A Fourier Transform Method for Spread Option Pricing[*]

T. R. Hurd[†] and Zhuowei Zhou[†]

Abstract. Spread options are a fundamental class of derivative contracts written on multiple assets and are widely traded in a range of financial markets. There is a long history of approximation methods for computing such products, but as yet there is no preferred approach that is accurate, efficient, and flexible enough to apply in general asset models. The present paper introduces a new formula for general spread option pricing based on Fourier analysis of the payoff function. Our detailed investigation, including a flexible and general error analysis, proves the effectiveness of a fast Fourier transform implementation of this formula for the computation of spread option prices. It is found to be easy to implement, stable, efficient, and applicable in a wide variety of asset pricing models.

Key words. spread options, multivariate spread options, jump diffusions, fast Fourier transform, gamma function

AMS subject classifications. 33B15, 65T50, 91B28

DOI. 10.1137/090750421

1. Introduction. When S_{jt}, $j = 1, 2$, $t \geq 0$, are two asset price processes, the basic spread option with maturity T and strike $K \geq 0$ is the contract that pays $(S_{1T} - S_{2T} - K)^+$ at time T. If we assume the existence of a risk-neutral pricing measure, the risk-neutral expectation formula for the time 0 price of this option, assuming a constant interest rate r, is

$$\text{(1.1)} \qquad \text{Spr}(S_0; T, K) = e^{-rT}\mathbb{E}_{S_0}[(S_{1T} - S_{2T} - K)^+].$$

The literature on applications of spread options is extensive and is reviewed by Carmona and Durrleman [2], who explore further applications of spread options beyond the case of equities modeled by geometric Brownian motion (GBM), in particular to energy trading. For example, the difference between the price of crude oil and a refined fuel such as natural gas is called a "crack spread." "Spark spreads" refer to differences between the price of electricity and the price of fuel: options on spark spreads are widely used by power plant operators to optimize their revenue streams. Energy pricing requires models with mean reversion and jumps very different from GBM, and pricing spread options in such situations can be challenging.

Closed formulas for (1.1) are known only for a limited set of asset models. In the Bachelier stock model, $S_t = (S_{1t}, S_{2t})$ is an arithmetic Brownian motion, and in this case (1.1) has a Black–Scholes-type formula for any T, K. In the special case $K = 0$ when S_t is a GBM, (1.1) is given by the Margrabe formula [14].

[†]Department of Mathematics and Statistics, McMaster University, Hamilton, ON, L8S 4K1, Canada (hurdt@mcmaster.ca, zhouz3@univmail.cis.mcmaster.ca).

In the basic case where S_t is a GBM and $K > 0$, no explicit pricing formula is known. Instead, there is a long history of approximation methods for this problem. Numerical integration methods, typically Monte Carlo based, are often employed. When possible, however, the fastest option pricing engines by numerical integration are usually those based on the fast Fourier transform (FFT) methods introduced by Carr and Madan [4]. Their first interest was in single asset option pricing for geometric Lévy process models like the variance gamma (VG) model, but their basic framework has since been adapted to a variety of option payoffs and a host of asset return models where the characteristic function is known. In this work, when the payoff function is not square integrable, it is important to account for singularities in the Fourier transform variables.

Dempster and Hong [5] introduced a numerical integration method for spread options based on two-dimensional FFTs that was shown to be efficient when the asset price processes are GBMs or to have stochastic volatility. Three more recent papers study the use of multi-dimensional convolution FFT methods to price a wide range of multiasset options, including basket and spread options. These newer methods also compute by discretized Fourier transforms over truncated domains, but unlike earlier work using the FFT, they apparently do not rely on knowing the analytic Fourier transform of the payoff function or integrability of the payoff function. Lord et al. [11] provide error analysis that explains their observation that errors decay as a negative power of the size N of the grid used in computing the FFT, provided the truncation is taken large enough. Leentvaar and Oosterlee [9] propose a parallel partitioning approach to tackle the so-called curse of dimensionality when the number of underlying assets becomes large. Jackson, Jaimungal, and Surkov [6] proposed a general FFT pricing framework for multiasset options, including variations with Bermudan early exercise features. These three papers all find that the FFT applied to the payoff function can perform well even if the payoff function is not square integrable and observe that errors can be made to decay as a negative power of N.

As an alternative to numerical integration methods, another stream uses analytical methods applicable to log-normal models that involve linear approximations of the nonlinear exercise boundary. Such methods are often very fast, but their accuracy is usually not easy to determine. Kirk [7] presented an analytical approximation that performs well in practice. Carmona and Durrleman [3] and later Li, Deng, and Zhou [10] demonstrate a number of lower and upper bounds for the spread option price that combine to produce accurate analytical approximation formulas in log-normal asset models. These results extend to approximate values for the Greeks.

The main purpose of the present paper is to give a numerical integration method for computing spread options in two or higher dimensions using the FFT. Unlike the above multiasset FFT methods, it is based on square integrable integral formulas for the payoff function, and like those methods it is applicable to a variety of spread option payoffs in any model for which the characteristic function of the joint return process is given analytically. Since our method involves only smooth square integrable integrands, the error estimates we present are quite straightforward and standard. In fact, we demonstrate that the asymptotic decay of errors is exponential, rather than polynomial, in the size N of the Fourier grid. For option payoffs that can be made square integrable, our method has the flexibility to handle a wide range of desirable asset return models, all with a very competitive computational expense.

The results we describe stem from the following new formula,[1] which gives a square integrable Fourier representation of the basic spread option payoff function $P(x_1, x_2) = (e^{x_1} - e^{x_2} - 1)^+$.

Theorem 1.1. *For any real numbers $\epsilon = (\epsilon_1, \epsilon_2)$ with $\epsilon_2 > 0$ and $\epsilon_1 + \epsilon_2 < -1$ and $x = (x_1, x_2)$,[2]*

$$(1.2) \qquad P(x) = (2\pi)^{-2} \iint_{\mathbb{R}^2 + i\epsilon} e^{iux'} \hat{P}(u) d^2 u, \quad \hat{P}(u) = \frac{\Gamma(i(u_1 + u_2) - 1)\Gamma(-iu_2)}{\Gamma(iu_1 + 1)}.$$

Here $\Gamma(z)$ is the complex gamma function defined for $\Re e(z) > 0$ by the integral $\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt$.

Using this theorem, whose proof is given in the appendix, we will find that we can follow the logic of Carr and Madan to derive numerical algorithms for efficient computation of a variety of spread options and their Greeks. The basic strategy to compute (1.1) is to combine (1.2) with an explicit formula for the characteristic function of the bivariate random variable $X_t = (\log S_{1t}, \log S_{2t})$. For the remainder of this paper, we make a simplifying assumption.

Assumption 1. For any $t > 0$, the increment $X_t - X_0$ is independent of X_0.

This implies that the characteristic function of X_T factorizes

$$(1.3) \qquad \mathbb{E}_{X_0}[e^{iuX_T'}] = e^{iuX_0'} \Phi(u; T), \quad \Phi(u; T) := \mathbb{E}_{X_0}[e^{iu(X_T - X_0)'}],$$

where $\Phi(u; T)$ is independent of X_0. Although the above assumption rules out mean-reverting processes that often arise in energy applications, it holds for typical stock models: moreover, the method we propose can be generalized to a variety of mean-reverting processes. Using Theorem 1.1 and (1.3), the spread option formula can be written as an explicit two-dimensional Fourier transform in the variable X_0:

$$
\begin{aligned}
\mathrm{Spr}(X_0; T) &= e^{-rT} \mathbb{E}_{X_0}[(e^{X_{1T}} - e^{X_{2T}} - 1)^+] \\
&= e^{-rT} \mathbb{E}_{X_0}\left[(2\pi)^{-2} \iint_{\mathbb{R}^2 + i\epsilon} e^{iuX_T'} \hat{P}(u) d^2 u\right] \\
&= (2\pi)^{-2} e^{-rT} \iint_{\mathbb{R}^2 + i\epsilon} \mathbb{E}_{X_0}[e^{iuX_T'}] \hat{P}(u) d^2 u \\
(1.4) \qquad &= (2\pi)^{-2} e^{-rT} \iint_{\mathbb{R}^2 + i\epsilon} e^{iuX_0'} \Phi(u; T) \hat{P}(u) d^2 u.
\end{aligned}
$$

The Greeks are handled in exactly the same way. For example, the Delta $\Delta^1 := \partial \mathrm{Spr}/\partial S_{10}$ is obtained as a function of S_0 by replacing Φ in (1.4) by $\partial \Phi/\partial S_{10}$.

Double Fourier integrals like this can be approximated numerically by a two-dimensional FFT. Such approximations involve both a truncation and discretization of the integral, and the two properties that determine their accuracy are the decay of the integrand of (1.4) in u-space

---

[1]It came to our attention after the submission of our paper that the result of this theorem has been simultaneously and independently stated in another working paper by Antonov and Arneguy [1].

[2]Here and in rest of the paper, some variables such as u, ϵ, x are defined to be row vectors with components $u = (u_1, u_2)$, etc. We use implied matrix multiplication so that $ux' = u_1 x_1 + u_2 x_2$, where x' denotes the (unconjugated) transpose of x.

and the decay of the function Spr in x-space. The remaining issue of computing the gamma function is not really difficult. Fast and accurate computation of the complex gamma function in, for example, MATLAB, is based on the Lanczos approximation popularized by [15].[3]

In this paper, we demonstrate how our method performs for computing spread options in three different two-asset stock models, namely GBM, a three factor stochastic volatility (SV) model, and the VG model. Section 2 provides the essential definitions of the three types of asset return models, including explicit formulas for their bivariate characteristic functions. Section 3 discusses how the two-dimensional FFT can be implemented for our problem. Section 4 provides error analysis that shows how the accuracy and speed will depend on the implementation choices made. Section 5 describes briefly how the method extends to the computation of spread option Greeks. Section 6 gives the detailed results of the performance of the method in the three asset return models. In this section, the accuracy of each model is compared to benchmark values computed by an independent method for a reference set of option prices. We also demonstrate that the computation of the spread option Greeks in such models is equally feasible. Section 7 extends all the above results to several kinds of basket options on two or more assets. Although the formulation is simple, the resulting FFTs become, in practice, much slower to compute in higher dimensions, due to the so-called curse of dimensionality: in such cases, one can implement the parallel partitioning approach of [9].

2. Three kinds of stock models.

2.1. The case of GBM. In the two-asset Black–Scholes model, the vector $S_t = (S_{1t}, S_{2t})$ has components

$$S_{jt} = S_{j0} \exp[(r - \sigma_j^2/2)t + \sigma_j W_t^j], \quad j = 1, 2,$$

where $\sigma_1, \sigma_2 > 0$ and W^1, W^2 are risk-neutral Brownian motions with constant correlation $\rho, |\rho| < 1$. The joint characteristic function of $X_T = (\log S_{1T}, \log S_{2T})$ as a function of $u = (u_1, u_2)$ is of the form $e^{iuX_0'}\Phi(u; T)$ with

$$(2.1) \qquad \Phi(u; T) = \exp[iu(rTe - \sigma^2 T/2)' - u\Sigma u' T/2],$$

where $e = (1, 1)$, $\Sigma = [\sigma_1^2, \sigma_1\sigma_2\rho; \sigma_1\sigma_2\rho, \sigma_2^2]$, and $\sigma^2 = \operatorname{diag} \Sigma$. We remind the reader that we use implied matrix multiplication and that u' denotes the (unconjugated) matrix transpose. Substituting this expression into (1.4) yields the spread option formula

$$(2.2) \quad \operatorname{Spr}(X_0; T) = (2\pi)^{-2} e^{-rT} \iint_{\mathbb{R}^2 + i\epsilon} e^{iuX_0'} \exp[iu(rTe - \sigma^2 T/2)' - u\Sigma u' T/2] \hat{P}(u) d^2 u.$$

As we discuss in section 3, we recommend that this be computed numerically using the FFT.

2.2. Three factor SV model. The spread option problem in a three factor stochastic volatility model was given as an example by Dempster and Hong [5]. Their asset model is

---

[3] According to these authors, computing the gamma function becomes "not much more difficult than other built-in functions that we take for granted, such as $\sin x$ or e^x."

defined by SDEs for $X_t = (\log S_{1t}, \log S_{2t})$ and the squared volatility v_t:

$$dX_1 = [(r - \delta_1 - \sigma_1^2/2)dt + \sigma_1\sqrt{v}dW^1],$$
$$dX_2 = [(r - \delta_2 - \sigma_2^2/2)dt + \sigma_2\sqrt{v}dW^2],$$
$$dv = \kappa(\mu - v)dt + \sigma_v\sqrt{v}dW^v,$$

where the three Brownian motions have correlations:

$$E[dW^1 dW^2] = \rho dt,$$
$$E[dW^1 dW^v] = \rho_1 dt,$$
$$E[dW^2 dW^v] = \rho_2 dt.$$

As discussed in that paper, the asset return vector has the joint characteristic function $e^{iuX_0'}\Phi(u; T, v_0)$, where

$$\Phi(u; T, v_0) = \left[\left(\frac{2\zeta(1 - e^{-\theta T})}{2\theta - (\theta - \gamma)(1 - e^{-\theta T})} \right) v_0 \right.$$
$$\left. + iu(re - \delta)'T - \frac{\kappa\mu}{\sigma_v^2} \left[2\log\left(\frac{2\theta - (\theta - \gamma)(1 - e^{-\theta T})}{2\theta} \right) + (\theta - \gamma)T \right] \right]$$

and

$$\zeta := -\frac{1}{2}\left[(\sigma_1^2 u_1^2 + \sigma_2^2 u_2^2 + 2\rho\sigma_1\sigma_2 u_1 u_2) + i\left(\sigma_1^2 u_1 + \sigma_2^2 u_2 \right) \right],$$
$$\gamma := \kappa - i(\rho_1\sigma_1 u_1 + \rho_2\sigma_2 u_2)\sigma_\nu,$$
$$\theta := \sqrt{\gamma^2 - 2\sigma_v^2\zeta}.$$

2.3. Exponential Lévy models. Many stock price models are of the form $S_t = e^{X_t}$, where X_t is a Lévy process for which the characteristic function is explicitly known. We illustrate with the example of the VG process introduced by [13] the three parameter process Y_t with Lévy characteristic triple $(0, 0, \nu)$, where the Lévy measure is $\nu(x) = \lambda[e^{-a_+ + x}\mathbf{1}_{x>0} + e^{a_- - x}\mathbf{1}_{x<0}]/|x|$ for positive constants $\lambda, a_\pm$. The characteristic function of Y_t is

$$(2.3) \qquad \Phi_{Y_t}(u) = \left[1 + i\left(\frac{1}{a_-} - \frac{1}{a_+} \right)u + \frac{u^2}{a_- a_+} \right]^{-\lambda t}.$$

To demonstrate the effects of correlation, we take a bivariate VG model driven by three independent VG processes Y_1, Y_2, Y with common parameters $a_\pm$ and $\lambda_1 = \lambda_2 = (1 - \alpha)\lambda$, $\lambda^Y = \alpha\lambda$. The bivariate log return process $X_t = \log S_t$ is a mixture:

$$(2.4) \qquad X_{1t} = X_{10} + Y_{1t} + Y_t, \quad X_{2t} = X_{20} + Y_{2t} + Y_t.$$

Here $\alpha \in [0, 1]$ leads to dependence between the two return processes but leaves their marginal laws unchanged. An easy calculation leads to the bivariate characteristic function $e^{iuX_0'}\Phi(u; T)$

with

$$(2.5) \quad \Phi(u; T) = \left[1 + i\left(\frac{1}{a_-} - \frac{1}{a_+}\right)(u_1 + u_2) + \frac{(u_1 + u_2)^2}{a_- a_+}\right]^{-\alpha\lambda t}$$

$$\times \left[1 + i\left(\frac{1}{a_-} - \frac{1}{a_+}\right)u_1 + \frac{u_1^2}{a_- a_+}\right]^{-(1-\alpha)\lambda t} \left[1 + i\left(\frac{1}{a_-} - \frac{1}{a_+}\right)u_2 + \frac{u_2^2}{a_- a_+}\right]^{-(1-\alpha)\lambda t}.$$

3. Numerical integration by FFT. To compute (1.4) in these models we approximate the double integral by a double sum over the lattice

$$\Gamma = \{u(k) = (u_1(k_1), u_2(k_2)) \mid k = (k_1, k_2) \in \{0, \ldots, N-1\}^2\}, \quad u_i(k_i) = -\bar{u} + k_i \eta$$

for appropriate choices of $N, \eta, \bar{u} := N\eta/2$. For the FFT it is convenient to take N to be a power of 2 and lattice spacing η such that truncation of the u-integrals to $[-\bar{u}, \bar{u}]$ and discretization leads to an acceptable error. Finally, we choose initial values $X_0 = \log S_0$ to lie on the reciprocal lattice with spacing $\eta^* = 2\pi/N\eta = \pi/\bar{u}$ and width $2\bar{x}$, $\bar{x} = N\eta^*/2$:

$$\Gamma^* = \{x(\ell) = (x_1(\ell_1), x_2(\ell_2)) \mid \ell = (\ell_1, \ell_2) \in \{0, \ldots, N-1\}^2\}, \ x_i(\ell_i) = -\bar{x} + \ell_i \eta^*.$$

For any $S_0 = e^{X_0}$ with $X_0 = x(\ell) \in \Gamma^*$ we then have the approximation

$$(3.1) \qquad \mathrm{Spr}(X_0; T) \sim \frac{\eta^2 e^{-rT}}{(2\pi)^2} \sum_{k_1, k_2 = 0}^{N-1} e^{i(u(k) + i\epsilon)x(\ell)'} \Phi(u(k) + i\epsilon; T)\hat{P}(u(k) + i\epsilon).$$

Now, as usual for the discrete FFT, as long as N is even,

$$iu(k)x(\ell)' = i\pi(k_1 + k_2 + \ell_1 + \ell_2) + 2\pi i k\ell'/N \quad (\mathrm{mod} \ 2\pi i).$$

This leads to the double inverse discrete Fourier transform (i.e., the MATLAB function ifft2)

$$\mathrm{Spr}(X_0; T) \sim (-1)^{\ell_1 + \ell_2} e^{-rT} \left(\frac{\eta N}{2\pi}\right)^2 e^{-\epsilon x(\ell)'} \left[\frac{1}{N^2} \sum_{k_1, k_2 = 0}^{N-1} e^{2\pi i k\ell'/N} H(k)\right]$$

$$(3.2) \qquad\qquad = (-1)^{\ell_1 + \ell_2} e^{-rT} \left(\frac{\eta N}{2\pi}\right)^2 e^{-\epsilon x(\ell)'} [\mathrm{ifft2}(H)](\ell),$$

where

$$H(k) = (-1)^{k_1 + k_2} \Phi(u(k) + i\epsilon; T)\hat{P}(u(k) + i\epsilon).$$

4. Error discussion. The selection of suitable values for ϵ, N, and η when implementing the above FFT approximation of (1.4) is a somewhat subtle issue whose details depend on the asset model in question. We now give a general discussion of the pure truncation error and pure discretization error in (3.1): a more complete analysis of the combined errors using methods described in [8] will lead to the same broad conclusions.

The pure truncation error, defined by taking $\eta \to 0$, $N \to \infty$ while keeping $\bar{u} = N\eta/2$ fixed, can be made smaller than $\delta_1 \ll 1$ if the integrand of (1.4) is small and decaying outside

the square $[-\bar{u} + i\epsilon_1,\ \bar{u} + i\epsilon_1] \times [-\bar{u} + i\epsilon_2,\ \bar{u} + i\epsilon_2]$. Corollary A.1, proved in the appendix, gives a uniform $O(|u|^{-2})$ upper bound on $\hat{P}$, while $\Phi(u)$ can generally be seen directly to have some u-decay. Thus the truncation error will be less than δ_1 if one picks $\bar{u}$ large enough so that $|\Phi| < O(\delta_1)$ and has decay outside the square.

The pure discretization error, defined by taking $\bar{u} \to \infty$, $N \to \infty$ while keeping $\bar{x} = \pi/\eta$ fixed, can be made smaller than $\delta_2 \ll 1$ if $e^{\epsilon X_0'}\mathrm{Spr}(X_0)$, taken as a function of $X_0 \in \mathbb{R}^2$, has rapid decay in X_0. This is related to the smoothness of $\Phi(u)$ and the choice of ϵ. The first two models are not very sensitive to ϵ, but in the VG model the following conditions are needed to ensure that singularities in u-space are avoided:

$$-a_+ < \epsilon_1, \epsilon_2, \epsilon_1 + \epsilon_2 < a_-.$$

By applying the Poisson summation formula to $e^{\epsilon X_0'}\mathrm{Spr}(X_0)$, one can write the discretization error as

$$(4.1) \qquad \mathrm{Spr}^{(\bar{x})}(X_0) - \mathrm{Spr}(X_0) = \sum_{\ell \in \mathbb{Z}^2 \setminus \{(0,0)\}} e^{2\bar{x}\epsilon\ell'}\mathrm{Spr}(X_0 + 2\bar{x}\ell).$$

One can verify using brute force bounds that the terms on the right-hand side of (4.1) are all small and decay in all lattice directions, provided $\bar{x}$ is sufficiently large. Thus the discretization error will be less than δ_2 for all $X_0 \in [-c\bar{x}, c\bar{x}]^2$ with $0 < c \ll 1$ if one picks $\bar{x}$ large enough so that $|e^{\epsilon X_0'}\mathrm{Spr}(X_0)| < O(\delta_2)$ and has decay outside the square $[-\bar{x}, \bar{x}]^2$.

In summary, one expects that the combined truncation and discretization error will be close to $\delta_1 + \delta_2$ if $\bar{u} = N\eta/2$ and $\eta = \pi/\bar{x}$ are each chosen as above. We shall see in section 6 that the observed errors are consistent with the above analysis that predicts an asymptotic exponential decay with the size N of the Fourier lattice for the models we address.

5. Greeks. The FFT method can also be applied to the Greeks, enabling us to tackle hedging and other interesting problems. It is particularly efficient for the GBM model, where differentiation under the integral sign is always permissible. For instance, the FFT formula for vega (the sensitivity to σ) takes the form

$$\frac{\partial \mathrm{Spr}(S_0; T)}{\partial \sigma_1} = (-1)^{\ell_1 + \ell_2} e^{-rT} \left(\frac{\eta N}{2\pi}\right)^2 e^{-\epsilon x(\ell)'} \left[\mathrm{ifft2}\left(\frac{\partial H}{\partial \sigma_1}\right)\right](\ell),$$

$$\frac{\partial H(k)}{\partial \sigma_1} = \left[-(u(k) + i\epsilon)\left(i\frac{\partial \sigma^2{}'}{\partial \sigma_1} + \frac{\partial \Sigma}{\partial \sigma_1}(u(k) + i\epsilon)'\right)\frac{T}{2}\right]H(k),$$

where $\frac{\partial \sigma^2}{\partial \sigma_1} = [2\sigma_1, 0]$ and $\frac{\partial \Sigma}{\partial \sigma_1} = [2\sigma_1, \rho\sigma_2; \rho\sigma_2, 0]$. Other Greeks including those of higher orders can be computed in a similar fashion. This method needs to be used with care for the SV and VG models, since it is possible that differentiation leads to an integrand that decays slowly.

6. Numerical results. Our numerical experiments were coded and implemented in MATLAB version 7.6.0 on an Intel 2.80 GHz machine running under Linux with 1 GB physical memory. If they were coded in C++ with similar algorithms, we should expect to see faster performance.
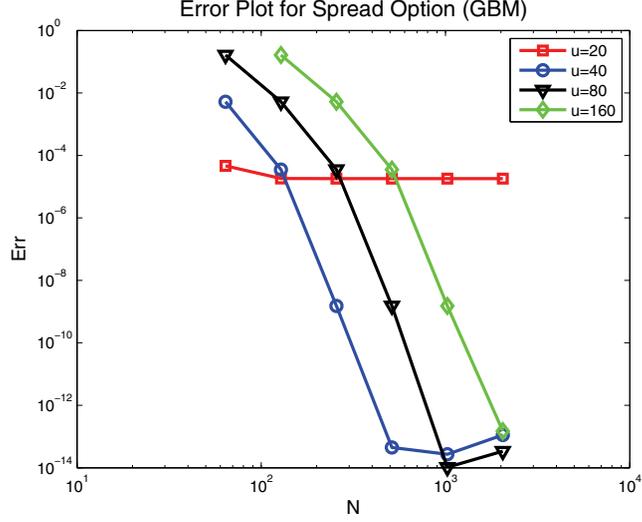
Figure 1. *This graph shows the objective function* Err *for the numerical computation of the GBM spread option versus the benchmark. Errors are plotted against the grid size for different choices of $\bar{u}$. The parameter values are those of the GBM model used by* [5]: $r = 0.1$, $T = 1.0$, $\rho = 0.5$, $\delta_1 = 0.05$, $\sigma_1 = 0.2$, $\delta_2 = 0.05$, $\sigma_2 = 0.1$.

The strength of the FFT method is demonstrated by comparison with accurate benchmark prices computed by an independent (usually extremely slow) method. Based on a representative selection of initial log-asset value pairs $\log S_{10}^i = \frac{i\pi}{10}$, $\log S_{20}^j = -\frac{\pi}{5} + \frac{j\pi}{10}$, $i, j \in 1, 2, 3, \ldots, 6$, the objective function we measure is defined as

$$(6.1) \qquad\qquad \mathrm{Err} = \frac{1}{36} \sum_{i,j=1}^{6} |\log(M^{ij}) - \log(B^{ij})|,$$

where M^{ij} and B^{ij} are the corresponding FFT computed prices and benchmark prices. These choices cover a wide range of moneyness, from deep out-of-the-money to deep in-the-money. Since these combinations all lie on lattices Γ^* corresponding to $N = 2^n$ and $\bar{u}/10 = 2^m$ for integers n, m, all 36 prices M^{ij} can be computed simultaneously with a single FFT.

Figure 1 shows how the FFT method performs in the two-dimensional GBM model for different choices of N and $\bar{u}$. Since the two factors are bivariate normal, benchmark prices can be calculated to high accuracy by one-dimensional integrations. In Figure 1 we can clearly see the effects of both truncation errors and discretization errors. For a fixed $\bar{u}$, the objective function decreases when N increases. The $\bar{u} = 20$ curve flattens out near 10^{-5} due to its truncation error of that magnitude. In turn, we can quantify its discretization errors with respect to N by subtracting the truncation error from the total error. The flattening of the curves with $\bar{u} = 40$, 80, and 160 near 10^{-14} should be attributed to MATLAB round-off errors: because of the rapid decrease of the characteristic function Φ, their truncation error is negligible. For a fixed N, increasing $\bar{u}$ brings two effects: reducing truncation error and enlarging discretization error. These effects are well demonstrated in Figure 1.
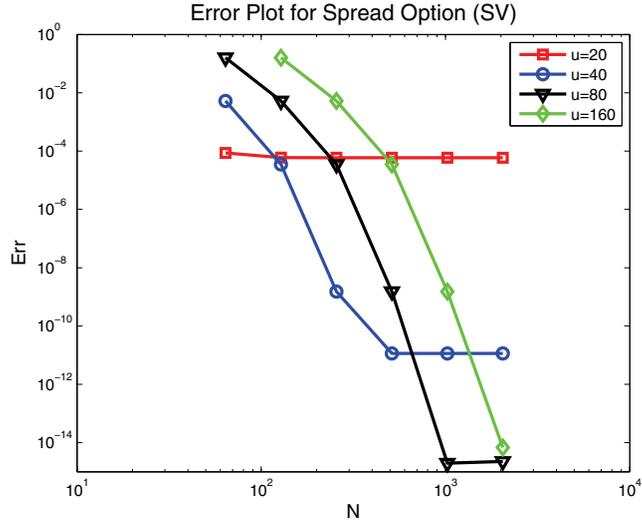
Figure 2. *This graph shows the objective function* Err *for the numerical computation of the SV spread option versus the benchmark computed using the FFT method itself with parameters $N = 2^{12}$ and $\bar{u} = 80$. The parameter values are those of the SV model used by* [5]: $r = 0.1$, $T = 1.0$, $\rho = 0.5$, $\delta_1 = 0.05$, $\sigma_1 = 1.0$, $\rho_1 = -0.5$, $\delta_2 = 0.05$, $\sigma_2 = 0.5$, $\rho_2 = 0.25$, $v_0 = 0.04$, $\kappa = 1.0$, $\mu = 0.04$, $\sigma_v = 0.05$.

For the SV model, no analytical or numerical method we know is consistently accurate enough to serve as an independent benchmark. Instead, we computed benchmark prices using the FFT method itself with $N = 2^{12}$ and $\bar{u} = 80$. The resulting objective function shows similar behavior to Figure 1 and is consistent with accuracies at the level of roundoff. We also verified that the benchmark prices are consistent to a level of 4×10^{-4} with those resulting from an intensive Monte Carlo computation using 1,000,000 simulations, each consisting of 2000 time steps. The computational cost to further reduce the Monte Carlo simulation error becomes prohibitive.

Because the VG process has an explicit probability density function in terms of a Bessel function [12], rather accurate benchmark spread option values for the VG model can be computed by a three-dimensional integration.[4] We used a Gaussian quadrature algorithm set with a high tolerance of 10^{-9} to compute the integrals for these benchmarks. The resulting objective function for various values of $\bar{u}, N$ is shown in Figure 3. The truncation error for $\bar{u} = 20$ is about 2×10^{-5}. The other three curves flatten out near 5×10^{-8}, a level we identify as the accuracy of the benchmark. A comparable graph (not shown), using benchmark prices computed with the FFT method with $N = 2^{12}$ and $\bar{u} = 80$, showed behavior similar to Figures 1 and 2 and is consistent with the FFT method being capable of producing accuracies at the level of roundoff.

The strength of the FFT method is further illustrated by the computation of individual prices and relative errors shown in Tables 1, 2, and 3. One can observe that an FFT with $N = 256$ is capable of producing very high accuracy in all three models. It is interesting to note that FFT prices in almost all cases were biased low compared to the benchmark. Exceptions

---

[4]We thank a referee for this suggestion.
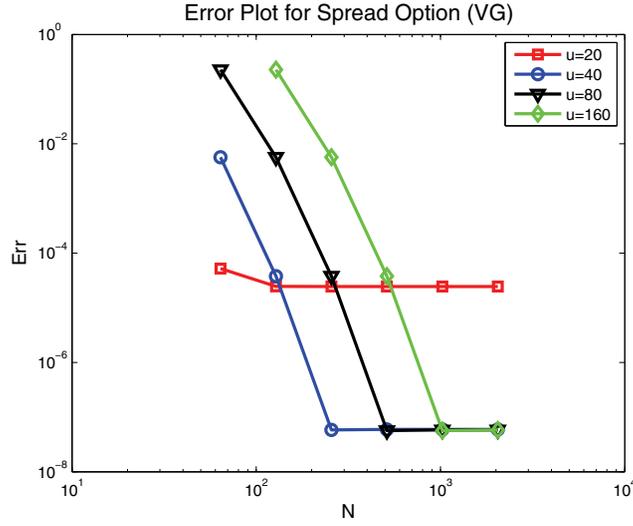
Figure 3. *This graph shows the objective function* Err *for the numerical computation of the VG spread option versus the benchmark values computed with a three-dimensional integration. Errors are plotted against the grid size for five different choices of* $\bar{u}$. *The parameters are* $r = 0.1$, $T = 1.0$, $\rho = 0.5$, $a_+ = 20.4499$, $a_- = 24.4499$, $\alpha = 0.4$, $\lambda = 10$.

Table 1

Benchmark prices for the two-factor GBM model of [5] *and relative errors for the FFT method with different choices of N. The parameter values are the same as Figure 1 except we fix* $S_{10} = 100$, $S_{20} = 96$, $\bar{u} = 40$. *The interpolation is based on a matrix of prices with discretization of* $N = 256$ *and a polynomial with a degree of 8.*

Strike K	Benchmark	64	128	256	512	Interpolation
0.4	8.312461	-3.8	-4.5E-4	-1.9E-8	-1.7E-14	1.9E-8
0.8	8.114994	-3.8E-1	-4.6E-4	-2.0E-8	-7E-15	2.0E-8
1.2	7.920820	-7.3E-2	-4.6E-4	-2.0E-8	-2.8E-14	2.0E-8
1.6	7.729932	-7.2E-2	-4.7E-4	-2.0E-8	-4.8E-14	2.0E-8
2.0	7.542324	-7.3E-2	-4.8E-4	-2.1E-8	-4.9E-14	2.1E-8
2.4	7.357984	-7.5E-2	-4.9E-4	-2.1E-8	-7.3E-14	2.1E-8
2.8	7.176902	-7.6E-2	-5.0E-4	-2.2E-8	-6.8E-14	2.2E-8
3.2	6.999065	-7.8E-2	-5.1E-4	-2.2E-8	-9.7E-14	2.2E-8
3.6	6.824458	-8.0E-2	-5.3E-4	-2.3E-8	-8.2E-14	2.3E-8
4.0	6.653065	-8.1E-2	-5.4E-4	-2.3E-8	-9.0E-14	2.3E-8

to this observation seem only to appear at a level of the accuracy of the benchmark itself.

The FFT computes in a single iteration an $N \times N$ panel of prices *spread* corresponding to initial values $S_{10} = e^{x_{10} + \ell_1 \eta^*}$, $S_{20} = e^{x_{20} + \ell_2 \eta^*}$, $K = 1$, $(\ell_1, \ell_2) \in \{0, \ldots, N-1\}^2$. If the desired selection of $\{S_{10}, S_{20}, K\}$ fits into this panel of prices, or its scaling, a single FFT suffices. If not, then one has to match (x_{10}, x_{20}) with each combination, and run several FFTs, with a consequent increase in computation time. However, we have found that an interpolation technique is very accurate for practical purposes. For instance, prices for multiple strikes with the same S_{10} and S_{20} are approximated by a polynomial fit along the diagonal

Table 2

Benchmark prices for the three factor SV model of [5] *and relative errors for the FFT method with different choices of N. The parameter values are the same as Figure 2 except we fix $S_{10} = 100$, $S_{20} = 96$, $\bar{u} = 40$. The interpolation is based on a matrix of prices with discretization of $N = 256$ and a polynomial with a degree of 8.*

Strike K	Benchmark	64	128	256	512	Interpolation
2.0	7.548502	−7.3E-2	−4.8E-4	−2.1E-8	1.6E-11	−2.1E-8
2.2	7.453536	−7.4E-2	−4.9E-4	−2.1E-8	1.2E-11	−2.1E-8
2.4	7.359381	−7.5E-2	−4.8E-4	−2.1E-8	8.6E-12	−2.1E-8
2.6	7.266037	−7.5E-2	−5.0E-4	−2.1E-8	4.6E-12	−2.1E-8
2.8	7.173501	−7.6E-2	−5.0E-4	−2.2E-8	6.1E-13	−2.2E-8
3.0	7.081775	−7.7E-2	−5.1E-4	−2.2E-8	−3.5E-12	−2.2E-8
3.2	6.990857	−7.8E-2	−5.2E-4	−2.2E-8	−7.7E-12	−2.2E-8
3.4	6.900745	−7.9E-2	−5.2E-4	−2.2E-8	−1.2E-11	−2.2E-8
3.6	6.811440	−8.0E-2	−5.3E-4	−2.3E-8	−1.7E-11	−2.3E-8
3.8	6.722939	−8.1E-2	−5.3E-4	−2.3E-8	−2.0E-11	−2.3E-8
4.0	6.635242	−8.1E-2	−5.4E-4	−2.3E-8	−2.4E-11	−2.3E-8

Table 3

Benchmark prices for the VG model and relative errors for the FFT method with different choices of N. The parameter values are the same as Figure 3 except we fix $S_{10} = 100$, $S_{20} = 96$, $\bar{u} = 40$. The interpolation is based on a matrix of prices with discretization of $N = 256$ and a polynomial with a degree of 8.

Strike K	Benchmark	64	128	256	512	Interpolation
2.0	9.727458	−5.9E-2	−3.9E-4	1.5E-8	3.2E-8	1.5E-8
2.2	9.630005	−5.9E-2	−3.9E-4	1.7E-8	3.4E-8	1.7E-8
2.4	9.533199	−6.0E-2	−3.9E-4	1.8E-8	3.5E-8	1.8E-8
2.6	9.437040	−6.0E-2	−4.0E-4	2.0E-8	3.7E-8	2.0E-8
2.8	9.341527	−6.0E-2	−4.0E-4	2.5E-8	4.3E-8	2.5E-8
3.0	9.246662	−6.1E-2	−4.0E-4	2.5E-8	4.3E-8	2.5E-8
3.2	9.152445	−6.1E-2	−4.1E-4	2.3E-8	4.1E-8	2.3E-8
3.4	9.058875	−6.2E-2	−4.1E-4	3.0E-8	4.8E-8	3.0E-8
3.6	8.965954	−6.2E-2	−4.1E-4	3.0E-8	4.8E-8	3.0E-8
3.8	8.873681	−6.3E-2	−4.2E-4	2.8E-8	4.6E-8	2.8E-8
4.0	8.782057	−6.4E-2	−4.2E-4	2.9E-8	4.7E-8	2.9E-8

of the price panel: $\mathrm{Spr}(S_0; K_1) = K_1 \cdot spread(1,1)$, $\mathrm{Spr}(S_0; K_1 e^{-\eta^*}) = K_1 e^{-\eta^*} \cdot spread(2,2)$, $\mathrm{Spr}(S_0; K_1 e^{-2\eta^*}) = K_1 e^{-2\eta^*} \cdot spread(3,3)$, ... The results of this technique are recorded in Tables 2 and 3 in the column "Interpolation." We can see this technique generates very accurate results and moreover, saves computational resources.

Finally, we computed first order Greeks using the method described at the beginning of section 3 and compared them with finite differences. As seen in Table 4, the two methods come up with very consistent results. The Greeks of our at-the-money spread option exhibit some resemblance to those of the at-the-money European put/call option. The delta of S_1 is close to the delta of the call option, which is about 0.5. And the delta of S_2 is close to the delta of the put option, which is also about 0.5. The time premium of the spread option is positive. The option price is much more sensitive to S_1 volatility than to S_2 volatility. It is an important feature that the option price is negatively correlated with the underlying correlation: Intuitively speaking, if the two underlyings are strongly correlated, their comovements

Table 4

The Greeks for the GBM model compared between the FFT method and the finite difference method. The FFT method uses $N = 2^{10}$ and $\bar{u} = 40$. The finite difference uses a two-point central formula, in which the displacement is $\pm 1\%$. Other parameters are the same as Table 1 except that we fix the strike $K = 4.0$ to make the option at-the-money.

	Delta(S1)	Delta(S2)	Theta	Vega(σ_1)	Vega(σ_2)	$\partial\text{Spr}/\partial\rho$
FD	0.512648	-0.447127	3.023823	33.114315	-0.798959	-4.193749
FFT	0.512705	-0.447079	3.023777	33.114834	-0.798972	-4.193728

Table 5

Computing time of FFT for a panel of prices.

Discretization	GBM	SV	VG
64	0.091647	0.083326	0.109537
128	0.099994	0.120412	0.139276
256	0.126687	0.234024	0.220364
512	0.240938	0.711395	0.621074
1024	0.609860	2.628901	2.208770
2048	2.261325	10.243228	8.695122

diminish the probability that S_{1T} develops a wide spread over S_{2T}. This result is consistent with observations made by [10].

Since the FFT method naturally generates a panel of prices and interpolation can be implemented accurately with negligible additional computational cost, it is appropriate to measure the efficiency of the method by timing the computation of a panel of prices. Such computing times are shown in Table 5. For the FFT method, the main computational cost comes from the calculation of the matrix H in (3.2) and the subsequent FFT of H. We see that the GBM model is noticeably faster than the SV and VG models: This is due to a recursive method used to calculate the H matrix entries of the GBM model, which is not applicable for the SV and VG models. The number of calculations for H is of order N^2, which for large N exceeds the $N \log N$ of the FFT of H, and thus the advantage of this efficient algorithm for the GBM model is magnified as N increases. However, our FFT method is still very fast for the SV and VG models and is able to generate a large panel of prices within a couple of seconds.

7. High-dimensional basket options. The ideas of section 2 turn out to extend naturally to two particular classes of basket options on $M \geq 2$ assets.

Proposition 7.1. Let $M \geq 2$.

1. *For any real numbers $\epsilon = (\epsilon_1, \ldots, \epsilon_M)$ with $\epsilon_m > 0$ for $2 \leq m \leq M$ and $\epsilon_1 < -1 - \sum_{m=2}^{M} \epsilon_m$,*

$$(7.1) \qquad \left(e^{x_1} - \sum_{m=2}^{M} e^{x_m} - 1 \right)^{+} = (2\pi)^{-M} \int_{\mathbb{R}^M + i\epsilon} e^{iux'} \hat{P}^M(u) d^M u,$$

where for $u = (u_1, \ldots, u_M) \in \mathbb{C}^M$

$$(7.2) \qquad \hat{P}^M(u) = \frac{\Gamma(i(u_1 + \sum_{m=2}^{M} u_m) - 1) \prod_{m=2}^{M} \Gamma(-iu_m)}{\Gamma(iu_1 + 1)}.$$

2. *For any real numbers $\epsilon = (\epsilon_1, \ldots, \epsilon_M)$ with $\epsilon_m > 0$ for all $m \leq M$,*

$$(7.3) \qquad \left(1 - \sum_{m=1}^{M} e^{x_m}\right)^+ = (2\pi)^{-M} \int_{\mathbb{R}^M + i\epsilon} e^{iux'} \hat{Q}^M(u) d^M u,$$

where for $u = (u_1, \ldots, u_M) \in \mathbb{C}^M$

$$(7.4) \qquad \hat{Q}^M(u) = \frac{\prod_{m=1}^{M} \Gamma(-iu_m)}{\Gamma(-i \sum_{m=1}^{M} u_m + 2)}.$$

Remark. Clearly, these two results can be applied directly to obtain an M-dimensional FFT method to price M-asset basket options that pay off either $(S_{1T} - S_{2T} - \cdots - S_{MT} - 1)^+$ or $(1 - S_{1T} - S_{2T} - \cdots - S_{MT})^+$. However, it is important to also note that by a generalized "put-call parity" one can also price options that pay off either $(1 + S_{2T} + \cdots + S_{MT} - S_{1T})^+$ or $(S_{1T} + S_{2T} + \cdots + S_{MT} - 1)^+$.

Proof. The proof of both parts of the above proposition is based on a simple lemma proved in the appendix.

Lemma 7.2. *Let $z \in \mathbb{R}$ and $u = (u_1, \ldots, u_M)' \in \mathbb{C}^M$ with $\Im m(u_m) > 0$ for all $m \leq M$. Then*

$$(7.5) \qquad \int_{\mathbb{R}^M} e^z \delta\left(e^z - \sum_{m=1}^{M} e^{x_m}\right) e^{-iux'} d^M x = \frac{\prod_{m=1}^{M} \Gamma(-iu_m)}{\Gamma(-i \sum_{m=1}^{M} u_m)} e^{-i(\sum_{m=1}^{M} u_m)z}.$$

To prove (7.2), we need to compute, for $u \in \mathbb{C}^M$,

$$\hat{P}^M(u) = \int_{\mathbb{R}^M} \left(e^{x_1} - \sum_{m=2}^{M} e^{x_m} - 1\right)^+ e^{-i\tilde{u}\tilde{x}} d^M x.$$

We introduce the factor $1 = \int_{\mathbb{R}} \delta(e^z - \sum_{m=2}^{M} e^{x_m}) e^z dz$ and interchange the z integral with the x integrals. Then using Lemma 7.2 one finds

$$\hat{P}^M(u) = \int_{\mathbb{R}^2} (e^{x_1} - e^z - 1)^+ \left[\int_{\mathbb{R}^{M-1}} e^z \delta\left(e^z - \sum_{m=2}^{M} e^{x_m}\right) e^{-iux'} dx_2 \ldots dx_M\right] dx_1 dz$$

$$= \frac{\prod_{m=2}^{M} \Gamma(-iu_m)}{\Gamma(-i \sum_{m=2}^{M} u_m)} \int_{\mathbb{R}^2} e^{-iu_1 x_1} e^{-i(\sum_{m=2}^{M} u_m)z} (e^{x_1} - e^z - 1)^+ dx_1 dz.$$

We can then apply Theorem 1.1 and obtain the result.

The proof of (7.4) is similar to the proof of (7.2), where the two-dimensional problem can be deduced first and extended to higher dimensions with the application of Lemma 7.2. ∎

8. Conclusion. This paper presents a new approach to the valuation of spread options, an important class of financial contracts. The method is based on a newly discovered explicit formula for the Fourier transform of the spread option payoff in terms of the gamma function.

In the final section we extended this formula to spread options in all dimensions and a certain class of basket options.

This mathematical result leads to simple and transparent algorithms for pricing spread options and other basket options in all dimensions. We have shown that the powerful tool of the FFT provides an accurate and efficient implementation of the pricing formula in low dimensions. For implementation of higher-dimensional problems, the curse of dimensionality sets in, and such cases should proceed using parallel partitioning methods as introduced in [9]. The difficulties and pitfalls of the FFT, of which there are admittedly several, are by now well understood, and thus the reliability and stability properties of our method are clear. We present a detailed discussion of errors and show which criteria determine the optimal choice of implementation parameters.

Many important processes in finance, particularly affine models and Lévy jump models, have well-known explicit characteristic functions and can be included in the method with little difficulty. Thus the method can easily be applied to important problems arising in energy and commodity markets.

Finally, the Greeks can be systematically evaluated for such models, with similar performance and little extra work.

While our method provides a basic analytic framework for spread options, much as has been done for one-dimensional options, it is certainly possible to add refinements that will improve convergence rates. Such techniques might include, for example, analytic computation of residues combined with contour deformation.

Appendix. Proof of Theorem 1.1 and Lemma 7.2.

Proof of Theorem 1.1. Suppose $\epsilon_2 > 0$, $\epsilon_1 + \epsilon_2 < -1$. One can then verify either directly or from the argument that follows that $e^{\epsilon \cdot x} P(x)$, $\epsilon = (\epsilon_1, \epsilon_2)$ is in $\mathbb{L}^2(\mathbb{R}^2)$. Therefore, application of the Fourier inversion theorem to $e^{\epsilon \cdot x} P(x)$, $\epsilon = (\epsilon_1, \epsilon_2)$ implies that

$$(A.1) \qquad P(x) = (2\pi)^{-2} \iint_{\mathbb{R}^2 + i\epsilon} e^{iu \cdot x} g(u) d^2 u,$$

where

$$g(u) = \iint_{\mathbb{R}^2} e^{-iu \cdot x} P(x) d^2 x.$$

By restricting to the domain $\{x : x_1 > 0, e^{x_2} < e^{x_1} - 1\}$ we have

$$g(u) = \int_0^\infty e^{-iu_1 x_1} \left[\int_{-\infty}^{\log(e^{x_1}-1)} e^{-iu_2 x_2} [(e^{x_1} - 1) - e^{x_2}] dx_2 \right] dx_1$$

$$= \int_0^\infty e^{-iu_1 x_1} (e^{x_1} - 1)^{1-iu_2} \left[\frac{1}{-iu_2} - \frac{1}{1 - iu_2} \right] dx_1.$$

The change of variables $z = e^{-x_1}$ then leads to

$$g(u) = \frac{1}{(1 - iu_2)(-iu_2)} \int_0^1 z^{iu_1} \left(\frac{1-z}{z} \right)^{1-iu_2} \frac{dz}{z}.$$

The beta function

$$B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$$

is defined for any complex a, b with $\Re e(a), \Re e(b) > 0$ by

$$B(a,b) = \int_0^1 z^{a-1}(1-z)^{b-1}dz.$$

From this and the property $\Gamma(z) = (z-1)\Gamma(z-1)$ we have the formulas

$$(A.2) \qquad g(u) = \frac{\Gamma(i(u_1+u_2)-1)\Gamma(-iu_2+2)}{(1-iu_2)(-iu_2)\Gamma(iu_1+1)} = \frac{\Gamma(i(u_1+u_2)-1)\Gamma(-iu_2)}{\Gamma(iu_1+1)}. \qquad \blacksquare$$

The above derivation also leads to the following bound on $\hat{P}$.

Corollary A.1. *Fix $\epsilon_2 = \epsilon$, $\epsilon_1 = -1 - 2\epsilon$ for some $\epsilon > 0$. Then*

$$(A.3) \qquad |\hat{P}(u_1, u_2)| \le \frac{\Gamma(\epsilon)\Gamma(2+\epsilon)}{\Gamma(2+2\epsilon)} \cdot \frac{1}{Q(|u|^2/5)^{1/2}},$$

where $Q(z) = (z + \epsilon^2)(z + (1+\epsilon)^2)$.

Proof. First note that for $z_1, z_2 \in \mathbb{C}$, $|B(z_1, z_2)| \le B(\Re e(z_1), \Re e(z_2))$. Then (A.2) and a symmetric formula with $u_2 \leftrightarrow -1 - u_1 - u_2$ lead to the upper bound

$$|\hat{P}(u_1 - i(\epsilon+1), u_2 + i\epsilon)| \le B(\epsilon, 2+\epsilon) \min\left(\frac{1}{Q(|u_2|)}, \frac{1}{Q(|u_1+u_2|)}\right).$$

But since Q is monotonic and $|u| \le \sqrt{5}\max(|u_2|, |u_1+u_2|)$ for all $u \in \mathbb{R}^2$, the required result follows. $\blacksquare$

Proof of Lemma 7.2. We make the change of variables $p = e^z$ and $q_m = e^{x_m}$ and prove by induction that

$$(A.4) \qquad \int_{\mathbb{R}^M} p\delta\left(p - \sum_{m=1}^M q_m\right) \prod_{m=1}^M q_m^{-iu_m-1} d^M q = \frac{\prod_{m=1}^M \Gamma(-iu_m)}{\Gamma(-i\sum_{m=1}^M u_m)} p^{-i(\sum_{m=1}^M u_m)}.$$

The above equation trivially holds when $M = 1$. If it holds for $M = N$, then for $M = N+1$ one finds

$$LHS = \int_{\mathbb{R}^{N+1}} p\delta\left(p - q_{N+1} - \sum_{m=1}^N q_m\right) q_{N+1}^{-iu_{N+1}-1} \prod_{m=1}^N q_m^{-iu_m-1} d^{N+1}q$$

$$(A.5) \qquad = \frac{\prod_{m=1}^N \Gamma(-iu_m)}{\Gamma(-i\sum_{m=1}^N u_m)} \int_0^p \frac{p}{p - q_{N+1}} (p - q_{N+1})^{-i(\sum_{m=1}^N u_m)} q_{N+1}^{-iu_{N+1}-1} dq_{N+1}.$$

The proof is complete when one notices that the q_{N+1} integral is simply $p^{-i(\sum_{m=1}^{N+1} u_m)}$ multiplied by a beta function with parameters $-i(\sum_{m=1}^N u_m)$ and $-iu_{N+1}$. $\blacksquare$

REFERENCES

[1] A. ANTONOV AND M. ARNEGUY, *Analytical Formulas for Pricing CMS Products in the LIBOR Market Model with the Stochastic Volatility*, SSRN eLibrary, 2009.

[2] R. CARMONA AND V. DURRLEMAN, *Pricing and hedging spread options*, SIAM Rev., 45 (2003), pp. 627–685.

[3] R. CARMONA AND V. DURRLEMAN, *Pricing and Hedging Spread Options in a Log-Normal Model*, Tech. report, Princeton University, Princeton, NJ, 2003. Available online from http://orfe.princeton.edu/~rcarmona/download/fe/spread.pdf.

[4] P. CARR AND D. MADAN, *Option valuation using the fast Fourier transform*, J. Comput. Finance, 2 (1999), pp. 61–73.

[5] M. A. H. DEMPSTER AND S. S. G. HONG, *Spread option valuation and the fast Fourier transform*, in Mathematical Finance—Bachelier Congress 2000, Springer, Berlin, 2002, pp. 203–220.

[6] K. R. JACKSON, S. JAIMUNGAL, AND V. SURKOV, *Fourier space time-stepping for option pricing with Lévy models*, J. Comput. Finance, 12 (2008), pp. 1–29.

[7] E. KIRK, *Correlations in the energy markets*, in In Managing Energy Price Risk, Risk Publications and Enron, London, 1995, pp. 71–78.

[8] R. LEE, *Option pricing by transform methods: Extensions, unification, and error control*, J. Comput. Finance, 7 (2004), pp. 51–86.

[9] C. C. W. LEENTVAAR AND C. W. OOSTERLEE, *Multi-asset option pricing using a parallel Fourier-based technique*, J. Comput. Finance, 12 (2008), pp. 1–26.

[10] M. LI, S. DENG, AND J. ZHOU, *Closed-form approximations for spread option prices and Greeks*, J. Derivatives, 15 (2008), pp. 58–80.

[11] R. LORD, F. FANG, F. BERVOETS, AND C. W. OOSTERLEE, *A fast and accurate FFT-based method for pricing early-exercise options under Lévy processes*, SIAM J. Sci. Comput., 30 (2008), pp. 1678–1705.

[12] D. MADAN, P. CARR, AND E. CHANG, *The variance gamma model and option pricing*, Eur. Finance Rev., 3 (1998), pp. 79–105.

[13] D. MADAN AND E. SENETA, *The VG model for share market returns*, J. Business, 63 (1990), pp. 511–524.

[14] W. MARGRABE, *The value of an option to exchange one asset for another*, J. Finance, 33 (1978), pp. 177–186.

[15] W. H. PRESS, S. A. TEUKOLSKY, W. T. VETTERLING, AND B. P. FLANNERY, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed., Cambridge University Press, Cambridge, UK, 2007.