Spitzer identity, Wiener-Hopf factorization and pricing of discretely monitored exotic options
—
Online Supplementary Material

Gianluca Fusai

Department of Economics and Business Studies (DiSEI), Università del Piemonte Orientale "Amedeo Avogadro", Novara

Faculty of Finance, Cass Business School, City University London

Guido Germano

Financial Computing and Analytics Group, Department of Computer Science, University College London

Systemic Risk Centre, London School of Economics

Daniele Marazzina

Department of Mathematics, Politecnico di Milano

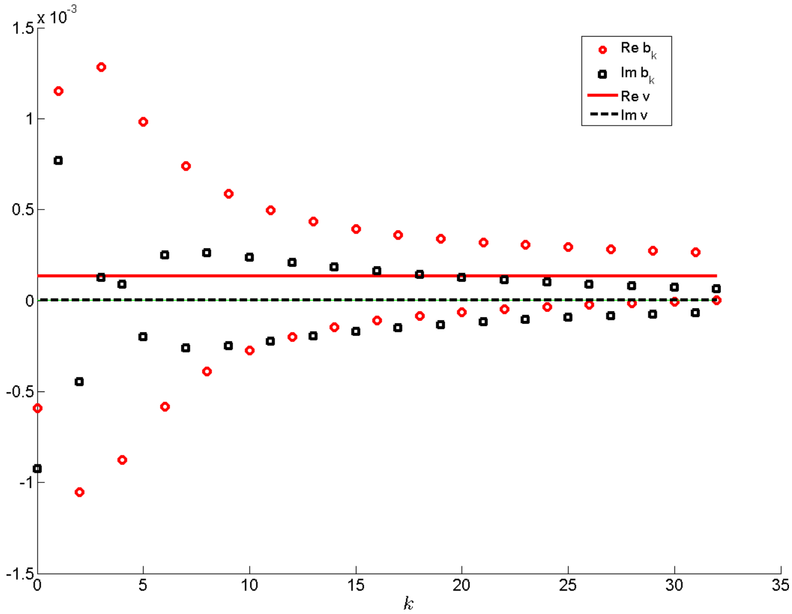
Email addresses: gianluca.fusai@unipmn.it, gianluca.fusai.1@city.ac.uk (Gianluca Fusai), g.germano@ucl.ac.uk, g.germano@lse.ac.uk (Guido Germano), daniele.marazzina@polimi.it (Daniele Marazzina)

Preprint submitted to Elsevier *September 28, 2015*

A. Acceleration of the inverse z-transform via Euler summation

In Figure A we show graphically the convergence of the partial sums b_k to the inverse z-transform.

Figure A: Convergence of the partial sum b_k to $\mathcal{Z}^{-1}_{q \to n} \widetilde{v}(\xi, q) \approx \frac{1}{2^{m_{\mathrm{E}}} n \rho^n} \sum_{j=0}^{m_{\mathrm{E}}} \binom{m_{\mathrm{E}}}{j} b_{n_{\mathrm{E}}+j}(\xi)$. The real (imaginary) part of b_k, $k = 0, 1, \ldots, n_E + m_E$, corresponds to the red circles (black squares), while the real (imaginary) part of the solution corresponds to the red (black dashed) line. The test case is related to the computation of $p_{X,m}$ with $X(t)$ a double exponential Lévy process, a log-barrier $l = 0.8$, $N = 100$ monitoring dates and $M = 2^{14}$ grid points.



B. Hilbert transform and Wiener-Hopf factorization

The Matlab code to compute the Hilbert transform via sinc functions and the FFT is shown in Figure B; the use of this fast Hilbert transform to achieve a Wiener-Hopf factorization is reported in Figure C.

Figure B: Matlab code to compute the Hilbert transform via sinc function expansion.

```
% Fast Hilbert transform: Hilbert transform through
% fast Fourier transforms.
function iHF = ifht(F)
% Setup
[M N] = size(F); % Dimension: number of equations and grid points
P = N; % Number of zero padding elements
Q = N+P; % Number of grid points after zero padding
% Define the auxiliary vector
t = (1-(-1).^(-Q/2:Q/2-1))./(pi*(-Q/2:Q/2-1));
t(Q/2+1) = 0;
vec = repmat(imag(fft(ifftshift(t))),M,1);
% Compute the Hilbert transform times the imaginary unit
f = ifft(F,Q,2); % Optional padding
iHF = fft(vec.*f,[],2);
iHF = iHF(:,1:N);
```

Figure C: Matlab code to compute the Wiener-Hopf factorization via the Hilbert transform.

```
% Factorise L = 1-H
lL = log(1-H);
iHlL = ifht(lL); % imaginary unit times fast Hilbert transform of L
lLp = (lL+iHlL)/2; % Plemelj-Sokhotsky
lLm = (lL-iHlL)/2; % Plemelj-Sokhotsky
Lp = exp(lLp);
Lm = exp(lLm);
```

C. Other Fourier-based transform methods

In this section we discuss other numerical methods presented in the literature which are based on Fourier and Hilbert transforms. We will not try to be exhaustive, but limit ourselves to those approaches that are most related to our own, and thus we will not cover e.g. the Cos method [2], as well as different approaches like the ones based on advanced quadrature [e.g. 1].

The general recursion pricing equation to compute the price (or cost) c of a plain vanilla derivative, such as an European call option, at time t given its value at time $t+\Delta$ can be computed from its price at time $t+\Delta$ using the backward-in-time density

$$c(x,t) = e^{-r\Delta}\int_{-\infty}^{+\infty} f_{\mathrm{b}}(x-x',\Delta)c(x',t+\Delta)dx'.$$

Here the derivative price is a function of the log-price x of the underlying asset and of the time t. This function $x \to c(x,t)$ is, in general, not square integrable and thus its Fourier transform does not exist. However, this problem can be worked around introducing the damped call price $C(x,t) = e^{\alpha x}c(x,t)$, $\alpha < 0$ being the so-called damping factor. The Fourier transform of the backward-in-time transition density $f_{\mathrm{b}}(x,\Delta) := f(-x,\Delta)$ is the conjugate $\Psi^*(\xi,\Delta)$ of the characteristic function. Therefore, in Fourier space the above equation becomes

$$\widehat{C}(\xi,t) = e^{-r\Delta}\Psi^*(\xi-i\alpha,\Delta)\widehat{C}(\xi,t+\Delta),$$

since

$$\begin{aligned}
\widehat{C}(\xi,t) &= \mathcal{F}_{x\to\xi}[e^{\alpha x}c(x,t)] = \int_{-\infty}^{+\infty} c(x,t)e^{ix(\xi-i\alpha)}dx \\
&= e^{-r\Delta}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f_{\mathrm{b}}(x-x',\Delta)c(x',t+\Delta)e^{ix(\xi-i\alpha)}dx'dx \\
&= e^{-r\Delta}\int_{-\infty}^{+\infty} f_{\mathrm{b}}(z,\Delta)e^{iz(\xi-i\alpha)}dz\int_{-\infty}^{+\infty} e^{i\xi x'}C(x',t+\Delta)dx' \\
&= e^{-r\Delta}\Psi^*(\xi-i\alpha,\Delta)\widehat{C}(\xi,t+\Delta),
\end{aligned}$$

changing the order of integration and defining $z = x - x'$; see Lord et al. [10] for further details. Therefore we have

$$c(\xi,t) = e^{-r\Delta}e^{-\alpha x}\mathcal{F}^{-1}_{\xi\to x}[\Psi^*(\xi-i\alpha,\Delta)\widehat{C}(\xi,t+\Delta)].$$

The methods considered in the following sections for pricing path dependent derivatives are based on the above described backward recursion from time

$t+\Delta$ to time t. For ease of exposition, we will consider only a down-and-out put barrier option and we neglect the damping factor; for a down-and-out put option the damping factor is not necessary anyway, since the payoff and the option price are square integrable functions.

C.1. Convolution and Hilbert transform

First of all, we briefly describe the convolution method [9, 10], as well as the method based on the Hilbert transform due to Feng and Linetsky [3]. Both are based on obtaining the option price recursively via

$$v(x,j) = e^{-r\Delta} \int_l^{+\infty} f_b(x-x',\Delta) v(x',j-1) dx', \tag{1}$$

where $v(x,j)$ is the value of the option for the log-price x at time $(N-j)\Delta$. Therefore

$$v(x,j) = e^{-r\Delta}\mathcal{P}_\Omega\big(f(-x,\Delta)*v(x,j-1)\big) = e^{-r\Delta}\mathcal{P}_\Omega\mathcal{F}^{-1}_{\xi\to x}\big(\Psi^*(\xi,\Delta)\widehat{v}(\xi,j-1)\big), \tag{2}$$

where we recall that $*$ is the convolution operator and $\mathcal{P}_\Omega$ is the projector operator on $\Omega := (l,+\infty)$, i.e., $\mathcal{P}_\Omega f(x) = \mathbf{1}_{x\in\Omega} f(x)$. The indicator function $\mathbf{1}_{x\in\Omega}$ can be replaced by the Heaviside step function centered on l: it is 1 if $x > l$ and 0 if $x < l$, while for $x = l$ it can be assigned the values 0 (left-continuous choice), 1 (right-continuous choice) or 1/2 (symmetric choice). The value for $x = l$ matters only from a numerical point of view, as the measure of this point is zero.

At each time step the convolution method proceeds by moving from the real to the Fourier space and backward trough the iteration

$$v_{j-1} \xrightarrow{\mathcal{F}} \widehat{v}_{j-1} \xrightarrow{*} \Psi^*\widehat{v}_{j-1} \xrightarrow{\mathcal{P}\mathcal{F}^{-1}} v_j, \quad j=1,\dots,N.$$

This method has been used, among others, by Jackson et al. [9] and Lord et al. [10]. Lord et al. improved this numerical methods in order to have a monotonic convergence to zero of the discretization error.

The method of Feng and Linetsky [3] is based on the Hilbert transform, Equation (26) in the article. In fact, considering the generalized Plemelj-Sokhotsky relation

$$\mathcal{F}\mathcal{P}_\Omega h = \frac{1}{2}\big[\mathcal{F}h + ie^{i\xi l}\mathcal{H}_\xi(e^{-i\xi l}\mathcal{F}h)\big],$$

the Fourier transform of Equation (2) yields

$$\widehat{v}(\xi,j) = \frac{1}{2}e^{-r\Delta}(\Psi^*(\xi,\Delta)\widehat{v}(\xi,j-1) + ie^{i\xi l}\mathcal{H}_\xi(e^{-i\xi l}\Psi^*(\xi,\Delta)\widehat{v}(\xi,j-1))).$$

Thus all the computations are in Fourier space:

$$v_0 \xrightarrow{\mathcal{F}} \widehat{v}_0 \longrightarrow \ldots \longrightarrow \widehat{v}_{j-1} \xrightarrow{*} \Psi^* \widehat{v}_{j-1} \xrightarrow{\mathcal{H}} \widehat{v}_j \longrightarrow \ldots \longrightarrow \widehat{v}_N \xrightarrow{\mathcal{F}^{-1}} v_N.$$

The Hilbert transform is computed in Fourier space via a sinc function expansion which provides an exponentially decaying error, as explained in Section 3.1 of the article. Therefore the Hilbert method is preferable to the convolution approach. The computational cost of both methods is $\mathcal{O}(NM \log M)$.

C.2. Quadrature methods

The recursion given by Equation (1) has been solved using quadrature [4, 7]. If the domain is truncated as in Fusai et al. [6], the quadrature nodes are $x_i,\ i = 1, \ldots, M$, $\mathbf{K}$ is an $M \times M$ square matrix with elements $K_{ij} = e^{-r\Delta} f(x_j - x_i, \Delta)$, $\mathbf{D}$ is an $M \times M$ diagonal matrix which contains the quadrature weights, and $(\mathbf{v}_j)_i = v(x_i, j),\ j = 0, \ldots, N$, then Equation (1) becomes

$$\mathbf{v}_j = \mathbf{K}\mathbf{D}\mathbf{v}_{j-1} \tag{3}$$

for $j = 1, \ldots, N$. Thus, in order to compute the option price, one only has to perform N matrix-vector multiplications.

This approach can be efficiently implemented using the FFT, provided Newton-Cotes quadrature rules are considered. In fact, if the quadrature formula is characterized by equidistant nodes, $\mathbf{K}$ is a Toeplitz matrix and the matrix-vector multiplication in Equation (3) can be performed using the FFT as follows.

We recall that an $M \times M$ Toeplitz matrix $\mathbf{T}$ can be embedded in a $2M \times 2M$ circulant matrix $\mathbf{C}$, i.e. a special kind of Toeplitz matrix where each row vector is rotated one element to the right relative to the preceding row vector. Thus, given an $M \times 1$ vector $\mathbf{x}$, we can compute the component i of $\mathbf{Tx}$, $i = 1, \ldots, M$, as

$$(\mathbf{Tx})_i = \left(\mathrm{FFT}^{-1}\left(\mathrm{FFT}(\mathbf{c})\,\mathrm{FFT}(\mathbf{x}^*)\right)\right)_i,$$

$\mathbf{c}$ being the first column of the circulant matrix $\mathbf{C}$ and $\mathbf{x}^*$ being the extension of the vector $\mathbf{x}$ obtained padding $\mathbf{x}$ with M zeros. Thus Equation (3) becomes

$$(\mathbf{v}_j)_i = \left(\mathrm{FFT}^{-1}\left(\mathrm{FFT}(\mathbf{c})\,\mathrm{FFT}\left((\mathbf{D}\mathbf{v}_{j-1})^*\right)\right)\right)_i,$$

$i = 1, \ldots, M$, $\mathbf{c}$ being the first column of the circulant matrix embedding $\mathbf{K}$. Since $(\mathbf{K})_{i,j} = e^{-r\Delta} f(x_j - x_i, \Delta) = e^{-r\Delta} f((j-i)h, \Delta)$, h being the distance between the quadrature nodes, and f is computed with an inverse Fourier transform of the characteristic function Ψ, it follows that $\widehat{\mathbf{c}} := \mathrm{FFT}(\mathbf{c})$

can be computed directly by using Ψ, avoiding one FFT. At the end the computational cost of this pricing procedure becomes $2NM \log M$, since for each iteration of the pricing recursion we have to compute one FFT and one inverse FFT. We also have to compute the matrix-vector multiplication $\mathbf{D}\mathbf{v}_{j-1}$, but as $\mathbf{D}$ is a diagonal matrix, the computational cost consists of M operations. Thus the scheme of the quadrature-FFT based approach is

$$\mathbf{v}_{j-1} \longrightarrow \mathbf{D}\mathbf{v}_{j-1} \xrightarrow{\mathcal{F}} \mathcal{F}[\mathbf{D}\mathbf{v}_{j-1}] \xrightarrow{*} \widehat{\mathbf{c}}\mathcal{F}[\mathbf{D}\mathbf{v}_{j-1}] \xrightarrow{\mathcal{F}^{-1}} \mathbf{v}_j.$$

C.3. The Z-WH algorithm

Another approach consists in relating the pricing problem to the solution of an integral equation [6]. After applying the z-transform to Equation (1), i.e., multiplying both sides by q^j and summing over $j \geq 1$, it is shown that $\widetilde{v}(x, q)$ solves the Wiener-Hopf integral equation

$$\widetilde{v}(x, q) = qe^{-r\Delta} \int_l^{+\infty} f_b(x - x', \Delta)\widetilde{v}(x', q)dx' + \phi(x) \quad \text{for } x \geq l. \tag{4}$$

Two solution strategies are possible to solve the latter. The first [5, 6] consists in applying a quadrature scheme to Equation (4) and therefore it reduces to solve the linear system

$$(\mathbf{I} - q\mathbf{K}\mathbf{D})\widetilde{\mathbf{v}} = \mathbf{g}$$

with parameter q, $\mathbf{I}$ being the $M \times M$ identity matrix, before inverting the z-transform to obtain the option price. This approach was introduced by Fusai et al. [6], who presented a numerical scheme based on a preconditioning technique to speed up the solution of the linear system: in fact, considering Newton-Cotes quadrature rules and an iterative linear system solution method, like the generalized minimal residual (GMRes) method, the authors provided an FFT-based method whose computational cost is $\mathcal{O}(\min\{N, m_{\mathrm{E}} + n_{\mathrm{E}}\} IM \log M)$, M being the number of quadrature nodes and I denoting the average number of GMRes iterations necessary to solve a linear system. The authors showed that the scheme provides a great accuracy at a low computational cost if the matrix $\mathbf{D}$ is nearly equal to $c\mathbf{I}$, for any constant c: in fact only in this case I is independent on the number of monitoring dates. This is true for the trapezoidal rule $\mathrm{diag}\,(\mathbf{D}) = h[0.5, 1, 1, \ldots, 1, 0.5]$, but not, for example, for the Simpson rule.

Another possibility consists in relating the Spitzer-Wiener-Hopf factorization to the solution of the integral equations. Indeed, the well-known methodology to solve a Wiener-Hopf integral equation also requires the knowledge of the Wiener-Hopf factors. More precisely, the main steps for solving the Wiener-Hopf integral equation (4) are:

1. Factorize the function $L(\xi, q) := 1 - qe^{-r\Delta}\Psi^*(\xi, \Delta)$,

$$L(\xi, q) = L_+(\xi, q)L_-(\xi, q).$$

2. Given the payoff function $\phi(x)$, define $P(\xi, q) := e^{-il\xi}\widehat{\phi}(\xi)/L_-(\xi, q)$ and decompose it into components that are analytic in the appropriate complex half planes:

$$P(\xi, q) = P_+(\xi, q) + P_-(\xi, q).$$

3. The Fourier transform of the solution of the integral equation (4) is now given by

$$\widetilde{\widehat{v}}(\xi, q) = e^{il\xi}\frac{P_+(\xi, q)}{L_+(\xi, q)}. \tag{5}$$

Therefore, the following pricing methodology works for a number of monitoring dates $N > 2$:

1. Compute the value of $\widehat{v}(\xi, 1)$ by convolution, i.e.,

$$\widehat{v}(\xi, 1) = \Psi^*(\xi, \Delta)\widehat{\phi}(\xi). \tag{6}$$

2. Compute $\widehat{v}(\xi, N-1)$, i.e., consider an option with $N-2$ monitoring dates and payoff $\phi(x) = \mathcal{F}^{-1}_{\xi\to x}\widehat{v}(\xi, 1)$, whose price is $v(\cdot, N-1)$, with

$$\widehat{v}(\xi, N-1) = \mathcal{Z}^{-1}_{q\to N-2}\left[\widetilde{\widehat{v}}(\xi, q)\right],$$

solving the Wiener-Hopf integral equations using the factorization to obtain $\widetilde{v}(\xi, q)$ for the different values of q necessary to invert the z-transform.
3. Compute the value of $\widehat{v}(x, N)$ by convolution, as in Equation (6).
4. Apply an inverse Fourier transform to obtain the option price $v(x, N)$.

All the computations are performed in Fourier space:

$$v_0 \xrightarrow{\mathcal{F}} \widehat{v}_0 \xrightarrow{\Psi^*} \widehat{v}_1 \xrightarrow{\mathcal{ZWH}} \widehat{v}_{N-1} \xrightarrow{\Psi^*} \widehat{v}_N \xrightarrow{\mathcal{F}^{-1}} v_N,$$

where $\mathcal{ZWH}$ stands for the second step of the above algorithm. As in the algorithm described in the article, Section 4.2, Steps 1 and 3 are necessary in order to smooth the tails of the payoff and of the inverse of the z-transform in Fourier space, $\widehat{v}(x, N-1)$, before applying the z-transform (Step 2) and the inverse Fourier transform (Step 4), respectively, and thus to obtain an exponential convergence considering the Wiener-Hopf factorization described in Section 3.1 of the article. We would like to stress that $\phi(x) = \mathcal{F}^{-1}_{\xi\to x}\widehat{v}(x,1)$ differs from $v(x,1)$ because of a projection, i.e., $v(x,1) = \mathcal{P}_{\{x>l\}}\mathcal{F}^{-1}_{\xi\to x}\widehat{v}(\xi,1)$.

In the case of double-barrier options, the Wiener-Hopf equation becomes a Fredholm equation of the second type with a convolution kernel. This problem is very old, but up to now no efficient and accurate procedure has been devised for its solution. So the scheme here presented deserves some interest on its own. More precisely, the pricing equation becomes

$$\widetilde{v}(x,q)(x) = qe^{-r\Delta}\int_l^u f_b(x-x',\Delta)\widetilde{v}(x',q)dx' + \phi(x), \tag{7}$$

and this can be solved using a fixed-point algorithm similar to the one presented in Section 4.3 of the article, with Equations (40)–(41) in the article replaced by

$$\frac{J_-(\xi,q)}{L_-(\xi,q)} = \left[\frac{e^{-il\xi}\Psi^*(\xi,\Delta)\widehat{\phi}(\xi) - e^{i(u-l)\xi}J_+(\xi,q)}{L_-(\xi,q)}\right]_-, \tag{8}$$

$$\frac{J_+(\xi,q)}{L_+(\xi,q)} = \left[\frac{e^{-iu\xi}\Psi^*(\xi,\Delta)\widehat{\phi}(\xi) - e^{i(l-u)\xi}J_-(\xi,q)}{L_+(\xi,q)}\right]_+. \tag{9}$$

Once $J_\pm$ are computed via the fixed-point algorithm described in Section 4.3 of the article, the option price is given by

$$v(x,N) = e^{-rT}\mathcal{F}^{-1}_{\xi\to x}\Big[\Psi^*(\xi,\Delta) \\ \mathcal{Z}^{-1}_{q\to N-2}\left[\frac{\widehat{\phi}(\xi)\Psi^*(\xi,\Delta)}{L(\xi,q)} - e^{il\xi}\frac{J_-(\xi,q)}{L(\xi,q)} - e^{iu\xi}\frac{J_+(\xi,q)}{L(\xi,q)}\right]\Big]. \tag{10}$$

Thus the pricing algorithm consists of the steps

$$\phi \equiv v_0 \xrightarrow{\mathcal{F}} \widehat{v}_0 \xrightarrow{\Psi^*} \widehat{v}_1 \xrightarrow{\mathcal{ZWH}} \widehat{v}_{N-1} \xrightarrow{\Psi^*} \widehat{v}_N \xrightarrow{\mathcal{F}^{-1}} v_N,$$

where here we denote with $\mathcal{ZWH}$ the operator

$$\mathcal{ZWH} : F(\xi) \to \mathcal{Z}^{-1}_{q\to N-2}\left[\frac{F(\xi) - e^{il\xi}J_-(\xi,q) - e^{iu\xi}J_+(\xi,q)}{L(\xi,q)}\right].$$

Fusai et al. [6] provided a pricing procedure based on Wiener-Hopf integral equations also for lookback options: therefore the Z-QUAD and Z-WH method can be applied to this class of contracts too. However, with respect to barrier options, a further step must be applied to compute the option price, since an integral needs to be evaluated once the Wiener-Hopf integral equations are solved Fusai et al. [6, Section 3.3.2, Step (6) of the algorithm]. As shown in Section 5 of our article, the Z-WH and the Z-S algorithms exhibit the same accuracy dealing with barrier options. Numerical results not reported here show that this is no more true for lookback options, since the further integral to be evaluated makes the Z-WH algorithm uncompetitive with respect to the Z-S one both in terms of computational cost and accuracy.

D. Continuous versus discrete monitoring

As mentioned in Section 2 of the article, identities similar to Equations (11)–(16) in the article exist for continuous monitoring too. The discrete minimum and maximum operators are replaced with their continuous versions,

$$M_T^{\mathrm{c}} = \sup_{t\in(0,T)} X(t) \quad \text{and} \quad m_T^{\mathrm{c}} = \inf_{t\in(0,T)} X(t).$$

In this case the z-transform is replaced by the Laplace transform, while the quantity to be factorized is obtained setting $q = e^{-s\Delta}$ and taking the limit $\Delta \to 0$ [8, Section 4.1.2]:

$$\lim_{\Delta\to 0} \frac{\Phi(\xi,q)}{\Delta} = \lim_{\Delta\to 0} \frac{1-q\Psi(\xi,\Delta)}{\Delta} = \lim_{\Delta\to 0} \frac{1-e^{(\psi(\xi)-s)\Delta}}{\Delta} = s-\psi(\xi) =: \phi(\xi,s).$$

Similar limits hold for the Wiener-Hopf factors of Φ and ϕ:

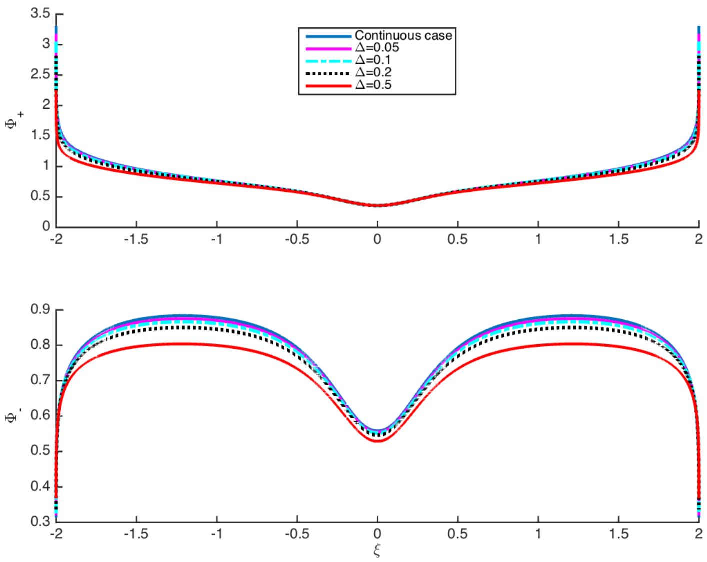
$$\lim_{\Delta\to 0} \frac{\Phi_\pm(\xi,q)}{\sqrt{\Delta}} = \phi_\pm(\xi,s).$$

Remarkably, the Wiener-Hopf factorization of $\phi(\xi,s)$ is not obtained through a passage to the limit of the Wiener-Hopf factorization of $\Phi(\xi,q)$, but directly from ϕ itself using the Hilbert transform like for the factorization of Φ. Therefore, our procedure can price contracts in the continuous monitoring case too, once the inverse z-transform is replaced with the inverse Laplace transform. To show that the Wiener-Hopf factorization of both the discrete and continuous case can be computed with our procedure, in Figure D we consider a double exponential distribution, whose characteristic exponent is

$$\psi(\xi) = i\gamma\xi - \frac{1}{2}\sigma^2\xi^2 + \eta\left(p\frac{\eta_1}{\eta_1+i\xi} + (1-p)\frac{\eta_2}{\eta_2-i\xi} - 1\right).$$

We set $\gamma = 0.2$, $\sigma = 0.2$, $\eta = 0.5$, $p = 0.5$, $\eta_1 = 0.4$, $\eta_2 = 0.4$ and $s = 0.2$, and we plot $\phi_\pm(\xi,s)$ as well as $\Phi_\pm(\xi,q)/\sqrt{\Delta}$ for different values of $\Delta \to 0$, showing numerically the convergence of the latter to the former. The method by Feng and Linetsky, as well as all the other methods described in

Figure D: Convergence of $\Phi_+(\xi, q)/\sqrt{\Delta}$ to $\phi_+(\xi, s)$ (top) and of $\Phi_-(\xi, q)/\sqrt{\Delta}$ to $\phi_-(\xi, s)$ (bottom), $\xi \in [-2, 2]$.



Section C, can deal only with the discrete monitoring case. In these cases, the continuous monitoring value can be obtained only through a passage to the limit, but it is well known that the convergence is slow. This clarifies the importance of an efficient numerical method able to deal with both the discrete and continuous monitoring cases. The methodology proposed here factorizes directly $\phi(\xi, s) = s - \psi(\xi)$ and is exempt from the problem of the slow convergence from discrete to continuous monitoring.

References

[1] Andricopoulos, A. D., M. Widdicks, D. P. Newton, P. W. Duck. 2007. Extending quadrature methods to value multi-asset and complex path dependent options. *Journal of Financial Economics* **83**(2) 471–499.

[2] Fang, F., C. W. Oosterlee. 2009. Pricing early-exercise and discrete barrier options by Fourier-cosine series expansions. *Numerische Mathematik* **114**(1) 27–62.

[3] Feng, L., V. Linetsky. 2008. Pricing discretely monitored barrier options

and defaultable bonds in Lévy process models: a Hilbert transform approach. *Mathematical Finance* **18**(3) 337–384.

[4] Fusai, G., G. Longo, M. Marena, M. C. Recchioni. 2009. Lévy processes and option pricing by recursive quadrature. C. W. Hurlington, ed., *Economic Dynamics: Theory, Games and Empirical Studies*. Nova Publishers, New York, 31–57.

[5] Fusai, G., D. Marazzina, M. Marena. 2010. Option pricing, maturity randomization and distributed computing. *Parallel Computing* **36**(7) 403–414.

[6] Fusai, G., D. Marazzina, M. Marena, M. Ng. 2012. z-transform and preconditioning techniques for option pricing. *Quantitative Finance* **12**(9) 1381–1394.

[7] Fusai, G., M. C. Recchioni. 2007. Analysis of quadrature methods for pricing discrete barrier options. *Journal of Economic Dynamics and Control* **31**(3) 826–860.

[8] Green, R., G. Fusai, I. D. Abrahams. 2010. The Wiener-Hopf technique and discretely monitored path-dependent option pricing. *Mathematical Finance* **20**(2) 259–288.

[9] Jackson, K., S. Jaimungal, V. Surkov. 2008. Fourier space time-stepping for option pricing with Lévy models. *Journal of Computational Finance* **12**(2) 1–29.

[10] Lord, R., F. Fang, F. Bervoets, C. W. Oosterlee. 2008. A fast and accurate FFT-based method for pricing early-exercise options under Lévy processes. *SIAM Journal on Scientific Computing* **30** 1678–1705.