Notes on the BENCHOP implementations for the Fourier FFT method

Erik Lindstrm (erikl@maths.lth.se)

April 24, 2015

Abstract

This text describes the Fourier method with FFT and its implementation for the BENCHOP-project.

1 Introduction

It was shown in the seminal Heston paper on stochastic volatility Heston [1993] that the option value could be computed by inverting to Fourier integrals. This meant that the computational demands was far less than using either Monte Carlo methods, binomial or trinomial tree, see Cox et al. [1979] or PDE methods, explaining the popularity of the model.

2 Carr-Madan method

We focus on a more recent implementation in these notes, namely the Carr-Madan algorithm, see Carr and Madan [1999].

Assume that the characteristic function of the log price $s(T) = \log S(T)$ is known

$$\psi(u) = \mathbf{E}\left[e^{ius(T)}\right] = \int e^{ius}\mathrm{d}\mathbb{Q}(s). \tag{1}$$

The risk neutral measure is absolutely continuous with respect to the Lebesgue measure in virtually every models we consider in this book, and we will therefore assume that we can use the density instead $\mathrm{d}\mathbb{Q}(s) = q(s)\mathrm{d}s$.

It is known from Section **??** that the price of a European Call option is given by

$$C(k) = \int_k^\infty e^{-rT}(e^s - e^k)q(s)\mathrm{d}s \tag{2}$$

where $k = \log(K)$ is the log strike price. The option price is not square integrable (which is required by the Parseval's theorem) but a modified version of the price is

$$c(k) = e^{\alpha k}C(k) \tag{3}$$

where α is some positive number. It is then possible to compute the Fourier transform of modified price $c(k)$ as

$$\phi(v) = \int e^{ivk} c(k) \mathrm{d}k. \tag{4}$$

This expression can be extend further accordingly

$$\begin{aligned}
\phi(v) &= \int e^{ivk} c(k) \mathrm{d}k & (5)\\
&= \int e^{ivk} e^{\alpha k} \int_k^\infty e^{-rT}(e^s - e^k) q(s) \mathrm{d}s \mathrm{d}k & (6)\\
&= \frac{e^{-rT}\psi(v-(\alpha+1)i)}{\alpha^2+\alpha-v^2+i(2\alpha+1)v}. & (7)
\end{aligned}$$

It is also possible to compute the option price by applying the inverse Fourier transform to (4)

$$C(k) = \frac{e^{-\alpha k}}{2\pi} \int e^{-ivk} \phi(v) \mathrm{d}v = \frac{e^{-\alpha k}}{\pi} \int_0^\infty e^{-ivk} \phi(v) \mathrm{d}v.. \tag{8}$$

where the second equality holds as $C(k)$ is real. Hence, call prices as given by inserting equation (7) into (8) arriving at

$$C(k) = \frac{e^{-\alpha k}}{\pi} \int_0^\infty e^{-ivk} \frac{e^{-rT}\psi(v-(\alpha+1)i)}{\alpha^2+\alpha-v^2+i(2\alpha+1)v} \mathrm{d}v. \tag{9}$$

The integral can be computed using either Fast Fourier Transform (FFT) or related fast transforms, see Hirsa [2013] or Gauss-Laguerre quadrature methods Lindström et al. [2008] as both types of methods provides very accurate approximations with very limited computational efforts

$$C(k) = \frac{e^{-\alpha k}}{\pi} \sum_{j=1}^{N} e^{-iv_j k} \phi(v_j) \omega_j. \tag{10}$$

where ω_j depends on the numerical quadrature method used and $v_j = (j-1)\eta$, $j = 1, \ldots, N+1$.

The modification, here parametrized by α, is needed due to Parseval, but it can also be seen that choosing $\alpha = 0$ would introduce a singularity in (9). It can also be shown that the α parameter can dampen numerical oscillations in the integrand, leading to better numerical approximations, see Lee [2004], Lindström et al. [2008] for further details.

2.1 Implementation with the FFT algorithm

The Fat Fourier Transform (FFT) algorithm, see Cooley and Tukey [1965], is an implementation of the Discrete Fourier Transform (DFT) with the distinct advantage that the computational complexity is $\mathcal{O}(N\log(N))$ rather than $\mathcal{O}(N^2)$.

It computes

$$X(m) = \sum_{j=1}^{N} x_j e^{-i\frac{2\pi}{N}(j-1)(m-1)}, \; j = 0, \ldots, N-1. \tag{11}$$

The FFT algorithm can be used to compute Equation (10) if it can fit this form. Doing so will not only compute the option price at a single value but for a whole range of values.

First we create a range of (log-) strikes $k_m = \log(S_0) - \frac{\Delta k N}{2} + (m-1)\Delta k = \beta + (m-1)\Delta k$ for $m = 1, \ldots, N$ which should cover all relevant strike and then some.

We now get that

$$C(k_m) = \frac{e^{-\alpha k_m}}{\pi} \sum_{j=1}^{N} e^{-iv_j k_m} \phi(v_j)\omega_j \tag{12}$$

$$= \frac{e^{-\alpha k_m}}{\pi} \sum_{j=1}^{N} e^{-i(j-1)\eta(\beta+(m-1)\Delta k)} \phi(v_j)\omega_j \tag{13}$$

$$= \frac{e^{-\alpha k_m}}{\pi} \sum_{j=1}^{N} e^{-i(j-1)(m-1)\eta\Delta k} e^{-i\beta v_j} \phi(v_j)\omega_j \tag{14}$$

Choosing $\eta\Delta K = 2\pi$ and defining $x_j = e^{-i\beta v_j}\phi(v_j)\omega_j$ transforms the problem so that the FFT algorithm can be applied.

References

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

James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.

John C Cox, Stephen A Ross, and Mark Rubinstein. Option pricing: A simplified approach. *Journal of Financial Economics*, 7(3):229–263, 1979.

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

Ali Hirsa. *Computational Methods in Finance*. CRC Press, 2013.

Roger W Lee. Option pricing by transform methods: Extensions, unification and error control. *Journal of Computational Finance*, 7(3):51–86, 2004.

Erik Lindström, Jonas Ströjby, Mats Brodén, Magnus Wiktorsson, and Jan Holst. Sequential Calibration of Options. *Computational Statistics & Data Analysis*, 52(6):2877–2891, 2008.