Fourier Methods for Option Pricing

Mathematical Foundations and Implementation

Casper J.P. van der Wouden*

November 2025

Abstract

This note provides the derivations underlying the Fourier transform-based option pricing implementation. The document shows the Carr and Madan (1999) formula, details its discretization via the Fast Fourier Transform, and present the characteristic functions for three models: Black-Scholes, Merton jump-diffusion, and Heston stochastic volatility. The document corresponds directly to the accompanying R-codebase: `fourier_option_pricing.R`.

1 Introduction & Motivation

The valuation of European options under the Black and Scholes (1973) model admits a closed-form solution. However, this model's assumptions: constant volatility and continuous price paths, are often empirically violated. More *realistic* models incorporating jumps (Merton, 1976) or stochastic volatility (Heston, 1993) generally lack analytical pricing formulas.

Two standard numerical approaches exist:

(i) **Monte Carlo simulation**: flexible but computationally expensive, with convergence rate $O(N^{-1/2})$.

(ii) **Finite difference methods**: solve the pricing PDE on a grid, but extending to jump processes requires solving PIDEs.

Fourier transform methods offer a compelling alternative. The key observation is that while the probability density of log-returns may lack a closed form, the *characteristic function* often does. Since characteristic functions uniquely determine distributions, we can recover option prices via Fourier inversion.

The Carr and Madan (1999) method achieves $O(N \log N)$ complexity via the *Fast Fourier Transform* (Cooley and Tukey, 1965), pricing an entire chain of options simultaneously.

Implementation Reference

This section corresponds to the description in the repository `README.md`. The accompanying R script `fourier_option_pricing.R` implements the Carr–Madan FFT pricing method developed throughout this note.

*Maastricht University SBE, Netherlands, Email: c.vanderwouden@student.maastrichtuniversity.nl

2 Setup and Notation

We work under a risk-neutral probability measure $\mathbb{Q}$. Let:

$$\begin{aligned} S_t &= \text{spot price at time } t \\ S_0 &= \text{current spot price} \\ K &= \text{strike price} \\ T &= \text{time to maturity} \\ r &= \text{risk-free rate (continuously compounded)} \\ q &= \text{continuous dividend yield} \end{aligned}$$

Define the **log-price** and **log-strike**:

$$x_T \coloneqq \ln(S_T), \qquad k \coloneqq \ln(K) \tag{1}$$

The **log-return** from time 0 to T is:

$$X_T \coloneqq \ln(S_T/S_0) = x_T - x_0 \tag{2}$$

Definition 2.1 (Characteristic Function)**.** The characteristic function of a random variable X is defined as:

$$\phi_X(\omega) \coloneqq \mathbb{E}\left[e^{\mathrm{i}\omega X}\right] = \int_{-\infty}^{\infty} e^{\mathrm{i}\omega x} f_X(x)\,\mathrm{d}x \tag{3}$$

where $\omega \in \mathbb{R}$ and f_X is the probability density function of X.

Remark 2.1. The characteristic function always exists (unlike moment generating functions) and uniquely determines the distribution. Crucially, ϕ_X is the Fourier transform of the density f_X.

For our models, we work with the characteristic function of $x_T = \ln(S_T)$ under the risk-neutral measure:

$$\phi_T(\omega) \coloneqq \mathbb{E}^{\mathbb{Q}}\left[e^{\mathrm{i}\omega x_T}\right] \tag{4}$$

3 European Option Pricing via Fourier Inversion

3.1 The Pricing Problem

A European call option with strike K and maturity T has time-0 price:

$$C(K) = e^{-rT}\mathbb{E}^{\mathbb{Q}}\left[(S_T - K)^+\right] \tag{5}$$

Expressing this in terms of the log-price density $f(x_T)$:

$$C(k) = e^{-rT}\int_k^{\infty} (e^x - e^k) f(x)\,\mathrm{d}x \tag{6}$$

where $k = \ln(K)$.

3.2 Why Naive Fourier Inversion Fails

One might attempt to use Fourier inversion directly. Since:

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-\mathrm{i}\omega x}\phi_T(\omega)\,\mathrm{d}\omega \tag{7}$$

we could substitute into the pricing integral. However, the call price $C(k)$ as a function of log-strike k is **not square-integrable**:

- As $k \to -\infty$ (deep ITM): $C(k) \to S_0 e^{-qT} - e^k e^{-rT} \to S_0 e^{-qT}$
- As $k \to +\infty$ (deep OTM): $C(k) \to 0$

The function does not decay as $k \to -\infty$, so $C \notin L^1(\mathbb{R}) \cap L^2(\mathbb{R})$, and its Fourier transform does not exist in the classical sense.

4 The Carr-Madan Formula

Carr and Madan (1999) resolved the integrability issue via exponential dampening!

4.1 The Dampening Trick

Definition 4.1 (Modified Call Price)**.** For a dampening parameter $\alpha > 0$, define:

$$c(k) \coloneqq e^{\alpha k} C(k) \tag{8}$$

Proposition 4.1. For sufficiently large α, the modified call price $c(k)$ is square-integrable, i.e., $c \in L^2(\mathbb{R})$.

Proof. Consider the behavior of $c(k) = e^{\alpha k} C(k)$:

- As $k \to +\infty$: $C(k)$ decays faster than any exponential (for reasonable models), so $e^{\alpha k} C(k) \to 0$.
- As $k \to -\infty$: $C(k) \to S_0 e^{-qT} - e^k e^{-rT}$, so $c(k) \approx e^{\alpha k} S_0 e^{-qT} \to 0$ since $\alpha > 0$.

Thus $c(k) \to 0$ as $|k| \to \infty$, ensuring integrability. □

4.2 Fourier Transform of the Modified Price

Define the Fourier transform of $c(k)$:

$$\psi(\omega) \coloneqq \int_{-\infty}^{\infty} e^{\mathrm{i}\omega k} c(k)\,\mathrm{d}k = \int_{-\infty}^{\infty} e^{\mathrm{i}\omega k} e^{\alpha k} C(k)\,\mathrm{d}k \tag{9}$$

Theorem 4.1 (Carr-Madan Formula)**.** The Fourier transform of the modified call price is:

$$\psi(\omega) = \frac{e^{-rT}\phi_T(\omega - (\alpha+1)\mathrm{i})}{\alpha^2 + \alpha - \omega^2 + \mathrm{i}(2\alpha+1)\omega} \tag{10}$$

where ϕ_T is the characteristic function of $x_T = \ln(S_T)$.

Proof. Start with the call price expressed as an expectation:

$$C(k) = e^{-rT} \int_k^{\infty} (e^x - e^k) f(x)\,\mathrm{d}x \tag{11}$$

Multiply by $e^{\alpha k}$ and integrate over all k:

$$\psi(\omega) = \int_{-\infty}^{\infty} e^{\mathrm{i}\omega k} e^{\alpha k} C(k)\,\mathrm{d}k \tag{12}$$

$$= e^{-rT} \int_{-\infty}^{\infty} e^{(\alpha+\mathrm{i}\omega)k} \int_k^{\infty} (e^x - e^k) f(x)\,\mathrm{d}x\,\mathrm{d}k \tag{13}$$

Applying Fubini's theorem to switch the order of integration:

$$\psi(\omega) = e^{-rT} \int_{-\infty}^{\infty} f(x) \int_{-\infty}^{x} e^{(\alpha+\mathrm{i}\omega)k} (e^x - e^k)\,\mathrm{d}k\,\mathrm{d}x \tag{14}$$

Evaluate the inner integral. Let $\beta = \alpha + \mathrm{i}\omega$:

$$\int_{-\infty}^{x} e^{\beta k}(e^x - e^k)\,\mathrm{d}k = e^x \int_{-\infty}^{x} e^{\beta k}\,\mathrm{d}k - \int_{-\infty}^{x} e^{(\beta+1)k}\,\mathrm{d}k \tag{15}$$

$$= e^x \cdot \frac{e^{\beta x}}{\beta} - \frac{e^{(\beta+1)x}}{\beta+1} \tag{16}$$

$$= \frac{e^{(\beta+1)x}}{\beta} - \frac{e^{(\beta+1)x}}{\beta+1} \tag{17}$$

$$= e^{(\beta+1)x}\left(\frac{1}{\beta} - \frac{1}{\beta+1}\right) \tag{18}$$

$$= \frac{e^{(\beta+1)x}}{\beta(\beta+1)} \tag{19}$$

Substituting back:

$$\psi(\omega) = \frac{e^{-rT}}{\beta(\beta+1)} \int_{-\infty}^{\infty} e^{(\beta+1)x} f(x)\,\mathrm{d}x \tag{20}$$

$$= \frac{e^{-rT}}{\beta(\beta+1)}\,\mathbb{E}^{\mathbb{Q}}\left[e^{(\beta+1)x_T}\right] \tag{21}$$

$$= \frac{e^{-rT}}{\beta(\beta+1)}\,\phi_T(-\mathrm{i}(\beta+1)) \tag{22}$$

where we used $\mathbb{E}[e^{ax}] = \phi(-\mathrm{i}a)$ for real a.
Substituting $\beta = \alpha + \mathrm{i}\omega$:

$$\beta + 1 = \alpha + 1 + \mathrm{i}\omega \tag{23}$$

Thus:

$$-\mathrm{i}(\beta+1) = -\mathrm{i}(\alpha + 1 + \mathrm{i}\omega) = \omega - \mathrm{i}(\alpha+1) \tag{24}$$

And:

$$\beta(\beta+1) = (\alpha + \mathrm{i}\omega)(\alpha + 1 + \mathrm{i}\omega) \tag{25}$$

$$= \alpha(\alpha+1) + \mathrm{i}\omega\alpha + \mathrm{i}\omega(\alpha+1) + (\mathrm{i}\omega)^2 \tag{26}$$

$$= \alpha^2 + \alpha + \mathrm{i}\omega(2\alpha+1) - \omega^2 \tag{27}$$

$$= \alpha^2 + \alpha - \omega^2 + \mathrm{i}(2\alpha+1)\omega \tag{28}$$

Therefore:

$$\psi(\omega) = \frac{e^{-rT}\phi_T(\omega - (\alpha+1)\mathrm{i})}{\alpha^2 + \alpha - \omega^2 + \mathrm{i}(2\alpha+1)\omega} \tag{29}$$

□

4.3 Fourier Inversion

To recover the call price, we invert the Fourier transform:

$$c(k) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega k}\,\psi(\omega)\,d\omega. \tag{30}$$

Since $c(k)$ is real, its Fourier transform satisfies $\psi(-\omega) = \overline{\psi(\omega)}$. Using this symmetry and the fact that the imaginary parts cancel, we can write

$$c(k) = \frac{1}{\pi} \int_{0}^{\infty} \mathrm{Re}\Big[e^{-i\omega k}\,\psi(\omega)\Big]\,d\omega. \tag{31}$$

Finally, recover the call price:

$$\boxed{C(k) = \frac{e^{-\alpha k}}{\pi} \int_0^\infty \operatorname{Re}\left[e^{-\mathrm{i}\omega k}\psi(\omega)\right] \mathrm{d}\omega} \tag{32}$$

Implementation Reference

The Carr–Madan pricing formula developed in this section is implemented by the function `carr_madan_fft()` in the accompanying R script `fourier_option_pricing.R`. The helper function `price_call()` wraps `carr_madan_fft()` and uses spline interpolation so that arbitrary strikes K can be priced from the FFT output. The characteristic function ϕ_T is passed via the argument `char_func`, allowing different models to be used without changing the pricing routine.

4.4 Choice of Dampening Parameter

The parameter α must satisfy:

1. $\alpha > 0$ (ensures $c(k) \to 0$ as $k \to -\infty$)
2. $\mathbb{E}^{\mathbb{Q}}[S_T^{\alpha+1}] < \infty$ (ensures $\phi_T(\omega - (\alpha+1)\mathrm{i})$ exists)

For most models, $\alpha \in [1, 2]$ works well. The optimal choice minimizes numerical error; Carr and Madan (1999) suggest $\alpha = 1.5$ as a default. In practice, sensitivity analysis is advisable.

5 FFT Discretization

To numerically evaluate (32), we discretize the integral and apply the FFT algorithm.

5.1 Discretization Scheme

Define a grid in frequency space:

$$\omega_j = j \cdot \eta, \qquad j = 0, 1, \ldots, N-1 \tag{33}$$

where $\eta > 0$ is the spacing and N is typically a power of 2.

The integral becomes a sum via the trapezoidal rule:

$$C(k) \approx \frac{e^{-\alpha k}}{\pi} \sum_{j=0}^{N-1} \operatorname{Re}\left[e^{-\mathrm{i}\omega_j k}\psi(\omega_j)\right] \eta \tag{34}$$

5.2 Log-Strike Grid

The FFT produces prices at log-strikes:

$$k_u = -b + u \cdot \lambda, \qquad u = 0, 1, \ldots, N-1 \tag{35}$$

where $b = \frac{N\lambda}{2}$ centers the grid around $k = 0$ (i.e., $K = S_0$).

The FFT constraint relates the grids:

$$\eta \cdot \lambda = \frac{2\pi}{N} \tag{36}$$

5.3 FFT Formulation

Substituting k_u and ω_j:

$$C(k_u) = \frac{e^{-\alpha k_u}}{\pi} \sum_{j=0}^{N-1} \text{Re}\left[e^{-\text{i}j\eta(-b+u\lambda)}\psi(j\eta)\right]\eta \tag{37}$$

$$= \frac{e^{-\alpha k_u}}{\pi}\text{Re}\left[e^{\text{i}b\eta\cdot 0}\sum_{j=0}^{N-1} e^{\text{i}j\eta b}\psi(j\eta)\eta\cdot e^{-\text{i}ju\eta\lambda}\right] \tag{38}$$

Using (36), $\eta\lambda = 2\pi/N$, so $e^{-\text{i}ju\eta\lambda} = e^{-2\pi\text{i}ju/N}$.
Define:

$$x_j := e^{\text{i}b\omega_j}\psi(\omega_j)\eta\cdot w_j \tag{39}$$

where w_j are Simpson's rule weights for improved accuracy:

$$w_j = \frac{\eta}{3}\left(3+(-1)^j-\delta_{j,0}\right) \tag{40}$$

Then:

$$\boxed{C(k_u) = \frac{e^{-\alpha k_u}}{\pi}\text{Re}\left[\sum_{j=0}^{N-1} x_j\, e^{-2\pi\text{i}ju/N}\right] = \frac{e^{-\alpha k_u}}{\pi}\text{Re}\left[\text{FFT}(\mathbf{x})_u\right]} \tag{41}$$

Algorithm 1 Carr-Madan FFT Option Pricing

Require: Characteristic function ϕ_T, parameters S_0, r, q, T, grid parameters N, η, α
Ensure: Call prices $C(K_u)$ for $u = 0, \ldots, N-1$

Compute $\lambda \leftarrow 2\pi/(N\eta)$ and $b \leftarrow N\lambda/2$
for $j = 0$ to $N-1$ **do**
  $\omega_j \leftarrow j\cdot\eta$
  Compute $\psi(\omega_j)$ via (10)
  $w_j \leftarrow \frac{\eta}{3}(3+(-1)^j - \mathbf{1}_{j=0})$ ▷ Simpson weights..
  $x_j \leftarrow e^{\text{i}b\omega_j}\psi(\omega_j)\cdot w_j$
end for
$\mathbf{y} \leftarrow \text{FFT}(\mathbf{x})$
for $u = 0$ to $N-1$ **do**
  $k_u \leftarrow -b + u\lambda$
  $K_u \leftarrow e^{k_u}$
  $C(K_u) \leftarrow \frac{e^{-\alpha k_u}}{\pi}\text{Re}(y_u)$
end for
return $\{(K_u, C(K_u))\}_{u=0}^{N-1}$

Implementation Reference

The FFT-based option pricing algorithm described in this section is implemented in the function `carr_madan_fft()` in `fourier_option_pricing.R`. The routine calls R's built-in `fft()` function and, by default, uses `N = 4096`, `eta = 0.25`, and `alpha = 1.5` for the grid parameters. These defaults can be modified via function arguments if desired.

5.4 Parameter Selection Guidelines

- N: Higher N gives more strike prices and finer resolution. $N = 2^{12} = 4096$ is standard.
- η: Controls the range of integration in frequency space. Smaller η captures slowly-varying characteristic functions but reduces strike range.
- **Trade-off**: From (36), $\lambda = 2\pi/(N\eta)$. The strike range is $[e^{-b}, e^{b}]$ where $b = N\lambda/2 = \pi/\eta$. Smaller $\eta \Rightarrow$ wider strike range but coarser frequency grid.

Remark 5.1. In the accompanying implementation, the raw FFT output is trimmed to a practical range of strikes ($\approx [0.2S_0, 2.5S_0]$), and strictly positive call prices are retained. This avoids numerical artofacts at the extreme ends of the grid when interpolating to arbitrary strikes.

6 Model Characteristic Functions

Now we derive the characteristic functions for three models, presented in order of complexity.

6.1 Black-Scholes Model

Under Black-Scholes, the stock price follows geometric Brownian motion:

$$\mathrm{d}S_t = (r - q)S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t^{\mathbb{Q}} \tag{42}$$

Proposition 6.1 (Black-Scholes Characteristic Function)**.** The characteristic function of $x_T = \ln(S_T)$ is:

$$\phi_T^{BS}(\omega) = \exp\left(\mathrm{i}\omega\left[\ln S_0 + (r - q - \tfrac{1}{2}\sigma^2)T\right] - \frac{1}{2}\sigma^2\omega^2 T\right) \tag{43}$$

Proof. By Itô's lemma applied to $x_t = \ln(S_t)$:

$$\mathrm{d}x_t = (r - q - \tfrac{1}{2}\sigma^2)\mathrm{d}t + \sigma\mathrm{d}W_t^{\mathbb{Q}} \tag{44}$$

Thus x_T is normally distributed:

$$x_T \sim \mathcal{N}\left(\ln S_0 + (r - q - \tfrac{1}{2}\sigma^2)T,\ \sigma^2 T\right) \tag{45}$$

For $X \sim \mathcal{N}(\mu, \sigma^2)$, the characteristic function is $\phi_X(\omega) = \exp(\mathrm{i}\omega\mu - \frac{1}{2}\sigma^2\omega^2)$. □

Implementation Reference

The Black–Scholes characteristic function and closed-form call price are implemented as `bs_char_func()` and `bs_call()` in the main R script `fourier_option_pricing.R`. The Carr–Madan FFT prices under Black–Scholes are compared to `bs_call()` in the numerical examples in this section as a verification check of the implementation.

6.2 Merton Jump-Diffusion Model

Merton (1976) extended Black-Scholes by adding jumps:

$$\mathrm{d}S_t = (r - q - \lambda\kappa)S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t^{\mathbb{Q}} + S_{t^-}(e^J - 1)\mathrm{d}N_t \tag{46}$$

where:

- N_t is a Poisson process with intensity λ

- $J \sim \mathcal{N}(\mu_J, \sigma_J^2)$ is the log-jump size
- $\kappa = \mathbb{E}[e^J - 1] = e^{\mu_J + \frac{1}{2}\sigma_J^2} - 1$ is the mean relative jump

Proposition 6.2 (Merton Characteristic Function)**.** The characteristic function of $x_T = \ln(S_T)$ under Merton dynamics is:

$$\phi_T^M(\omega) = \phi_T^{BS}(\omega) \cdot \exp\left(\lambda T \left[e^{\mathrm{i}\omega\mu_J - \frac{1}{2}\sigma_J^2\omega^2} - 1\right]\right) \tag{47}$$

where ϕ_T^{BS} uses drift adjusted by $-\lambda\kappa$.

Proof. The log-price evolves as:

$$x_T = x_0 + (r - q - \tfrac{1}{2}\sigma^2 - \lambda\kappa)T + \sigma W_T + \sum_{i=1}^{N_T} J_i \tag{48}$$

Conditional on $N_T = n$ jumps:

$$x_T \,|\, N_T = n \;\sim\; \mathcal{N}\left(x_0 + (r - q - \tfrac{1}{2}\sigma^2 - \lambda\kappa)T + n\mu_J,\; \sigma^2 T + n\sigma_J^2\right) \tag{49}$$

The conditional characteristic function is:

$$\phi_{x_T|N_T=n}(\omega) = \exp\left(\mathrm{i}\omega\mu_n - \tfrac{1}{2}\omega^2\sigma_n^2\right) \tag{50}$$

where $\mu_n = x_0 + (r - q - \frac{1}{2}\sigma^2 - \lambda\kappa)T + n\mu_J$ and $\sigma_n^2 = \sigma^2 T + n\sigma_J^2$.

By the law of total expectation:

$$\phi_T^M(\omega) = \mathbb{E}[\phi_{x_T|N_T}(\omega)] = \sum_{n=0}^{\infty} \phi_{x_T|N_T=n}(\omega) \cdot \frac{(\lambda T)^n e^{-\lambda T}}{n!} \tag{51}$$

The diffusion component factors out. For the jump component:

$$\sum_{n=0}^{\infty} \exp\left(\mathrm{i}\omega n\mu_J - \tfrac{1}{2}\omega^2 n\sigma_J^2\right) \frac{(\lambda T)^n e^{-\lambda T}}{n!} \tag{52}$$

$$= e^{-\lambda T} \sum_{n=0}^{\infty} \frac{\left(\lambda T e^{\mathrm{i}\omega\mu_J - \frac{1}{2}\omega^2\sigma_J^2}\right)^n}{n!} \tag{53}$$

$$= e^{-\lambda T} \cdot \exp\left(\lambda T e^{\mathrm{i}\omega\mu_J - \frac{1}{2}\omega^2\sigma_J^2}\right) \tag{54}$$

$$= \exp\left(\lambda T \left[e^{\mathrm{i}\omega\mu_J - \frac{1}{2}\omega^2\sigma_J^2} - 1\right]\right) \tag{55}$$

□

Remark 6.1. The Merton model nests Black-Scholes when $\lambda = 0$ (no jumps).

Implementation Reference

The Merton jump–diffusion characteristic function is implemented by the function `merton_char_func()` in `fourier_option_pricing.R`, with parameters `sigma` (diffusion volatility), `lambda` (jump intensity), `mu_j` (mean log-jump), and `sigma_j` (jump volatility). These parameters are passed to `price_call()` to generate the Merton call price surface and implied volatility smiles shown in the numerical results.

6.3 Heston Stochastic Volatility Model

The Heston (1993) model allows volatility to vary stochastically:

$$\mathrm{d}S_t = (r-q)S_t\,\mathrm{d}t + \sqrt{V_t}S_t\,\mathrm{d}W_t^{(1)} \tag{56}$$

$$\mathrm{d}V_t = \kappa(\theta - V_t)\mathrm{d}t + \xi\sqrt{V_t}\,\mathrm{d}W_t^{(2)} \tag{57}$$

with $\mathrm{d}W_t^{(1)}\mathrm{d}W_t^{(2)} = \rho\,\mathrm{d}t$.

Parameters:

- V_0: initial variance
- θ: long-run variance (mean-reversion level)
- κ: mean-reversion speed
- ξ: volatility of volatility
- ρ: correlation between spot and volatility

Theorem 6.1 (Heston Characteristic Function)**.** The characteristic function of $x_T = \ln(S_T)$ is:

$$\phi_T^H(\omega) = \exp\left(\mathrm{i}\omega(x_0 + (r-q)T) + A(\omega,T) + B(\omega,T)V_0\right) \tag{58}$$

where:

$$A(\omega,T) = \frac{\kappa\theta}{\xi^2}\left[(\kappa - \rho\xi\mathrm{i}\omega + d)T - 2\ln\left(\frac{1-ge^{dT}}{1-g}\right)\right] \tag{59}$$

$$B(\omega,T) = \frac{\kappa - \rho\xi\mathrm{i}\omega + d}{\xi^2}\cdot\frac{1-e^{dT}}{1-ge^{dT}} \tag{60}$$

and:

$$d = \sqrt{(\rho\xi\mathrm{i}\omega - \kappa)^2 + \xi^2(\mathrm{i}\omega + \omega^2)} \tag{61}$$

$$g = \frac{\kappa - \rho\xi\mathrm{i}\omega + d}{\kappa - \rho\xi\mathrm{i}\omega - d} \tag{62}$$

Proof Sketch. The characteristic function satisfies the Feynman-Kac PDE:

$$\frac{\partial\phi}{\partial t} + (r-q)x\frac{\partial\phi}{\partial x} + \kappa(\theta - v)\frac{\partial\phi}{\partial v} + \frac{1}{2}v\frac{\partial^2\phi}{\partial x^2} + \rho\xi v\frac{\partial^2\phi}{\partial x\partial v} + \frac{1}{2}\xi^2 v\frac{\partial^2\phi}{\partial v^2} = 0 \tag{63}$$

Guess an affine solution $\phi = \exp(A + Bv + \mathrm{i}\omega x)$. Substituting yields a system of Riccati ODEs for $A(t)$ and $B(t)$. The solution follows from standard Riccati equation techniques.

See Heston (1993) for the complete derivation. □

Remark 6.2 (Numerical Stability)**.** The formulation above can suffer from branch-cut issues for large $|\omega|$. The implementation uses the "Little Heston Trap" formulation of Albrecher et al. (2007) for improved stability.

Implementation Reference

The Heston stochastic volatility model is implemented via the characteristic function `heston_char_func()` in the main R script `fourier_option_pricing.R`. The function takes the parameters `V0`, `theta`, `kappa`, `xi`, and `rho`, and is passed to `price_call()` to obtain Heston call prices and implied volatilities. These outputs are used to produce the Heston price surfaces and smiles in the figures of this section.

7 Model Calibration

Given market option prices, calibration finds model parameters that best match observed prices.

7.1 Problem Formulation

Let $\{C_i^{\text{mkt}}\}_{i=1}^M$ be observed market prices for options with strikes $\{K_i\}$ and maturities $\{T_i\}$. Let $C_i^{\text{model}}(\boldsymbol{\theta})$ denote model prices for parameter vector $\boldsymbol{\theta}$.

The calibration problem is:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{M} w_i \left(C_i^{\text{model}}(\boldsymbol{\theta}) - C_i^{\text{mkt}} \right)^2 \tag{64}$$

where $w_i > 0$ are weights (e.g., inverse bid-ask spread or vega-weighting).

7.2 Objective Function Variants

Price-based (absolute):

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{M} \left(C_i^{\text{model}} - C_i^{\text{mkt}} \right)^2 \tag{65}$$

Price-based (relative):

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{M} \left(\frac{C_i^{\text{model}} - C_i^{\text{mkt}}}{C_i^{\text{mkt}}} \right)^2 \tag{66}$$

Implied volatility-based:

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{M} \left(\sigma_i^{\text{model}}(\boldsymbol{\theta}) - \sigma_i^{\text{mkt}} \right)^2 \tag{67}$$

The IV-based objective is often preferred as it normalizes across strikes and maturities.

7.3 Optimization

The objective (64) is non-convex with potentially many local minima. Recommended approaches:

1. **Global search**: Differential evolution or simulated annealing for initial parameter guess
2. **Local refinement**: Levenberg-Marquardt or L-BFGS-B for polishing

Parameter constraints (Heston example):

- $\kappa > 0$, $\theta > 0$, $\xi > 0$, $V_0 > 0$
- $-1 < \rho < 1$
- Feller condition: $2\kappa\theta > \xi^2$ (ensures $V_t > 0$)

Implementation Reference

This section is just conceptual in the current project! Calibration routines for estimating model parameters from market option data (for example, using global search methods followed by local refinement in R) are not implemented in `fourier_option_pricing.R`. Instead, this section outlines how the FFT pricer could be embedded into an estimation framework in future work.

8 Implementation Notes

8.1 Computational Complexity

- **Carr-Madan FFT**: $O(N \log N)$ for N strikes simultaneously
- **Monte Carlo**: $O(M \cdot P)$ for M paths and P strikes, with $O(M^{-1/2})$ convergence
- **Advantage**: FFT is typically 10^3-10^4 times faster for comparable accuracy

8.2 Numerical Considerations

1. **Complex arithmetic**: Ensure proper handling of complex exponentials, especially $\phi_T(\omega - (\alpha+1)\mathrm{i})$ which evaluates the characteristic function at complex arguments.
2. **Singularity at $\omega = 0$**: The denominator in (10) vanishes at $\omega = 0$. Use L'Hôpital's rule or simply evaluate $\psi(0)$ directly from the definition.
3. **Interpolation**: FFT produces prices at specific log-strikes. For arbitrary strikes, interpolate (cubic spline recommended).
4. **Put prices**: Use put-call parity: $P(K) = C(K) - S_0 e^{-qT} + Ke^{-rT}$.

Acknowledgments

This document accompanies the code repository at:
`[https://github.com/cjpvanderwouden/fourier-options-pricing]`.

References

Albrecher, H., Mayer, P., Schoutens, W., and Tistaert, J. (2007). The little Heston trap. *Wilmott Magazine*, pages 83–92.

Black, F. and Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637–654.

Carr, P. and Madan, D. (1999). Option valuation using the fast Fourier transform. *Journal of Computational Finance*, 2(4):61–73.

Cooley, J. W. and Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–301.

Heston, S. L. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The Review of Financial Studies*, 6(2):327–343.

Merton, R. C. (1976). Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics*, 3(1-2):125–144.

Appendix A: Numerical Output

```
> table_1
   Strike BS_Analytical FFT_Price Abs_Error
1      70       30.7488   30.7488  2.21e-07
2      75       25.9262   25.9262  2.24e-07
3      80       21.2161   21.2161  2.18e-07
4      85       16.7436   16.7436  2.15e-07
5      90       12.6719   12.6719  1.95e-07
6      95        9.1590    9.1590  2.01e-07
7     100        6.3076    6.3076  1.66e-07
8     105        4.1367    4.1367  1.78e-07
9     110        2.5859    2.5859  2.15e-07
10    115        1.5438    1.5438  2.25e-07
11    120        0.8825    0.8825  2.32e-07
12    125        0.4846    0.4846  2.21e-07
13    130        0.2563    0.2563  2.41e-07
```

Figure A.1: BS prices and errors

```
> table_2
   Strike Call_Price Implied_Vol
1      70    30.9789       29.29%
2      75    26.3209       28.05%
3      80    21.8012       26.73%
4      85    17.4787       25.35%
5      90    13.4426       23.97%
6      95     9.8233       22.72%
7     100     6.7682       21.67%
8     105     4.3822       20.89%
9     110     2.6734       20.35%
10    115     1.5513       20.04%
11    120     0.8696       19.91%
12    125     0.4803       19.96%
13    130     0.2666       20.15%
```

Figure A.2: Merton prices and vols

```
> table_3
   Strike Call_Price Implied_Vol
1      70    30.8460       25.73%
2      75    26.1055       24.68%
3      80    21.4893       23.64%
4      85    17.0766       22.62%
5      90    12.9732       21.61%
6      95     9.3053        20.6%
7     100     6.2023       19.62%
8     105     3.7683       18.66%
9     110     2.0426       17.75%
10    115     0.9691       16.94%
11    120     0.3989       16.24%
12    125     0.1435        15.7%
13    130     0.0463       15.31%
```

Figure A.3: Heston prices and vols

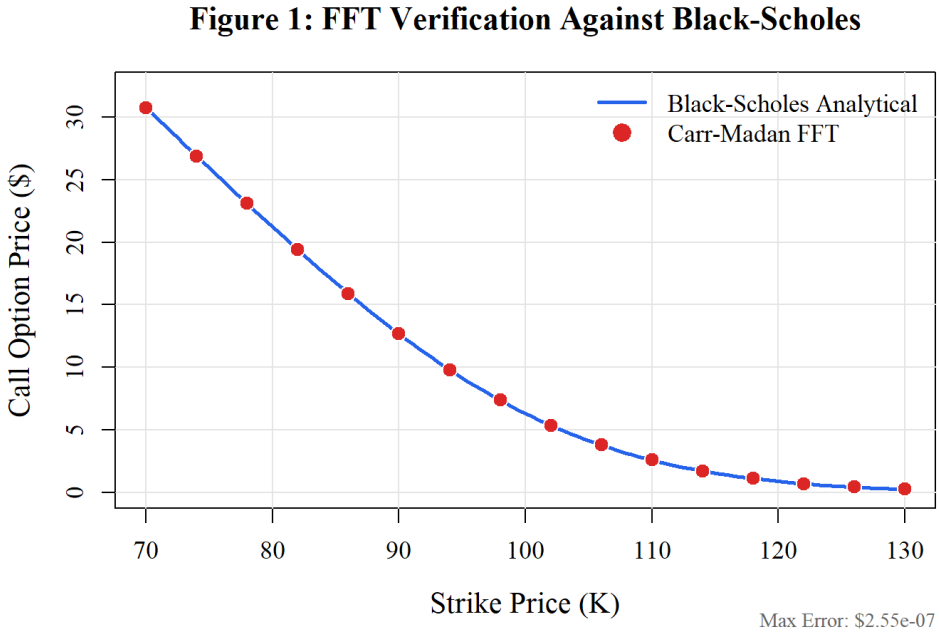

Figure A.4: FFT vs Black–Scholes

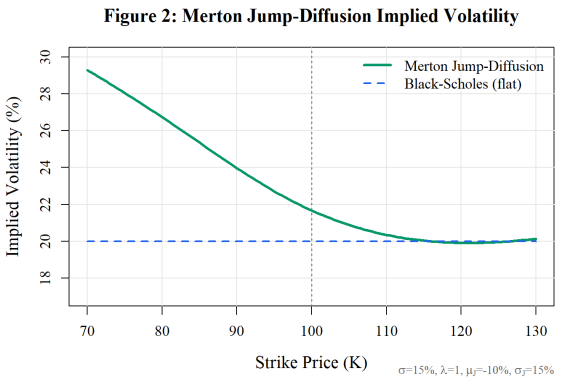

Figure A.5: Merton implied vol

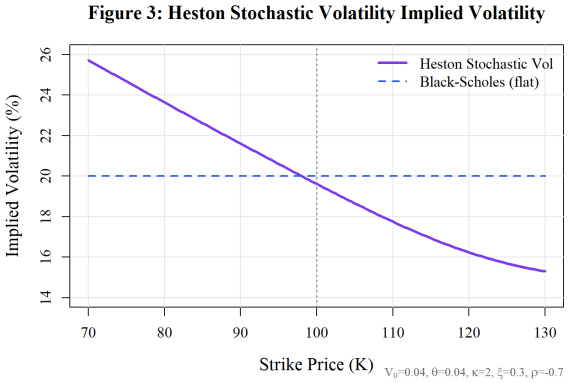

Figure A.6: Heston implied vol

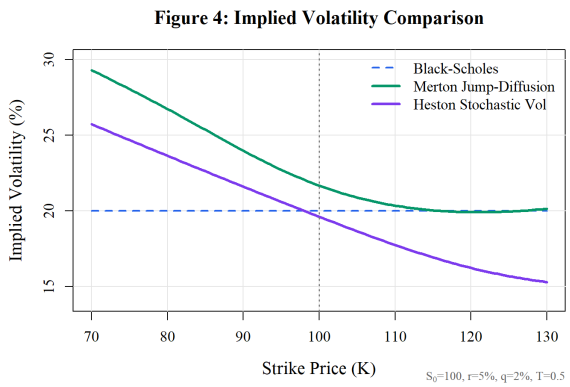

Figure A.7: IV comparison

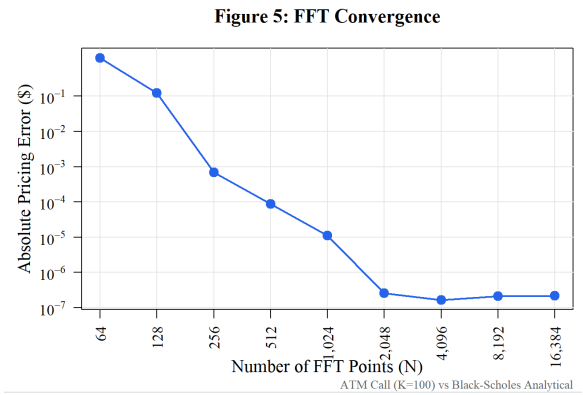

Figure A.8: FFT convergence

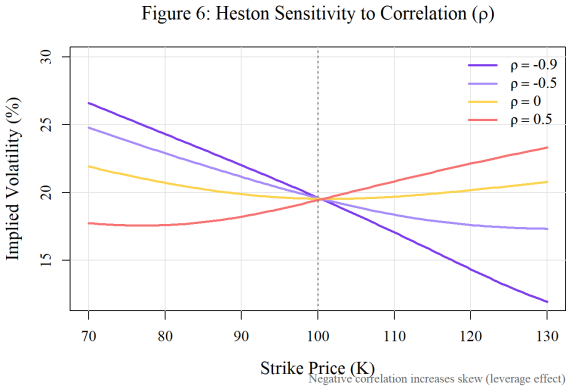

Figure A.9: Heston, varying ρ

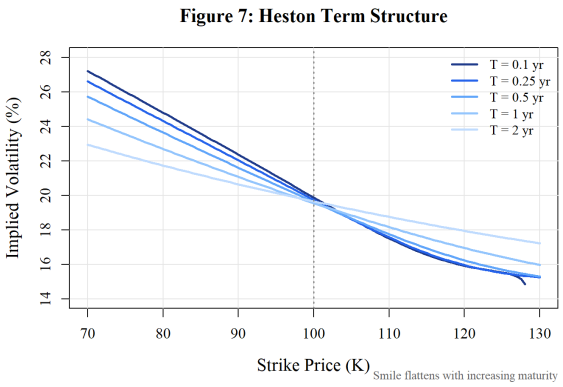

Figure A.10: Heston term structure