21.4875480848	6.1519187192	0.8825362585
	256	20.1496204558	1.7004411150	0.0000087624	21.5036236623	6.1679942967	0.8986118360
	512	**20.1496256242**	**1.7004462835**	**0.0000139309**	**21.5036288308**	**6.1679994652**	0**.8986170045**
	1024	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	4096	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
FFT	16	28.2798382222	8.7539883498	5.7137666216	30.9956621087	13.3705415064	6.8339658821
	64	20.1839002131	1.7436317072	0.0429040929	21.5403695644	6.2112193777	0.9409450521
	128	20.1404210818	1.7004848691	0.0000664350	21.4968737300	6.1680380513	0.8969426660
	256	20.1403824743	**1.7004462835**	0.0000133825	21.4968351258	**6.1679994652**	0.8969030154
	512	20.1403824743	1.7004462835	0.0000133824	21.4968351258	6.1679994652	0.8969030154
	1024	20.1403824743	1.7004462835	0.0000133824	21.4968351258	6.1679994652	0.8969030154
	4096	20.1403824743	1.7004462835	0.0000133824	21.4968351258	6.1679994652	0.8969030154
FFT-SA	16	28.5412052262	8.7539883498	5.7155184631	31.3434804938	13.3705415064	6.8210738060
	64	20.1931245646	1.7436317072	0.0428858381	21.5471484645	6.2112193777	0.9415371960
	128	20.1496642149	1.7004848691	0.0000525116	21.5036674216	6.1680380513	0.8986555860
	256	**20.1496256242**	**1.7004462835**	**0.0000139309**	**21.5036288308**	**6.1679994652**	**0.8986170045**
	512	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	1024	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	4096	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
CM-OPT	16	28.5412052262	8.7539883498	5.7155184631	31.3434804938	13.3705450641	6.8210738060
	64	20.1931245646	1.7436317072	0.0428858381	21.5471484645	6.2112193777	0.9415371960
	128	20.1496642148	1.7004848691	0.0000525116	21.5036674216	6.1680380513	0.8986555860
	256	**20.1496256242**	**1.7004462835**	**0.0000139309**	**21.5036288308**	**6.1679994652**	**0.8986170045**
	512	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	1024	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
	4096	20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045
Ref. value		20.1496256242	1.7004462835	0.0000139309	21.5036288308	6.1679994652	0.8986170045

As Table 1 shows, most Fourier methods converge well to the reference BSM value. The only exceptions are the OTM and ITM strikes in the standard FFT, which fail to provide full accuracy. Specifically:

1. The DPD and DPD-OPT require between 16 and 64 sampling points to achieve an accuracy of 10^{-10}, the lowest of all methods.
2. For the AT-OPT, full convergence is achieved with 512 points. However, due to discretization errors, negative option prices are obtained for all strikes and maturities when the integration grid is small.
3. The FFT achieves an accuracy of 10^{-10} for the two ATM options. Conversely, a single FFT run is unable to attain full convergence for the other strikes. The biases stem from the FFT constraints explained in section 2.4. Specifically, a single FFT grid cannot exactly match all the required strikes, and therefore the OTM and ITM prices have been exponentially interpolated, introducing an interpolation error.
4. When all the strikes are evaluated through an FFT grid, the FFT-SA delivers full convergence for all options. 256 sampling points are required to attain full accuracy.
5. Finally, the CM-OPT results mirror those of the FFT-SA. Both methods rely on the same pricing equation and can evaluate any required strike. Therefore, when the same sampling grid and integration domain is used, they are equivalent in terms of accuracy.

In summary, both truncation and sampling errors are small and easy to manage in the BSM model: High precision values can be obtained in the domain w = (0, 100] integrating with sizes between 16 and 512 points (i.e. 0.16 to 5.12 points per unit of w). These results derive from the well-behaved diffusive properties of the geometric Brownian motion, which in turns entails a smooth and rapidly decaying characteristic function.

2.6.2 Computational speed in the BSM model

To investigate the CPU effort, we first obtain the w-ranges required to attain full convergence and the number of sampling points that deliver an accuracy of 10^{-4}. Reported times are calculated by averaging the CPU effort in 100 independent runs. Table 2 shows the results.

CPU times required to achieve a 10^{-4} accuracy in the BSM model [milliseconds] TABLE 2

Method	W-range	Minimum N	N. of options priced					
			1	10	25	100	500	2500
DPD	(0, 89]	26	0.17591	1.79141	4.37267	17.39508	88.43997	435.05447
DPD-OPT	(0, 89]	26	0.17591	0.23206	0.27932	0.35656	1.08824	3.63379
AT-OPT	(0, 79]	173	0.15621	0.24390	0.32789	0.87359	2.68967	18.10151
FFT	(0, 77]	128	0.52439	0.52439	0.52439	0.52439	5.78116	317.90680
CM-OPT	(0, 77]	97	0.10406	0.15385	0.19354	0.50436	1.68204	11.28898

1. Despite requiring the highest w-range, the DPD and DPD-OPT only need 26 sampling points to achieve a 10^{-4} accuracy. Performance-wise, restarting computations for each new option clearly drags down the DPD speed, as CPU times increase almost linearly with the number of options. Conversely, strike vector computations significantly improve the computational efforts. For instance, when 2500 options are considered, the DPD-OPT is roughly 120 times faster than the unoptimized DPD. Moreover, due to its high sampling efficiency, the DPD-OPT is the fastest alternative when pricing 100 or more options.

2. The AT-OPT single-integral strategy is faster than the DPD-OPT for any fixed integration grid. However, to achieve a comparable 10^{-4} accuracy, the AT-OPT requires almost 7 times more integration points (173 vs 26). Consequently, as the number of options increases, the DPD-OPT more than offsets the initial AT-OPT advantages.

3. Leaving aside interpolation biases, the FFT requires 128 points to achieve the target 10^{-4} accuracy. Therefore, the FFT always computes a minimum of 128 option prices, impacting its performance when fewer prices are required. The FFT results improve when more options are considered, and quickly surpass the speed of the unoptimized DPD. However, on average, the FFT is still 5, 7 and 17 times slower than the AT-OPT, CM-OPT and DPD-OPT respectively.

4. Compared to the FFT, the CM-OPT has three main advantages. It allows: (i) pricing any required strikes (ii) avoiding interpolation biases and (iii) achieving a 10^{-4} accuracy with fewer integration points. As a result, the CM-OPT is both faster and more accurate than the FFT, thus rendering this method inefficient. Following these figures, we decided not to pursue the speed comparison for the FFT-SA, which requires at least twice the FFT's computing times and still cannot improve the CM-OPT accuracy.

To sum up, under the BSM dynamics the DPD-OPT is the most efficient when pricing 100 or more options, whereas the CM-OPT is the best performer for pricing needs of 25 or less.

3. The Bates Jump-diffusion Model

3.1 Model description

In 1996 Bates proposed a modelling framework which blends the Heston model with lognormally distributed price jumps. Under the risk-neutral measure, the Bates dynamics are given by

$$\begin{aligned} dS_t &= (r - \lambda\ \mu_J)S_t dt + \sqrt{V_t} S_t dW_t^1 + J_t S_t dN_t \\ dV_t &= a(\overline{V} - V_t)dt + \eta\sqrt{V_t} dW_t^2 \\ dW_t^1 dW_t^2 &= \rho dt \end{aligned} \tag{3.1}$$

where S_t is the price of the underlying asset at time t, r the risk free rate, V_t the variance at time t, $\overline{V}$ the long-term variance, a the variance mean-reversion speed, η the volatility of the variance process and dW_t^1, dW_t^2 are two Weiner processes with correlation ρ. In addition, N_t is a Poisson process with intensity λ, and J_t are the jump sizes, which are lognormally distributed with an average jump size μ_J and standard deviation v_J. Therefore, conditional on a jump occurring, the logarithm of the jump size is normally distributed with parameters

$$\ln(1 + J_t) \sim N\left(\ln(1 + \mu_J) - \frac{v_J^2}{2}, v_J\right) \tag{3.2}$$

The rationale for mixing stochastic volatility and jumps is based on empirical analyses. Evidence show that volatility may change drastically over time and that asset prices can experience price jumps. As a result, observed returns as well as market expectations are typically characterized by distributions that exhibit substantial asymmetries and fat-tails, particularly in the short-term (Cont, 2001). Furthermore, empirical studies generally support the main features of the Heston model –mean-reverting volatility and correlated volatility and asset shocks–, concluding that Heston dynamics provide a good fit to the prices of long-term options. Bakshi, Cao, and Chen (1997) and Crisostomo (2014) among others, validate this claim.

However, the diffusive behavior of the Heston model struggles to generate the asymmetric and leptokurtic distributions that are routinely implied by short-term options (see, for example Jones, 2003 or Sepp, 2003). To address this problem, the Bates model introduces a lognormal jump component which complements the Heston dynamics. As explained in Carr and Wu (2003), lognormal jumps can significantly contribute to explaining the price of short-term options, but their smile effects flatten out quickly in longer time periods. Consequently, by combining stochastic volatility and lognormal jumps, the Bates model offers a versatile modelling scheme that can be used to accommodate both the short and the long end of the volatility surface.

3.2 Bates characteristic function

Since the lognormal jumps are statistically independent from the Heston dynamics, the Bates characteristic function can be obtained by multiplying its individual components

$$\psi_{\ln(S_t)}^{Bates}(w) = \psi_{\ln(S_t)}^{Heston}(w)\,.\,\psi_{\ln(S_t)}^{Jump}(w) \tag{3.3}$$

For the Heston model, we follow the formulation in Gatheral (2006), which is free of the complex logarithm problem mentioned in section 2 (Lord and Kahl, 2010). For the lognormal jump, we use the derivation in Schoutens, Simons, and Tistaert (2004). The corresponding characteristic functions, expressed in compact form, are given by

$$\psi_{\ln(S_t)}^{Heston}(w) = e^{[C(t,w)\bar{V} + D(t,w)V_0 + iw\ln(S_0 e^{rt})]} \tag{3.4}$$

$$\psi_{\ln(S_t)}^{Jump}(w) = e^{[J(t,w) + iw\ln(S_0 e^{-\lambda \mu_J t})]} \tag{3.5}$$

multiplying these and rearranging terms yields

$$\psi_{\ln(S_t)}^{Bates}(w) = e^{[C(t,w)\bar{V} + D(t,w)V_0 + J(t,w) + iw\ln(S_0 e^{(r-\lambda \mu_J)t}]} \tag{3.6}$$

with

$$\begin{aligned}
C(t,w) &= a\left[r_- \cdot t - \frac{2}{\eta^2} ln\left(\frac{1-ge^{-ht}}{1-g}\right)\right] \\
D(t,w) &= r_- \frac{1-e^{-ht}}{1-ge^{-ht}} \\
J(t,w) &= \lambda t\left[\left(1+\mu_J\right)^{iw} e^{\frac{1}{2}v_J^2 iw(iw-1)} - 1\right] \\
r_\pm &= \frac{\beta \pm h}{\eta^2};\;\; h = \sqrt{\beta^2 - 4\alpha\gamma};\;\; g = \frac{r_-}{r_+} \\
\alpha &= -\frac{w^2}{2} - \frac{iw}{2};\;\; \beta = a - \rho\eta iw;\;\; \gamma = \frac{\eta^2}{2}
\end{aligned} \tag{3.7}$$

In particular, $C(t,w)\bar{V}$ and $D(t,w)V_0$ come from the Heston model, $J(t,w)$ is a jump-specific component, while $iw\ln(S_0 e^{(r-\lambda \mu_J)t})$ accounts for the combined risk-neutral drift.

3.3 Numerical results

3.3.1 Pricing accuracy in the Bates model

Our parameter set is taken from Duffie, Pan, and Singleton (2000): S_0= 100, V_0= 0.008836, $\bar{V}$= 0.014, a= 3.99, η= 0.27, r= 0.0319, ρ= -0.79, λ= 0.11, μ_J= -0.12 and v_J = 0.15. The accuracy is evaluated at three strikes K= [60, 100, 140] and two tenors T = [0.1, 1]. Due to the jump component, the Bates characteristic function exhibits fatter tails than the BSM model, thus increasing truncation errors. Specifically, to achieve a 10^{-10} accuracy, the integration range needs to be expanded to w = (0, 500] in most option configurations. Working in this domain, reference values are obtained through the concurrent prices of the AT-OPT and the CM-OPT, integrating with 10^6 points. Our reference values reproduce the numerical results in Broadie and Kaya (2006) which considers the same parameter set and an expiry T= 5. Table 3 shows the convergency for all options.

Bates pricing results for different implementations and grid sizes. Shaded areas indicate an accuracy of 10^{-10}

TABLE 3

		$T=0.1$			$T=1$		
Method	N	$K=60$	$K=100$	$K=140$	$K=60$	$K=100$	$K=140$
DPD / DPD-OPT	16	120.388979560	1.5993860470	41.565546998	141.124118123	9.8443801278	46.170596585
	64	40.1935961577	1.4818199384	0.0142396119	42.0421746717	6.7681519218	0.4588922289
	128	40.1913715042	1.4817911118	0.0000689118	41.9030506537	6.7577771807	0.0059976446
	256	40.1913715007	1.4817911118	0.0000688879	**41.9030506459**	**6.7577754525**	**0.0058803882**
	512	40.1913715000	1.4817911118	0.0000688882	41.9030506459	6.7577754525	0.0058803882
	1024	40.1913714998	1.4817911118	0.0000688883	41.9030506459	6.7577754525	0.0058803882
	4096	40.1913714998	1.4817911118	0.0000688883	41.9030506459	6.7577754525	0.0058803882
AT-OPT	16	-379. 72556331	-447. 5135611	-431.4957298	-372.55076003	-441.42435148	-431.1683858
	64	-40.779258363	-79.488914563	-80.96927475	-39.062068273	-74.211437357	-80.89549137
	128	15.1618983729	-23.547682032	-25.02940426	16.8735775090	-18.271697518	-25.02358118
	512	40.0305088361	1.3209284304	-0.160793800	41.7421879719	6.5969127785	-0.154982285
	1024	40.1911135720	1.4815331663	-0.000189064	41.9027927078	6.7575175143	0.0056224500
	2048	40.1913715094	1.4817911037	0.0000688734	41.9030506452	6.7577754518	0.0058803875
	4096	**40.1913715101**	**1.4817911043**	**0.0000688740**	**41.9030506459**	**6.7577754525**	**0.0058803882**
FFT	16	124.980745155	103.62458491	89.765405818	134.987904869	113.29334262	97.080477335
	64	57.5350496338	21.632547426	15.246969250	59.7231156346	27.255657862	15.746027439
	128	45.8369819055	6.6448119788	4.6790510018	47.5693863707	11.955043669	4.7328337892
	512	40.1910533056	1.4830775307	0.0013549399	41.9029321867	6.7590619372	0.0071709441
	1024	40.1897660192	1.4817911209	0.0000688774	41.9016448724	6.7577754691	0.0058810474
	2048	40.1897660026	**1.4817911043**	0.0000688606	41.9016448557	**6.7577754525**	0.0058810307
	4096	40,1897660026	1.4817911043	0,0000688606	41.9016448557	6.7577754525	0.0058810307
FFT-SA / CM-OPT	16	256.0930361179	103.62458491	58.140360108	276.812732631	113.293342621	62.813272340
	64	65.2465904830	21.632547426	15.247231379	67.2167401727	27.2556578616	15.746232159
	128	45.8384232224	6.6448119788	4.6790589109	47.5706507670	11.9550436695	4.7328168403
	512	40.1926587616	1.4830775307	0.0013544753	41.9043379324	6.7590619372	0.0071660711
	1024	40.1913715268	1.4817911209	0.0000688906	41.9030506625	6.7577754691	0.0058804048
	2048	**40.1913715101**	**1.4817911043**	**0.0000688740**	**41.9030506459**	**6.7577754525**	**0.0058803882**
	4096	40.1913715101	1.4817911043	0.0000688740	41.9030506459	6.7577754525	0.0058803882
Ref. value		40.1913715101	1.4817911043	0.0000688740	41.9030506459	6.7577754525	0.0058803882

As Table 3 shows, most pricing choices converge to the Bates reference values with two exceptions: (i) the short-term options in the DPD and DPD-OPT and (ii) the OTM and ITM options in the FFT. In particular:

1. In the DPD and the DPD-OPT full convergence is achieved for the options at $T = 1$ with an integration size of 256 points (0.51 points per unit of w). Conversely, increasing the sampling frequencies does not result in full accuracy for the options at $T = 0.1$. Since the mispricings do not taper off as N is increased, we next consider possible truncation errors. Specifically, the observed biases progressively diminish by expanding the integration domain, and full convergence is achieved with an upper limit of $w = 649$. These results highlight the higher truncation error of the DPD integrands while exposing the lower decay of the Bates characteristic function in short expiries, a finding that is consistent with Lee (2004).

2. For the AT-OPT, 4096 sampling points are required to obtain an accuracy of 10^{-10}. Therefore, no truncation errors are observed in the range $w = (0, 500]$, but discretization biases are notably higher than in other methods. First, a sampling density of 8.19 points per unit of w is required to achieve full accuracy, the highest of all methods. Second, regardless of maturity and strike, the AT-OPT significantly underprices all options with small integration grids, producing negative prices for sampling sizes as high as $N = 1024$ (2.05 points per unit of w).

3. Interpolation errors prevent a single FFT run from attaining full convergence in the OTM and ITM strikes. Conversely, for the two ATM strikes –both covered in the FFT grid–, a 10^{-10} accuracy is obtained when $N = 2048$ points.

4. Finally, the CM-OPT and FFT-SA achieve full accuracy in all options with 2048 sampling points. Therefore, these methods are free of truncation errors for an upper integration limit $w = 500$ and a sampling density of 4.10 points per w is required to deliver an accuracy of 10^{-10}.

In summary, except for small biases of order $O(10^{-3})$ or lower, no major convergence problems are observed in the Bates model, and high precision values can be obtained in the domain $w = (0, 500]$ integrating with sampling densities from 0.51 to 8.19 points. However, these results highlight the increased complexity of the Bates model compared to the BSM dynamics, which entails (i) fatter tails due to a slower decaying characteristic function, thus increasing truncation errors and (ii) a less smooth probabilistic distribution, increasing sampling errors.

3.3.2 Computational speed in the Bates model

To evaluate the CPU speed, we first obtain the w-ranges required for full convergence and the number of sampling points that deliver a 10^{-4} accuracy. As expected, the higher integration domains and sampling densities impact the required number of sampling points. Overall, computational times under the Bates model are, on average, 5 times higher than in the BSM framework. Table 4 shows the CPU times, obtained by averaging the waiting times in 100 independent runs.

CPU times required to achieve a 10^{-4} accuracy in the Bates model [milliseconds] TABLE 4

			N. of options priced					
Method	**W-range**	**Minimum N**	**1**	**10**	**25**	**100**	**500**	**2500**
DPD	(0, 649]	176	0.49055	4.92784	12.34723	49.5285	246.35524	1231.8061
DPD-OPT	(0, 649]	176	0.49055	0.69773	0.82721	1.81093	5.21412	35.49049
AT-OPT	(0, 478]	1091	1.39819	2.32650	2.79309	4.99878	24.98206	129.30169
FFT	(0, 470]	1024	24.59742	24.59742	24.59742	24.59742	24.59742	303.61574
CM-OPT	(0, 470]	622	0.33039	0.55383	1.07258	2.41625	14.33041	77.89331

1. Although both require the highest w-range, the DPD and DPD-OPT only need 176 sampling points to achieve a 10^{-4} accuracy, the lowest of all methods. CPU times increase nearly linearly without optimization, but the use of strike vectorization completely reverses the picture, making the DPD-OPT the fastest method for pricing needs of 25 options or more.

2. The AT-OPT requires 1091 points to deliver a 10^{-4} accuracy, the highest of all methods. However, vectorization clearly pays off in terms of speed: despite its lower sampling efficiency, the AT-OPT is faster than the FFT and the DPD.

3. Both the FFT and the CM-OPT minimize truncation errors. Leaving interpolation biases aside, the FFT requires integrating with $N = 1024$ points to achieve the target accuracy. As a result, the FFT always computes a minimum of 1024 options, rendering this method inefficient for low pricing requirements. Although its performance improves with the number of options, the FFT is still slower than any strike-optimized method: on average, the FFT is 8, 24 and 26 times slower than the AT-OPT, CM-OPT and DPD-OPT, respectively.

4. The CM-OPT outperforms the FFT, delivering a 10^{-4} accuracy with just 622 sampling points. Although they are based on the same approach, the CM-OPT's flexibility allows pricing any required strikes and avoids interpolation biases, thus improving both the speed and accuracy of the FFT. Specifically, the CM-OPT stands out as the fastest alternative for pricing needs of 10 options and lower.

Summing up, despite exhibiting higher computation times, the relative speed comparisons are similar to those of the BSM model: the DPD-OPT is the most efficient method when pricing a high number of options whereas the CM-OPT is the best performer for low to modest amounts. These results demonstrate the greater efficiency of the strike-vectorized methods compared to the FFT and the unoptimized DPD.

4. The Asymmetric Variance Gamma

4.1 Model description

The Variance Gamma model was introduced by Madan and Seneta (1990). However, it is its asymmetric version in Madan, Carr, and Chang (1998) which has achieved the greatest acceptance. The AVG is a purely discontinuous process where the underlying asset evolves through a combination of many small jumps and a limited number of big jumps. Under the risk-neutral measure, the AVG dynamics are given by:

$$S_t = S_0 e^{(r+\lambda)t + X(t;\,\sigma, v, \theta)} \tag{4.1}$$

with

$$\lambda = \frac{1}{v} \ln(1 - \theta v - \frac{\sigma^2 v}{2}) \tag{4.2}$$

and

$$X(t;\sigma, v, \theta) = \theta G(t;v) + \sigma G(t;v) W_t \tag{4.3}$$

where $G(t;v)$ is a Gamma distribution with mean t and variance vt, and W_t is a Weiner process N(0,1). Besides the risk free rate r, the model has three free parameters: $\sigma > 0$, $v > 0$ and θ. In broad terms, σ governs the underlying asset volatility, θ dictates the sign of the skewness, and v provides control over the kurtosis. However, except for simplifying cases[4], it is the particular combination of these three parameters which jointly determines the higher moments of the AVG distribution. We refer to Fiorani (2004) for a detailed statistical characterization.

Among jump models, the AVG presents two advantages. First, the AVG offers one the most parsimonious approaches that can consistently price options with different moneyness and maturities. Second, several empirical studies have shown that the AVG dynamics provide a very good fit to the observed equity returns; see, for instance, Rebonato (2004) or Göncü, Karahan, and Kuzubaş (2016).

4 For instance, when $\theta = 0$ the AVG distribution is symmetric, and v alone determines the excess kurtosis, which is equal to $3(1+v)$.

4.2 Characteristic function of the AVG model

The AVG model exhibits a closed-form solution for the valuation of European options. However, its numerical implementation requires working with Bessel functions of the second type and hypergeometric functions, making it complex and numerically unstable. See, for instance, Matsuda (2004).

Alternatively, the characteristic function of the AVG model is simply given by

$$\psi_{\ln(S_t)}^{AVG}(w) = \left(\frac{1}{1 - i\theta v w - (\sigma^2 v / 2) w^2} \right)^{t/v} \tag{4.4}$$

and can be directly used to calculate option prices through the Fourier methods presented in Section 2. However, some popular choices, including the FFT, can blow up for certain AVG parameter values, as reported by Itkin (2010). The next section investigates these claims.

4.3 Numerical results

4.3.1 Pricing accuracy in the AVG model

We analyze three different parameter regions.

Parameters based upon Madan, Carr, and Chang (1998)

For our *first pricing* test we employ the parameters S_0= 100, σ= 0.12136, v= 0.3, θ= -0.1436, r = 0.1. We consider three strikes K = [60, 101, 140] and two tenors T= [0.1, 1]. Despite its simple mathematical form, the slow hyperbolic decay of the AVG characteristic function complicates its practical implementation. Specifically, to obtain an accuracy of 10^{-10} across most pricing variants, the required integration domain stands at w = (0, 500] for the options at T= 1, but it explodes to w= (0, 60000] for those at T=0.1. Working through these domains, reference values can be obtained with the concurrent prices of the CM-OPT and AT-OPT. This methodology reproduces the AVG prices in Ribeiro and Webber (2004). Furthermore, it also reproduces the related example in Hirsa (2012), for which Lewis (2013) reports a high accuracy value of 11.3700278104. Table 5 shows the convergency for all options.

AVG pricing results for a parameter set based on Madan, Carr, and Chang (1998). Shaded areas indicate an accuracy of 10^{-10}. Integration performed over w = [0, 500] for T = 1 and w = (0, 60000] for T = 0.1

TABLE 5

		$T=0.1$			$T=1$		
Method	N	$K=60$	$K=101$	$K=140$	$K=60$	$K=101$	$K=140$
DPD / DPD-OPT	64	3197.76162270	29.529241721	1895.8914932	49.1153370260	10.9847710563	0.2828767412
	128	1607.65046065	14.771010245	939.47444626	45.7164396703	10.9815616823	0.1019906756
	256	812.251435965	7.3699053393	458.02998666	**45.7164396686**	**10.9815614276**	**0.1019706457**
	4096	79.9792045063	1.4022900624	2.0177043314	45.7164396686	10.9815614276	0.1019706457
	32768	40.5972174428	1.3938413039	0.0000033815	45.7164396686	10.9815614276	0.1019706457
	131072	40.5972172862	1.3938413038	0.0000033082	45.7164396686	10.9815614276	0.1019706457
	524288	40.5972172766	1.3938413037	0.0000033036	45.7164396686	10.9815614276	0.1019706457
AT-OPT	64	-13400.437512	-14855.745639	-13877.853867	-35.1355089448	-69.9887207229	-80.843471846
	512	-1613.6021231	-1813.2088139	-1707.9279004	45.5555769945	10.8206987535	-0.0588920283
	2048	-350.29134880	-416.62945631	-403.05433836	45.7164396679	10.9815614269	0.1019706449
	4096	-140.23509785	-185.30543391	-186.40540438	**45.7164396686**	**10.9815614276**	**0.1019706457**
	32768	37.2551588597	-1.9482165141	-3.3420543348	45.7164396686	10.9815614276	0.1019706457
	262144	40.5972193354	1.3938439615	0.0000061408	45.7164396686	10.9815614276	0.1019706457
	524288	**40.5972193355**	**1.3938439616**	**0.0000061410**	45.7164396686	10.9815614276	0.1019706457
FFT	64	7747.721961182	3114.40894147	1758.79256655	72,3022003954	32.1522733400	16,9592069813
	512	968.3640798850	389.314034377	219.828322682	45,5660131979	10.9828480255	0,1063975932
	1024	483.4617873522	194.478580870	109.782124358	45,5647258461	10.9815614442	0,1051087216
	2048	243.7658095435	96.542232909	55.3220233580	45,5647258295	**10.9815614276**	0,1051087049
	32768	40.83971468629	1.6331043100	0.23618661120	45,5647258295	10.9815614276	0,1051087049
	131072	40.59721650427	1.3938439653	0.00000614471	45,5647258295	10.9815614276	0,1051087049
	262144	40.59721650054	**1.3938439616**	0.00000614100	45,5647258295	10.9815614276	0,1051087049
FFT-SA / CM-OPT	64	7747.72161857	3114.40894147	1758.79248754	72.5340534309	32.1522733400	16.9354162770
	512	968.205950852	389.314034377	219.828322588	45.7177270342	10.9828480255	0.1032565134
	1024	483.383990426	194.478580870	109.782124231	45.7164396851	10.9815614442	0.1019706623
	2048	243.765827548	96.542232909	55.322027310	**45.7164396686**	**10.9815614276**	**0.1019706457**
	32768	40.8397175049	1.6331043100	0.2361866110	45.7164396686	10.9815614276	0.1019706457
	131072	40.5972193392	1.3938439653	0.0000061447	45.7164396686	10.9815614276	0.1019706457
	262144	**40.5972193355**	**1.3938439616**	**0.0000061410**	45.7164396686	10.9815614276	0.1019706457
Ref. value		40.5972193355	1.3938439616	0.0000061410	45.7164396686	10.9815614276	0.1019706457

For the options at T= 1, all methods deliver full convergence except for the usual FFT biases. In contrast, for those at T= 0.1, the DPD and DPD-OPT also fail to provide full convergence:

1. The DPD and DPD-OPT require 256 sampling points to deliver a 10^{-10} accuracy in the T= 1 options (0.51 points per unit of w). Conversely, an upper limit w= 60000 does not provide full accuracy for the options at T= 0.1. The mispricings can be attributed to the higher truncation error of the DPD and DPD-OPT compared to other methods. Specifically, the DPD integration limit should be fur-

ther increased by a factor of 383[5] to eliminate the remaining $O(10^{-6})$ biases, evidencing a remarkably slow decay.

2. The AT-OPT achieves an accuracy of 10 decimal places for all strikes and maturities. However, it requires higher sampling densities than other implementations: in order to reach full convergence, the AT-OPT requires up to 2^{19} sampling points (8.74 points per unit of w) for the T= 0.1 options, the highest of all methods.

3. The FFT achieves full convergence for the two ATM strikes. However, it suffers from interpolation biases in ITM and OTM strikes[6]. The required sampling densities range from 4.10 to 4.37 points per w.

4. Finally, when all the options are specifically evaluated, both the FFT-SA and CM-OPT attain full accuracy for all configurations. The integration sizes are equivalent to those observed in the FFT.

Overall, after accounting for the expanded T= 0.1 integration domain, all variants produce accurate prices for the parameter set in Madan, Carr, and Chang (1998). However, the required sampling densities are notably different, ranging from 0.51 to 8.74 points per w.

Parameters based upon Itkin (2010): Analysis of two problematic cases

For our *second pricing* test we consider a parametrization S_0= 100, σ= 1, θ= 2, ν= 0.5 and r= 0.02. As in previous cases, we evaluate the accuracy at three strikes K = [60, 90, 140] and two tenors T= [0.1, 1]. Convergency problems surface immediately when trying to calculate the reference values. Despite substantially increasing the integration domain and sampling densities, we were unable to obtain concurrent AVG prices for any two Fourier-pricing methods. Furthermore, a simple inspection reveals that most pricing choices completely blow up under this parameter set. In particular, except for the AT-OPT, all methods produce negative call prices or unfeasible option values.

The problem, according to Itkin (2010), can be traced down to the inequality constraint

$$\frac{1}{\nu} > \theta + \frac{\sigma^2}{2} \tag{4.5}$$

which must be respected in order to obtain a valid risk-neutral measure. However, it is remarkable that, despite being in a region where (4.5) is not obeyed, the AT-OPT still delivers apparently feasible option prices. In contrast to other methods, the AT-OPT produces call values that are: (i) within reasonable positive bounds, (ii) monotonically increasing with time and (iii) monotonically decreasing across strikes. Table 6 shows the results[7].

5 Up to $w = 2.2*10^7$.

6 The FFT strike grid is centered on K= 101, thus exactly covering the near-ATM strike.

7 The FFT strike grid is centered on K= 90, thus exactly covering this strike.

AVG pricing results for a parameter set where inequality (4.5) is not respected. Integration performed over w = (0, 1200000]

TABLE 6

Method	N	$T=0.1$ $K=60$	$T=0.1$ $K=90$	$T=0.1$ $K=140$	$T=1$ $K=60$	$T=1$ $K=90$	$T=1$ $K=140$
DPD / DPD-OPT	2^{15}	5.5431*10^{22}	5.5431*10^{22}	5.5431*10^{22}	-6581.25996406	-6582.37082991	-6326.12578292
	2^{18}	6.9289*10^{21}	6.9289*10^{21}	6.9289*10^{21}	-804.637710681	-817.641676505	-807.052891605
	2^{20}	1.7322*10^{21}	1.7322*10^{21}	1.7322*10^{21}	-185.709482456	-199.985228368	-215.714705023
	2^{22}	4.3306*10^{20}	4.3306*10^{20}	4.3306*10^{20}	-31.0958006456	-44.8385968717	-65.5425150565
	2^{23}	2.1653*10^{20}	2.1653*10^{20}	2.1653*10^{20}	-12.3634864099	-25.5655486355	-46.0987178248
	2^{24}	1.0826*10^{20}	1.0826*10^{20}	1.0826*10^{20}	-10.5629165519	-23.6928307073	-44.2580733896
AT-OPT	2^{15}	-430.96286656	-484.17907474	-448.91667837	-189.31520068	-125.494168396	102.619046972
	2^{18}	7.3494999555	-12.466544758	-29.401466540	38.1053847408	33.2179061890	40.2902121315
	2^{20}	48.3693519584	31.2863798015	11.9329937296	62.4727400386	50.2258743878	33.6188355799
	2^{22}	50.9420851731	34.0217613501	14.4889004287	68.0466014576	54.2138578360	32.4154270823
	2^{23}	51.0312183887	34.1176540978	14.5742753420	68.4902856913	54.5295221850	32.3142801153
	2^{24}	51.0532777316	34.1413947115	14.5953765503	68.6036096897	54.6090386494	32.2846763861
FFT	2^{15}	48.5975777114	23.903138043	11.032066282	-0.2767*10^{-16}	-0.1361*10^{-16}	-0.0628*10^{-16}
	2^{18}	6.22093343383	3.0458963513	1.3295708551	0.0608*10^{-17}	0.6202*10^{-17}	-0.4205*10^{-17}
	2^{20}	2.16046286880	1.7659137695	0.8803717982	0.2908*10^{-6}	0.1149*10^{-6}	0.0263*10^{-6}
	2^{22}	-0.0914866217	0.7623279882	0.4766541273	-0.2880*10^{-4}	-0.1407*10^{-4}	-0.0634*10^{-4}
	2^{23}	-0.5460290452	0.5430298406	0.3776074074	-0.4065*10^{-4}	-0.2044*10^{-4}	-0.0956*10^{-4}
	2^{24}	-0.7810652966	0.4284830228	0.3252734458	-0.4405*10^{-4}	-0.2226*10^{-4}	-0.1048*10^{-4}
FFT-SA / CM-OPT	2^{15}	48.597577715	23.903138043	11.032025288	-0.2767*10^{-16}	-0.1361*10^{-16}	-0.0628*10^{-16}
	2^{18}	6.2209334343	3.0458963513	1.3295658553	0.0608*10^{-17}	0.6202*10^{-17}	-0.4205*10^{-17}
	2^{20}	2.1604628689	1.7659137695	0.8803684141	0.2908*10^{-6}	0.1149*10^{-6}	0.0263*10^{-6}
	2^{22}	-0.091486621	0.7623279882	0.4766525517	-0.2880*10^{-4}	-0.1407*10^{-4}	-0.0634*10^{-4}
	2^{23}	-0.546029045	0.5430298406	0.3776062103	-0.4065*10^{-4}	-0.2044*10^{-4}	-0.0956*10^{-4}
	2^{24}	-0.781065296	0.4284830228	0.3252724457	-0.4405*10^{-4}	-0.2226*10^{-4}	-0.1048*10^{-4}

Finally, we explore a *third* parametrization where S_0= 100, σ= 1, θ= 1.5, ν= 0.2 and r = 0.02, considering again the same strikes and maturities. This choice includes a problematic case reported in Itkin (2010), but in a region where inequality (4.5) is respected. In this region, AVG reference values can be obtained through the concurrent prices of the AT-OPT and DPD-OPT. Again, truncation biases are highest in the DPD-OPT and for the option at T= 0.1, which requires a range of w = (0, 1200000) to achieve an accuracy of 10^{-10}. Conversely, the AT-OPT delivers the same accuracy with a domain 17 times lower (i.e. with an upper limit w = 70000).

An outstanding result is the failure of both the FFT and the CM-OPT in this region. The pricing failure arises due to a singularity that appears after substituting the AVG characteristic function into Carr-Madan's integrand. This substitution generates a divergence that is not addressed by the $e^{\alpha \ln(K)}$ factor (Itkin, 2010). The FRFT in Chourdakis (2005), based on the same pricing equation, also fails to control this divergence.

Drilling down, we find that the blow-ups are connected to the specific values of the FFT dampening parameter α. As originally reported by Carr and Madan (1999), in order to keep the AVG characteristic function finite, the choice of α should respect

$$\alpha < \sqrt{\frac{\theta^2}{\sigma^4} + \frac{2}{\sigma^2 v}} - \frac{\theta}{\sigma^2} - 1 \tag{4.6}$$

thus requiring an α<1 in the *third* AVG parametrization. As a result, our initial choice α= 1.75 fails to provide reasonable prices. However, simply employing an α within the (0, 1) feasibility range can also generate substantial mispricings. In our tests, in order to achieve full accuracy, α must be specifically chosen between 0.35 and 0.55, thus further restricting the optimal α values. The pricing errors increase when using an α outside this optimal range and, even within the feasibility region, both the FFT and the CM-OPT completely blow up as α approaches either 0 or 1. In contrast, as Table 7 shows, neither the DPD nor the AT-OPT suffer this problem.

AVG pricing results for a parameter set where inequality (4.5) is respected. Shaded areas indicate an accuracy of 10^{-10}. Integration performed over w = (0, 1200000] TABLE 7

		$T=0.1$			$T=1$		
Method	N	$K=60$	$K=90$	$K=140$	$K=60$	$K=90$	$K=140$
DPD / DPD-OPT	2^{15}	270.683053581	160.312855480	237.86695077	1449.21661285	1496.35893184	1833.0259535
	2^{18}	54.0800276206	28.2704404214	15.914270370	199.206760987	192.219586279	212.743974011
	2^{20}	40.7020635729	20.1016563822	10.7855954314	74.6050157872	65.6279579754	56.2478146759
	2^{22}	40.5900314502	20.0293202567	10.7405868468	66.0965449591	58.9490639706	51.1509829305
	2^{23}	**40.5900314461**	**20.0293202541**	**10.7405868451**	**66.0965123856**	**58.9490408593**	**51.1509670470**
	2^{24}	40.5900314461	20.0293202541	10.7405868451	66.0965123856	58.9490408593	51.1509670470
AT-OPT	2^{15}	-421.09668122	-469.47887013	-425.90148223	-92.808830987	19.2653861422	3.27800576253
	2^{18}	8.11345483450	-13.012414180	-22.611830285	50.1646610093	51.3260989767	68.4322474755
	2^{20}	33.7963800774	13.2095933195	3.9033996984	60.9405228006	54.2608416231	49.7045688115
	2^{22}	40.5900314172	20.0293202251	10.7405868161	66.0965123565	58.9490408303	51.1509670179
	2^{23}	**40.5900314461**	**20.0293202541**	**10.7405868451**	**66.0965123856**	**58.9490408593**	**51.1509670470**
	2^{24}	40.5900314461	20.0293202541	10.7405868451	66.0965123856	58.9490408593	51.1509670470
FFT α=1.75	2^{15}	0,07843707418	-0.016599681	0,0151481402	-4,1151238493	-2.0240511287	-0,9341646344
	2^{18}	3,59480556391	-1.323163272	-0,2603100721	-0,5143834782	-0.2530040694	-0,1167728173
	2^{20}	12,9433291575	2.9380067936	0,5859217681	-0,1202676882	-0.0585836307	-0,0272927868
	2^{22}	14,7504640457	4.1059080343	1,2786976895	-0,0009051229	-0.0005506626	-0,3196390234
	2^{23}	14,8275732557	4.1548246008	1,3068472175	0.1568*10^{-8}	-0.1004*10^{-8}	-0.0617*10^{-8}
	2^{24}	14,8466387257	4.1669054042	1,3137851814	-0,2209*10^{-17}	0.3313*10^{-17}	0.1453*10^{-17}
FFT-SA / CM-OPT α=1.75	2^{15}	0.07843835368	-0.016599681	0.0151488067	-4.1151074745	-2.0240511287	-0.9341611630
	2^{18}	3.59478391991	-1.323163272	-0.260304099	-0.5143814312	-0.2530040694	-0.1167723833
	2^{20}	12.9432751140	2.9380067936	0.5859169265	-0.1202671977	-0.0585836307	-0.0272926881
	2^{22}	14.7504058715	4.1059080343	1.2786910516	-0.0009051203	-0.0005506626	-0.3196381856
	2^{23}	14.8275148996	4.1548246008	1.3068405019	-0.1568*10^{-8}	-0.1004*10^{-8}	-0.0617*10^{-8}
	2^{24}	14.8465803245	4.1669054042	1.3137784466	-0.4969*10^{-17}	0.3313*10^{-17}	0.1657*10^{-17}
FFT α=0.45	2^{15}	1304.40320341	1086.77815136	890.882030352	4973.89374494	4144.33632509	3397.08896528
	2^{18}	164.799488909	133.209481719	111.913110884	621.736726651	518.041782559	424.636263523
	2^{20}	50.4646458353	29.6599628754	20.1926560221	156.497151924	130.543150369	106.939721645
	2^{22}	40.5953058620	20.0344523771	10.7457395671	66.4468678658	59.1987747008	51.3240149513
	2^{23}	40.5901605037	20.0293205149	10.7406172558	66.0965743593	58.9490564174	51.1510145005
	2^{24}	40.5901602429	**20.0293202541**	10.7406169951	66.0965519079	**58.9490408593**	51.1510040436
FFT-SA / CM-OPT α=0.45	2^{15}	1304.40186937	20.0293202541	890.881180372	4973.88865557	4144.33632509	3397.08571924
	2^{18}	164.799293679	133.209481719	111.913018810	621.736090478	518.041782559	424.635857770
	2^{20}	50.4645153736	29.6599628754	20.1926251636	156.496993453	130.543150369	106.939618656
	2^{22}	40.5951770652	20.0344523771	10.7457094171	66.4468276776	59.1987747008	51.3239776506
	2^{23}	40.5900317067	20.0293205149	10.7405871058	66.0965348370	58.9490564174	51.1509775039
	2^{24}	**40.5900314461**	**20.0293202541**	**10.7405868451**	**66.0965123856**	**58.9490408593**	**51.1509670470**
Ref. value		40.5900314461	20.0293202541	10.7405868451	66.0965123856	58.9490408593	51.1509670470

Overall, our findings corroborate the problems described in Itkin (2010), extending the problematic cases to several strikes in addition to maturities. Furthermore, we show that the AT-OPT is the only method that doesn't blow up in any AVG problematic region, a result that to the best of our knowledge has not been reported before.

4.3.2 Computational speed in the AVG model

For the speed test we consider the Madan, Carr, and Chang (1998) parametrization, where all methods are blow-up free and can thus be compared on an equal basis. Since truncation errors are remarkably different depending on the option's expiry, we report the comparison for both $T = 1$ and $T = 0.1$. Tables 8 and 9 show the results.

CPU times required to achieve a 10^{-4} accuracy in the AVG model for $T = 1$ [milliseconds] TABLE 8

			N. of options priced					
Method	**W-range**	**Minimum N**	**1**	**10**	**25**	**100**	**500**	**2500**
DPD	(0, 462]	114	0.25333	2.53871	6.32706	25.31646	126.30225	632.12565
DPD-OPT	(0, 462]	114	0.25333	0.37439	0.49019	1.07109	3.34189	21.89656
AT-OPT	(0, 334]	773	0.47363	0.80103	1.30047	2.89957	15.84225	90.30666
FFT	(0, 295]	512	5.86752	5.86752	5.86752	5.86752	5.86752	320.7581
CM-OPT	(0, 295]	388	0.22453	0.34399	0.86634	1.71727	8.69480	47.84920

For the $T = 1$ options, the speed comparisons are similar to those of the Bates model: the DPD-OPT is the fastest method when pricing a high number of options, while the CM-OPT performs best for 10 or fewer options. In contrast, the FFT and the plain DPD are the slowest alternatives. Specifically, the AT-OPT, CM-OPT and DPD-OPT are, on average 5, 10 and 12 times faster than the FFT. These results characterize the CPU effort in an AVG region where no blow-ups or exploding truncation errors are observed.

Conversely, for the $T = 0.1$ options, the slow AVG hyperbolic decay significantly impacts the CPU speed. For the comparisons, due to exploding truncation errors, we first compute the integration range that delivers a 10^{-6} accuracy (instead of the usual 10^{-10}), and then obtain the number of sampling points that achieve an accuracy of 10^{-4}.

CPU times required to achieve a 10^{-4} accuracy in the AVG model for T = 0.1 [milliseconds]

TABLE 9

			N. of options priced					
Method	**W-range**	**Minimum N**	**1**	**10**	**25**	**100**	**500**	**2500**
DPD	(0, 79980]	14889	4.25683	45.74770	115.07808	461.01605	2304.7316	11533.798
DPD-OPT	(0, 79980]	14889	4.25683	13.2001	30.55344	126.34450	631.73191	3130.0113
AT-OPT	(0, 6997]	16041	5.56649	11.12723	19.95001	71.65330	347.83121	1785.9742
FFT	(0, 6536]	16384	5223.4364	5223.4364	5223.4364	5223.4364	5223.4364	5223.4364
CM-OPT	(0, 6536]	8625	1.57107	3.66537	9.42893	39.63055	205.72119	1024.4328

As Table 9 shows, due to the substantially expanded w-ranges, the computational times for the T= 0.1 options are, on average: (i) 59 times higher than in the T= 1 expiries, (ii) 40 times higher than in the Bates model and (iii) roughly 200 times higher than in the BSM model. When truncation errors play a prominent role, the faster decay of the CM-OPT, combined with its moderate sampling efficiency, allow this method to minimize the number of sampling points. In contrast, the sluggish decay of the DPD/DPD-OPT and the low sampling efficiency of the AT-OPT result in notably higher integration sizes. Therefore, the CM-OPT is the fastest method in all cases, whereas the AT-OPT and DPD-OPT rank as second best. These results demonstrate again the significantly higher CPU efficiency of strike vectorizations compared to classical alternatives like the FFT or the plain DPD.

5. Conclusions

This paper analyses the accuracy and speed of several Fourier-based implementation choices. In terms of pricing biases, we show that truncation errors increase as we move from the BSM to the Bates model and further intensify under the AVG dynamics. Discretization errors also increase when discontinuous jumps are considered, but the rise is modest and remains alike in both jump models. Across different methods: (i) the DPD and DPD-OPT exhibit the highest sampling efficiency but also the slowest decay rate, (ii) the CM-OPT stands out for minimizing truncation errors and (iii) the AT-OPT suffers from the highest discretization errors.

We show that obtaining accurate option values can be particularly challenging in the AVG model. While all methods show good convergence under the BSM and Bates dynamics, high truncation errors significantly complicate the practical AVG implementation. Moreover, depending on the AVG parameters, specific Fourier implementations may completely fail to provide reasonable option prices: both the FFT and the CM-OPT can blow up even in regions where inequality (4.5) is respected, whereas the DPD and DPD-OPT also fail when (4.5) is not obeyed. In contrast, the AT-OPT seems to work fine for any AVG parameter values.

Our speed analyses demonstrate the benefits of using strike vectorization compared to other choices. In our tests, computing option prices through the AT-OPT, CM-OPT and DPD-OPT is up to 78, 90 and 239 times faster than in the FFT. Overall, the DPD-OPT is the fastest alternative when pricing a high number of options, whereas the CM-OPT performs best when only a few prices are required.

Finally, the comparison between the FFT and the CM-OPT deserves a special mention. While both are based on the same pricing approach, the CM-OPT's flexibility allows (i) pricing any required strikes, (ii) choosing any integration size and technique and (iii) avoiding interpolation biases. As a result, the CM-OPT is both faster and more accurate than the FFT, thus rendering this method inefficient. Based on our results, we see no reason to employ the FFT over the CM-OPT, but further analysis may be needed in order to confirm this hypothesis.

6. References

Albrecher, Hansjörg, Philipp Mayer, Wim Schoutens, and Jurgen Tistaert, 2007, The little Heston trap, *Wilmott Magazine*, 83–92.

Attari, Mukarram, 2004, Option Pricing Using Fourier Transforms: A Numerically Efficient Simplification, *SSRN Electronic Journal.*

Bakshi, Gurdip, Charles Cao, and Zhiwu Chen, 1997, Empirical Performance of Alternative Option Pricing Models, *The Journal of Finance* 52, 2003–2049.

Bakshi, Gurdip, and Dilip B. Madan, 2000, Spanning and derivative-security valuation, *Journal of Financial Economics* 55, 205–238.

Bates, David S., 1996, Jumps and Stochastic Volatility: Exchange Rate Processes Implicit in Deutsche Mark Options, *Review of Financial Studies* 9, 69–107.

Black, Fischer, and Myron Scholes, 1973, The Pricing of Options and Corporate Liabilities, *Journal of Political Economy* 81, 637.

Broadie, Mark, and Ozgur Kaya, 2006, Exact Simulation of Stochastic Volatility and Other Affine Jump Diffusion Processes, *Operations Research* 54, 217–231.

Carr, Peter, and Dilip B. Madan, 1999, Option valuation using the fast Fourier transform, *Journal of Computational Finance* 3, 463–520.

Carr, Peter, and Liuren Wu, 2003, The finite moment logstable process and option pricing, *The Journal of Finance* 58, 753–778.

Chourdakis, Kyriakos, 2005, Option Pricing Using The Fractional FFT, *Journal of Computational Finance* 7, 1–23.

Cont, Rama, 2001, Empirical properties of asset returns: stylized facts and statistical issues, *Quantitative Finance* 1, 223–236.

Cont, Rama, and Peter Tankov, 2004, *Financial Modelling with Jump Processes* (Chapman & Hall/CRC).

Crisostomo, Ricardo, 2014, An Analysis of the Heston Stochastic Volatility Model: Implementation and Calibration using Matlab, *CNMV Working papers No. 58.*

Duffie, Darrell, Jun Pan, and Kenneth J Singleton, 2000, Transform Analysis and Asset Pricing for Affine Jump-Diffusions, *Econometrica* 68, 1343–1376.

Fang, F., and C. W. Oosterlee, 2009, A Novel Pricing Method for European Options Based on Fourier-Cosine Series Expansions, *SIAM Journal on Scientific Computing* 31, 826–848.

Fiorani, Filo, 2004, Option Pricing Under the Variance Gamma Process, *MPRA Paper No. 15395*.

Foresi, Silvio, and Liuren Wu, 2005, Crash-o-phobia: a domestic fear or a worldwide concern?, *The Journal of Derivatives* 13, 8–21.

Göncü, Ahmet, Mehmet O. Karahan, and Tolga U. Kuzubaş, 2016, A comparative goodness-of-fit analysis of distributions of some Lévy processes and Heston model to stock index returns, *North American Journal of Economics and Finance* 36, 69–83.

Heston, Steven, 1993, A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options, *The Review of Financial Studies* 6, 327–343.

Hirsa, Ali, 2012, *Computational Methods in Finance* (Chapman & Hall/CRC).

Itkin, Andrey, 2010, Pricing options with VG model using FFT, *arXiv working paper*.

Jones, Christopher S., 2003, The dynamics of stochastic volatility: Evidence from underlying and options markets, *Journal of Econometrics* 116, 181–224.

Kilin, Fiodar, 2011, Accelerating the Calibration of Stochastic Volatility Models, *The Journal of Derivatives* 18, 7–16.

Lee, Roger W., 2004, Option Pricing by Transform Methods: Extensions, Unification, and Error Control, *Journal of Computational Finance* 7, 51–86.

Lewis, Alan, 2001, A simple option formula for general jump-diffusion and other exponential Lévy processes, *OptionCity.net working paper*.

Lewis, Alan, 2013, Variance Gamma - different results, *Wilmott Forums*.

Lord, Roger, and Christian Kahl, 2010, Complex logarithms in Heston-like models, *Mathematical Finance* 20, 671–694.

Madan, Dilip B., Peter P. Carr, and Eric C. Chang, 1998, The Variance Gamma Process and Option Pricing, *Review of Finance* 2, 79–105.

Madan, Dilip B., and Eugene Seneta, 1990, The Variance Gamma (V.G.) Model for Share Market Returns, *The Journal of Business* 63, 511–524.

Matsuda, Kazuhisa, 2004, Introduction to Option Pricing with Fourier Transform: Option Pricing with Exponential Lévy Models, *Working Paper*.

Merton, Robert C., 1973, Theory of Rational Option Pricing, *Bell Journal of Economics* 4, 141–183.

Rebonato, Riccardo, 2004, *Volatility and Correlation: The Perfect Hedger and the Fox: Second Edition* (John Wiley & Sons, Inc).

Ribeiro, Claudia, and Nick Webber, 2004, Valuing path dependent options in the variance-gamma model by Monte Carlo with a gamma bridge, *Journal of Computational Finance* 7, 81–100.

Schmelzle, Martin, 2010, Option pricing formulae using Fourier transform: Theory and application, *Preprint, http://pfadintegral. com*.

Schobel, Rainer, and Jianwei Zhu, 1999, Stochastic Volatility With an Ornstein-Uhlenbeck Process: An Extension, *Review of Finance* 3, 23–46.

Schoutens, Wim, Erwin Simons, and Jurgen Tistaert, 2004, A Perfect Calibration! Now What?, *Willmott Magazine*, 66–78.

Sepp, Artur, 2003, Fourier Transform for Option Pricing under Affine Jump-Diffusions: An Overview, *SSRN Electronic Journal*.

Zhu, Jianwei, 2010, *Applications of Fourier Transform to Smile Modeling: Theory and Implementation* (Springer Finance).