Asian SWIFT method
Efficient wavelet-based valuation of arithmetic Asian options

Álvaro Leitao, Luis Ortiz-Gracia and Emma I. Wagner

Seminar - UB

December 19, 2018
Motivation

- Arithmetic Asian options are still attractive in financial markets, but their numerical treatment is rather challenging.
- The valuation methods relying on Fourier inversion are highly appreciated, particularly for calibration purposes, since they are extremely fast, very accurate, and easy to implement.
- Lack of robustness in the existing methods (number of terms in the expansion, numerical quadratures, truncation, etc.).
- The use of wavelets for other option problems (Europeans, early-exercise, etc.) has resulted in significant improvements in this sense.
- In the context of arithmetic Asian options, SWIFT provides extra benefits.
Outline

1. Definitions
2. Problem formulation
3. The SWIFT method
4. SWIFT for Asian options
5. Numerical results
6. Conclusions
Option

A contract that offers the buyer the right, but not the obligation, to buy (call) or sell (put) a financial asset at an agreed-upon price (the strike price) during a certain period of time or on a specific date (exercise date). Investopedia.

Option price

The fair value to enter in the option contract. In other (mathematical) words, the (discounted) expected value of the contract.

\[ v(S, t) = D(t) \mathbb{E} [\mathcal{P}(S(t))] \]

where \( \mathcal{P} \) is the payoff function, \( S \) the underlying asset, \( t \) the exercise time and \( D(t) \) the discount factor.
Definitions (II)

### Pricing techniques
- Stochastic process, $S(t)$, governed by a SDE.
- Underlying models: Black-Scholes, Lévy-based, Heston, etc.
- Simulation: Monte Carlo method.
- PDEs: Feynman-Kac theorem.
- Fourier inversion techniques: characteristic function.

### Types of options - payoff function
- Vanilla: involves only the value of $S$ at exercise. Standardized.
- Exotic: involves more complicated features. Over the counter.
- Path-dependent: Asian, Barrier, Lookback, ...
Problem formulation

- In Asian derivatives, the option payoff function relies on some *average* of the underlying values at a prescribed monitoring dates.
- Thus, the final value is less volatile and the option price cheaper.
- Consider \( N + 1 \) monitoring dates \( t_i \in [0, T], i = 0, \ldots, N \).
- Where \( T \) is the maturity and \( \Delta t := t_{i+1} - t_i, \forall i \) (equal-spaced).
- Assume the initial state of the price process to be known, \( S(0) = S_0 \).
- Let averaged price be defined as \( A_N := \frac{1}{N+1} \sum_{i=0}^{N} S(t_i) \), the payoff of the *European-style* Asian call option is
  \[
  v(S, T) = (A_N - K)^+. 
  \]
- The risk-neutral option valuation formula,
  \[
  v(x, t) = e^{-r(T-t)} \mathbb{E} [v(y, T)|x] = e^{-r(T-t)} \int_{\mathbb{R}} v(y, T) f(y|x) dy,
  \]
  with \( r \) the risk-free rate, \( T \) the maturity, \( f(y|x) \) the transitional density, typically unknown, and \( v(y, T) \) the payoff function.
A structure for wavelets in $L^2(\mathbb{R})$ is called a multi-resolution analysis. We start with a family of closed nested subspaces in $L^2(\mathbb{R})$,

$$\ldots \subset \mathcal{V}_{-1} \subset \mathcal{V}_0 \subset \mathcal{V}_1 \subset \ldots, \quad \bigcap_{m \in \mathbb{Z}} \mathcal{V}_m = \{0\}, \quad \bigcup_{m \in \mathbb{Z}} \mathcal{V}_m = L^2(\mathbb{R}),$$

where

$$f(x) \in \mathcal{V}_m \iff f(2x) \in \mathcal{V}_{m+1}.$$  

Then, it exists a function $\varphi \in \mathcal{V}_0$ generating an orthonormal basis, denoted by $\{\varphi_{m,k}\}_{k \in \mathbb{Z}}$, for each $\mathcal{V}_m$, $\varphi_{m,k}(x) = 2^{m/2} \varphi(2^m x - k)$.

The function $\varphi$ is called the scaling function or father wavelet.

For any $f \in L^2(\mathbb{R})$, a projection map of $L^2(\mathbb{R})$ onto $\mathcal{V}_m$, denoted by $\mathcal{P}_m : L^2(\mathbb{R}) \rightarrow \mathcal{V}_m$, is defined by means of

$$\mathcal{P}_m f(x) = \sum_{k \in \mathbb{Z}} c_{m,k} \varphi_{m,k}(x), \quad \text{with} \quad c_{m,k} = \langle f, \varphi_{m,k} \rangle.$$
The SWIFT method

- In this work, we employ Shannon wavelets. A set of Shannon scaling functions $\varphi_{m,k}$ in the subspace $\mathcal{V}_m$ is defined as,

$$
\varphi_{m,k}(x) = 2^{m/2} \frac{\sin(\pi(2^mx - k))}{\pi(2^mx - k)} = 2^{m/2} \varphi(2^mx - k), \quad k \in \mathbb{Z},
$$

where $\varphi(z) = \text{sinc}(z)$, with $\text{sinc}$ the cardinal sine function.

- Given a function $f \in L^2(\mathbb{R})$, we will consider its expansion in terms of Shannon scaling functions at the level of resolution $m$. 
The SWIFT method

- Our aim is to recover the coefficients $c_{m,k}$ of this approximation from the Fourier transform of the function $f$, denoted by $\hat{f}$, defined as
  \[ \hat{f}(\xi) = \int_\mathbb{R} e^{-i\xi x} f(x) dx, \]
  where $i$ is the imaginary unit.

- Following wavelets theory, a function $f \in L^2(\mathbb{R})$ can be approximated at the level of resolution $m$ by,
  \[ f(x) \approx P_m f(x) = \sum_{k \in \mathbb{Z}} c_{m,k} \varphi_{m,k}(x), \]
  where $P_m f$ converges to $f$ in $L^2(\mathbb{R})$, i.e. $\| f - P_m f \|_2 \to 0$, $m \to +\infty$.

- The infinite series is well-approximated (see Lemma 1 of [3]) by
  \[ P_m f(x) \approx f_m(x) := \sum_{k=k_1}^{k_2} c_{m,k} \varphi_{m,k}(x), \]
  for certain accurately chosen values $k_1$ and $k_2$. 
The SWIFT method

- Computation of the coefficients $c_{m,k}$: by definition,

$$c_{m,k} = \langle f, \varphi_{m,k} \rangle = \int_{\mathbb{R}} f(x) \bar{\varphi}_{m,k}(x) dx = 2^{m/2} \int_{\mathbb{R}} f(x) \varphi(2^m x - k) dx.$$  

- By using the classical Vieta’s formula,

$$\varphi(x) = \text{sinc}(x) = \prod_{j=1}^{+\infty} \cos \left( \frac{\pi x}{2j} \right).$$

- We truncate the infinite product into a finite product with $J$ terms, then, thanks to the cosine product-to-sum identity,

$$\prod_{j=1}^{J} \cos \left( \frac{\pi x}{2j} \right) = \frac{1}{2^{J-1}} \sum_{j=1}^{2^{J-1}} \cos \left( \frac{2j - 1}{2^J} \pi x \right).$$

- Then,

$$\int_{\mathbb{R}} f(x) \bar{\varphi}_{m,k}(x) dx \approx \frac{2^{m/2}}{2^{J-1}} \sum_{i=-1}^{2^{J-1}} \int_{\mathbb{R}} f(x) \cos \left( \frac{2j - 1}{2^J} \pi (2^m x - k) \right) dx.$$
The SWIFT method

- Noting that $\Re \left( \hat{f}(\xi) \right) = \int_{\mathbb{R}} f(x) \cos(\xi x) dx$ and
  
  $$\hat{f}(\xi)e^{ik\pi \frac{2j-1}{2^J}} = \int_{\mathbb{R}} e^{-i(\xi x - k\pi \frac{2j-1}{2^J})} f(x) dx.$$

- Thus, we have,
  
  $$c_{m,k} \approx \frac{2^m}{2^{J-1}} \sum_{j=1}^{2^{J-1}} \Re \left[ \hat{f} \left( \frac{(2j - 1)\pi 2^m}{2^J} \right) e^{i k\pi (2j-1) / 2^J} \right].$$

- Putting everything together gives the following approximation of $f$,
  
  $$f(x) \approx \sum_{k=k_1}^{k_2} c_{m,k} \varphi_{m,k}(x).$$
SWIFT option valuation formulas

- Truncating the integration range on \([a, b]\) in the risk-neutral valuation formula, and replacing density \(f\) by the SWIFT approximation,

\[
v(x, t_0) \approx e^{-rT} \sum_{k=k_1}^{k_2} c_{m,k} V_{m,k},
\]

where,

\[
V_{m,k} := \int_{a}^{b} v(y, T) \varphi_{m,k}(y|x) dy.
\]

- By employing the Vieta’s formula again and interchanging summation and integration operations, we obtain that

\[
V_{m,k} \approx \frac{2^{m/2}}{2^{J-1}} \sum_{j=1}^{2^{J-1}} \int_{a}^{b} v(y, T) \cos \left( \frac{2j - 1}{2^{j}} \pi \left( 2^{m} y - k \right) \right) dy.
\]
SWIFT option sensitivities

- Under the SWIFT framework, the estimation of the option price sensitivities, the so-called Greeks.
- The Greeks are defined as the partial derivatives of the option price with respect to some market/model parameter.
- They can be efficiently calculated by constructing similar series expansions.
- Generally, two possible situations can appear: the option price depends only on the parameter of interest either through the density function or payoff function.
- The partial derivative of the characteristic function and, hence, the density coefficients and the payoff function can be analytically computed for many financial models and option contracts.
SWIFT option sensitivities

- We firstly assume that the option price depends on the parameter of interest only through the density function,

\[ c_{m,k}(\xi, \varsigma) = \frac{2^{m/2}}{2^{J-1}} \sum_{j=1}^{2^{J-1}} \Re \left[ \hat{f}(\xi; \varsigma) e^{i k_\xi} \right], \]

where \( \xi = \frac{(2j-1)\pi 2^m}{2^j} \) and \( \varsigma \) the parameter of interest.

- By differentiating \( n \) times the characteristic function, the “Greek” density coefficients

\[ c_{m,k}^{(n)}(\xi) := \frac{\partial^n c_{m,k}(\xi, \varsigma)}{\partial \varsigma^n} = \frac{2^{m/2}}{2^{J-1}} \sum_{j=1}^{2^{J-1}} \Re \left[ \frac{\partial^n \hat{f}(\xi; \varsigma)}{\partial \varsigma^n} e^{i k_\xi} \right]. \]

- For example, the so-called Delta, \( \Delta \), and Gamma, \( \Gamma \), the first and second derivatives w.r.t. \( S_0 \), are computed by plugging the \( c_{m,k}^{(n)} \),

\[ \Delta := e^{-rT} \sum_{k} c_{m,k}^{(1)} V_{m,k}, \quad \Gamma := e^{-rT} \sum_{k} c_{m,k}^{(2)} V_{m,k}. \]
SWIFT option sensitivities

- A second possible situation appears when the option value depends on the parameter of interest, $\zeta$, through the payoff coefficients, i.e., $V_{m,k}(\zeta)$.
- Thus, the “Greek” payoff coefficients need to be determined by differentiating $V_{m,k}$ with respect to $\zeta$.
- Particularly, the solution for the Greeks $\Delta$ and $\Gamma$ would be

$$
\Delta := e^{-rT} \sum_{k=k_1}^{k_2} c_{m,k} V_{m,k}^{(1)}(\zeta), \quad \Gamma := e^{-rT} \sum_{k=k_1}^{k_2} c_{m,k} V_{m,k}^{(2)}(\zeta),
$$

where now the $c_{m,k}$ are kept invariant and $V_{m,k}^{(n)}$ represents the $n$-th derivative of $V_{m,k}$.
- In the context of Fourier inversion techniques, closed-form solutions for these coefficients can be usually derived.
- The case of the arithmetic Asian payoff will be addressed in the next section.
The quality in the approximation provided by the SWIFT method is affected by the scale $m$, the number of terms in the Vieta’s approximation and the series truncation limits, $k_1$ and $k_2$.

By Lemma 3 of [2], the error in the projection approximation of function $f$ is bounded by

$$|f(x) - \mathcal{P}_m f(x)| \leq \frac{1}{2\pi} \int_{|\xi|>2m\pi} \left| \hat{f}(\xi) \right| d\xi.$$  

As the characteristic function, $\hat{f}$, is assumed to be known, we can compute $m$ given a prescribed tolerance $\epsilon_m$.

Applying a simple quadrature rule, the error bound reads

$$\frac{1}{2\pi} \left( \left| \hat{f}(-2^m\pi) \right| + \left| \hat{f}(2^m\pi) \right| \right).$$

More involved numerical quadratures have been tested, but the observed differences are negligible.
Optimal scale $m$, series bounds $k_1$ and $k_2$, and parameter $J$

- $k_1$ and $k_2$ can be computed based on the integration range $[a, b]$ as
  \[
  k_1 := \left\lfloor 2^m a \right\rfloor \quad \text{and} \quad k_2 := \left\lceil 2^m b \right\rceil,
  \]
  where $m$ is the scale of approximation.
- Therefore we first need to choose the interval limits, $a$ and $b$, in such a way that the loss of density mass is minimized.
- Cumulants-based approach,

  \[
  [a, b] := \left[ \kappa_1(Y) - L \sqrt{\kappa_2(Y) + \sqrt{\kappa_4(Y)}}, \kappa_1(Y) + L \sqrt{\kappa_2(Y) + \sqrt{\kappa_4(Y)}} \right],
  \]

  with $\kappa_n(Y)$ representing the $n$-th cumulant (defined from the cumulant-generating function, $\mathcal{K}(\tau)$, as $\kappa_n = \mathcal{K}^{(n)}(0)$) of the random variable $Y$ and $L$ a constant conveniently chosen.
The dependence on $m$ turns out to be very convenient also in the selection of the interval $[a, b]$.

This constitutes one of the great advantages of the SWIFT method with respect to other Fourier inversion-based techniques, where $a$ and $b$ are arbitrarily selected.

Thus, as we know that our approximation at scale $m$ satisfies the tolerance $\epsilon_m$, the error order due to the truncation should not exceed the order of $\epsilon_m$.

We can therefore develop an adaptive interval selection algorithm that updates the truncated range $[a, b]$ in each iteration, computes the truncation error, $\epsilon_\tau$, in the approximated density using that interval and stops when the same tolerance condition $\epsilon_m$ is prescribed.
The parameter $J$ is then chosen to be constant (it could be selected as a function of $k$) based on the previously determined quantities.

Doing so, we can benefit from the use of FFT algorithm.

By Theorem 1 of [3], let $c_{m,k}^*$ the approximated coefficients,

$$|c_{m,k} - c_{m,k}^*| \leq 2^{m/2} \left( 2\epsilon + \sqrt{2A} \|f\|_2 \frac{ (\pi M_{m,k})^2 }{ 2^{2(J+1)} - (\pi M_{m,k})^2 } \right),$$

assuming $J \geq \log_2(\pi M_{m,k})$, and with $M_{m,k} := \max(|2^m A - k|, |2^m A + k|), A := \max(|a|, |b|)$, $H(x) = F(-x) + 1 - F(x)$ and $H(A) < \epsilon$.

Thus, the number of Vieta factor is selected as

$$J := \lceil \log (\pi M_m) \rceil \quad \text{with} \quad M_m := \max_{k_1 < k < k_2} M_{m,k},$$
Exponential Lévy models: \( \log S(t) \), follows a Lévy process.

The Lévy dynamics have a stationary and i.i.d. increments and it can be written in the form

\[
X(t) = \mu t + W(t) + J(t) + \lim_{\varepsilon \to 0} D^\varepsilon(t),
\]

where \( W \) is a \( d \)-dimensional Brownian motion with covariance matrix \( \Sigma \), drift vector \( \mu \in \mathbb{R}^d \), \( J \) is a compound Poisson process and \( D^\varepsilon \) is a compensated compound Poisson process. A measure \( \nu \) on \( \mathbb{R}^d \) is adopted, called Lévy measure.

The Lévy processes are fully determined by the characteristic triplet \( [\Sigma, \mu, \nu] \). From the Lévy-Khintchine formula, the characteristic function, defined as \( \hat{f}(\xi) = \mathbb{E}\left[ e^{i\xi X(t)} \right] \), reads

\[
\hat{f}(\xi) = e^{t\vartheta(\xi)}, \quad \vartheta(\xi) = i\mu \cdot \xi + \frac{1}{2} \Sigma \xi \cdot \xi + \int_{\mathbb{R}^d} \left( e^{i\xi \cdot x} - 1 - i\xi \cdot x1_{|x|\leq 1} \right) \nu(dx),
\]

where \( \vartheta \) is often called the characteristic exponent.
The explicit representation of the characteristic function in the Lévy processes framework supposes a great advantage. Allows to recover the density, \( f \), by Fourier inversion numerical techniques and price European options highly efficiently. The characteristic function of exponential Lévy dynamics is often available in a tractable form (ex. Black-Scholes, Merton, Variance Gamma (VG), Normal Inverse Gaussian (NIG)). But, for arithmetic Asian options, the derivation of the corresponding characteristic function is rather involved. 

Let's start by defining the return or increment process \( R_i \),

\[
R_i := \log \left( \frac{S(t_i)}{S(t_{i-1})} \right) \quad i = 1, \ldots, N.
\]

Based on \( R_i \), we define a new process

\[
Y_i := R_{N+1-i} + Z_{i-1}, \quad i = 2, \ldots, N,
\]

where \( Y_1 = R_N \) and \( Z_i := \log (1 + e^{Y_i}) \), \( \forall i \).
Applying the Carverhill-Clewlow-Hodges factorization to $Y_i$,

$$
\frac{1}{N+1} \sum_{i=0}^{N} S(t_i) = \frac{(1 + e^{Y_N})}{N+1} S_0.
$$

Thus, the option price for arithmetic Asian contracts can be now expressed in terms of the transitional density of the $Y_N$ as

$$
\nu(x, t_0) = e^{-rT} \int_{\mathbb{R}} \nu(y, T) f_{Y_N}(y|x) dy,
$$

where $x = \log S_0$ and the call payoff function is given by

$$
\nu(y, T) = \left( \frac{S_0 (1 + e^y)}{N+1} - K \right)^+.
$$

Again, the probability density function $f_{Y_N}$ is generally not known, even for Lévy processes. However, as the process $Y_N$ is defined in a recursive manner, the characteristic function of $Y_N$ can be computed iteratively as well.
**Characteristic function of $Y_N$**

- By the definition of $Y_i$, the initial and recursive characteristic functions are
  \[ \hat{f}_{Y_1}(\xi) = \hat{f}_{R_N}(\xi) = \hat{f}_R(\xi), \]
  \[ \hat{f}_{Y_i}(\xi) = \hat{f}_{R_{N+1-i}+Z_{i-1}}(\xi) = \hat{f}_{R_{N+1-i}}(\xi) \cdot \hat{f}_{Z_{i-1}}(\xi) = \hat{f}_R(\xi) \cdot \hat{f}_{Z_{i-1}}(\xi). \]

- By definition, the characteristic function of $Z_{i-1}$ reads
  \[ \hat{f}_{Z_{i-1}}(\xi) := \mathbb{E} \left[ e^{-i\xi \log(1+e^{Y_{i-1}})} \right] = \int_{\mathbb{R}} (1 + e^{x})^{-i\xi} f_{Y_{i-1}}(x)dx. \]

- We can again apply the wavelet approximation to $f_{Y_{i-1}}$ as
  \[ \hat{f}_{Z_{i-1}}(\xi) \approx \int_{\mathbb{R}} (1 + e^{x})^{-i\xi} \sum_{k=k_1}^{k_2} c_{m,k} \varphi_{m,k}(x)dx \]
  \[ = 2^{m/2} \sum_{k=k_1}^{k_2} c_{m,k} \int_{\mathbb{R}} (e^{x} + 1)^{-i\xi} \text{sinc} (2^{m}x - k) dx. \]
Characteristic function of $Y_N$

- The integral on the right hand side needs to be computed efficiently to make the method easily implementable, robust and very fast.
- State-of-the-art methods from the literature rely on solving the integral by means of quadratures.

**Theorem (Theorem 1.3.2 of [4])**

Let $f$ be defined on $\mathbb{R}$ and let its Fourier transform $\hat{f}$ be such that for some positive constant $d$, $|\hat{f}(\omega)| = O\left(e^{-d|\omega|}\right)$ for $\omega \to \pm \infty$, then as $h \to 0$

$$\frac{1}{h} \int_{\mathbb{R}} f(x)S_{j,h}(x)dx - f(jh) = O\left(e^{-\frac{\pi d}{h}}\right),$$

where $S_{j,h}(x) = \text{sinc}\left(\frac{x}{h} - j\right)$ for $j \in \mathbb{Z}$.

- The theorem above allows us to approximate the integral above provided that $g(x) := (e^x + 1)^{-i\xi}$ satisfies the hypothesis.
If we consider $h = \frac{1}{2^m}$, then it follows from Theorem 1 that
\[
\int_{\mathbb{R}} g(x) \text{sinc} \left(2^m x - k\right) \, dx \approx hg \left(kh\right) = \frac{1}{2^m} \left(e^{\frac{k}{2^m}} + 1\right)^{-i\xi}.
\]

Thus, $\hat{f}_{Z_{i-1}}$ can be approximated by
\[
\hat{f}_{Z_{i-1}}(\xi) \approx 2^{-m/2} \sum_{k=k_1}^{k_2} c_{m,k} \left(e^{\frac{k}{2^m}} + 1\right)^{-i\xi}.
\]

Finally,
\[
\hat{f}_{Y_i}(\xi) = \hat{f}_R(\xi) \hat{f}_{Z_{i-1}} \approx \hat{f}_R(\xi) 2^{-m/2} \sum_{k=k_1}^{k_2} c_{m,k} \left(e^{\frac{k}{2^m}} + 1\right)^{-i\xi},
\]
where the density coefficients $c_{m,k}$ are computed as follows
\[
c_{m,k} \approx \frac{2^{m/2}}{2^{J-1}} \sum_{j=0}^{2^J - 1} \Re \left\{ \hat{f}_{Y_{i-1}} \left(\frac{(2j - 1)\pi 2^m}{2^J}\right) e^{\frac{ik\pi(2j-1)}{2^J}} \right\}.
\]
It remains to prove that function $g(x) = (e^x + 1)^{-i\xi}$ satisfies $|\hat{g}(\omega)| = \mathcal{O}(e^{-d|\omega|})$ for $\omega \to \pm\infty$.

We have derived an expression for $\hat{g}(\omega)$,

**Proposition**

Let $g(x) = (e^x + 1)^z$, where $z = -i\xi$ and $x, \xi \in \mathbb{R}$. Then,

$$\hat{g}(\omega) = \sum_{n=0}^{\infty} \binom{z}{n} \frac{2n - z}{(n - i\omega)(n + i(\omega + \xi))}, \quad \omega \in \mathbb{R}.$$
Proof expression $\hat{g}(\omega)$

**Proposition**

Let $z \in \mathbb{C}$ and $(\frac{z}{n}) = \frac{z(z-1)(z-2)\cdots(z-n+1)}{n!}$. Then the series $\sum_{n=0}^{\infty} \binom{z}{n} x^n$ converges to $(1 + x)^z$ for all complex $x$ with $|x| < 1$.

**Corollary**

Let $z \in \mathbb{C}$. Then the series $\sum_{n=0}^{\infty} \binom{z}{n} x^n y^{z-n}$ converges to $(x + y)^z$ for all complex $x, y$ with $|x| < |y|$.

**Proof.**

The proof follows from Proposition by taking into account that $(x + y)^z = \left(y \left[\frac{x}{y} + 1\right]\right)^z$. \qed
Proof.

From the definition, we split the integral in two parts

\[ \hat{g}(\omega) = \int_{\mathbb{R}} e^{-i\omega x} g(x) dx = \int_{-\infty}^{0} e^{-i\omega x} g(x) dx + \int_{0}^{\infty} e^{-i\omega x} g(x) dx, \]

and observe that, by Corollary above,

\[ (e^x + 1)^z = \sum_{n=0}^{\infty} \binom{Z}{n} e^{nx}, \text{ for } x < 0, \text{ and } (e^x + 1)^z = \sum_{n=0}^{\infty} \binom{Z}{n} e^{(z-n)x}, \text{ for } x > 0. \]

Replacing expressions and interchanging the integral and the sum, then we obtain,

\[ \hat{g}(\omega) = \sum_{n=0}^{\infty} \binom{Z}{n} \int_{-\infty}^{0} e^{-i\omega x} e^{nx} dx + \sum_{n=0}^{\infty} \binom{Z}{n} \int_{0}^{\infty} e^{-i\omega x} e^{(z-n)x} dx. \]

Finally, solving the integrals,

\[ \hat{g}(\omega) = \sum_{n=0}^{\infty} \binom{Z}{n} \frac{1}{n - i\omega} + \sum_{n=0}^{\infty} \binom{Z}{n} \frac{1}{n + i(\omega + \xi)} = \sum_{n=0}^{\infty} \binom{Z}{n} \frac{2n - z}{(n - i\omega)(n + i(\omega + \xi))}. \]
Proof expression $\hat{g}(\omega)$

- It is rather complicated to get a closed-form solution for the modulus of $\hat{g}(\omega)$ from this expression.
- By using Wolfram Mathematica 11.2, the infinite sum is written as

$$
\hat{g}(\omega) = \frac{\xi}{2\omega + \xi} \left[ e^{-\pi\omega} (B_{-1}(-i\omega, 1 + z) + 2B_{-1}(1 - i\omega, z)) + \\
+ \Gamma(i\omega - z) \left( 2(i\omega - z) \tilde{F}_1(1 - z, 1 + i\omega - z; 2 + i\omega - z; -1) - \\
- 2\tilde{F}_1(-z, i\omega - z; 1 + i\omega - z; -1) \right) \right],
$$

in terms of gamma, $\Gamma$, beta, $B$, and regularized hypergeometric, $\tilde{F}_1(a, b; c; \nu)$. (Modulus represented in next slide).
- The shape of $|\hat{g}(\omega)|$ does not depend on the value given to $\xi$.
- Different $\xi$ just originates a shift of the same function.
- The two peaks observed in the plot correspond to the poles of $\hat{g}(\omega)$ located at $\omega = 0$ and $\omega = -\xi$.
- $\hat{g}(\omega)$ presents a symmetry at $\omega = -\xi/2$ (it is straightforward to see that $\hat{g}(\omega - \xi/2) = \hat{g}(-\omega - \xi/2)$).
“Proof” modulus of $\hat{g}(\omega)$

- Representing $|\hat{g}(\omega)|$,

![Graph showing the modulus of $\hat{g}(\omega)$](image)

**Figure**: Modulus of $\hat{g}(\omega)$. 
The following theorem generalises the results stated in the previous Theorem. Thus, it can be applied under weaker conditions on the decay of $|\hat{g}(\omega)|$.

**Theorem**

Let $f$ be defined on $\mathbb{R}$ and let $\hat{f}$ be its Fourier transform. Then,

$$\left| \frac{1}{h} \int_{\mathbb{R}} f(x) S_{j,h}(x) dx - f(jh) \right| \leq \frac{1}{2\pi} \int_{|\omega| > \frac{\pi}{h}} |\hat{f}(\omega)| \, d\omega,$$

where $S_{j,h}(x) = \text{sinc} \left( \frac{x}{h} - j \right)$ for $j \in \mathbb{Z}$. 

A. Leitao & L. Ortiz-Gracia & E. Wagner

Asian SWIFT method

December 19, 2018 31 / 45
Proof.

As mentioned in Lemma 3 of [2], the approximation error $|f(x) - P_m f(x)|$ is uniformly bounded for all $x \in \mathbb{R}$,

$$|f(x) - P_m f(x)| \leq \frac{1}{2\pi} \int_{|\omega| > 2^m \pi} |\hat{f}(\omega)| \, d\omega,$$

where $P_m f(x) = \sum_{k \in \mathbb{Z}} c_{m,k} \varphi_{m,k}(x)$. In particular, this is valid for $x = jh$ with $h = 1/2^m$,

$$|f(jh) - P_m f(jh)| \leq \frac{1}{2\pi} \int_{|\omega| > 2^m \pi} |\hat{f}(\omega)| \, d\omega.$$

We observe that

$$P_m f(jh) = \sum_{k \in \mathbb{Z}} c_{m,k} \varphi_{m,k}(jh) = \sum_{k \in \mathbb{Z}} c_{m,k} 2^{m/2} \varphi(j - k),$$

where $\varphi(j - k) = \delta_{jk}$, and $\delta_{jk}$ is the Kronecker delta and then $P_m f(jh) = 2^{m/2} c_{m,j}$. Finally, if we take into account that $c_{m,j} = \int_{\mathbb{R}} f(x) \varphi_{m,j}(x) \, dx$. Thus,

$$P_m f(jh) = 2^{m/2} \cdot 2^{m/2} \int_{\mathbb{R}} f(x) \varphi(2^m x - j) \, dx = 2^m \int_{\mathbb{R}} f(x) \text{sinc}(2^m x - j) \, dx,$$

and this concludes the proof since $2^m = 1/h$. 

(A. Leitao & L. Ortiz-Gracia & E. Wagner)
The error committed in the approximation of $\hat{f}_{Z_{i-1}}(\xi)$ is bounded.

**Proposition**

Let $F_{Z_{i-1}}(\xi)$, $G_{Z_{i-1}}(\xi)$ and $E(\xi)$ be defined as follows,

$$F_{Z_{i-1}}(\xi) = 2^{\frac{m}{2}} \sum_{k=k_1}^{k_2} c_{m,k} \int_{\mathbb{R}} (e^x + 1)^{-i\xi} \text{sinc} (2^m x - k) \, dx,$$

$$G_{Z_{i-1}}(\xi) = 2^{-\frac{m}{2}} \sum_{k=k_1}^{k_2} c_{m,k} \left( e^{\frac{k}{2^m}} + 1 \right)^{-i\xi},$$

and the difference, $E(\xi) = F_{Z_{i-1}}(\xi) - G_{Z_{i-1}}(\xi)$.

Then, $|E(\xi)|$ is uniformly bounded by

$$|E(\xi)| \leq C(k_2 - k_1 + 1)e^{-\frac{\pi^2}{2}2^m}$$

where $C$ is a constant.
Proof.

We observe that,

\[ E(\xi) = 2^{-\frac{m}{2}} \sum_{k=k_1}^{k_2} c_{m,k} \left[ 2^m \int_{\mathbb{R}} (e^x + 1)^{-i\xi} \text{sinc} (2^m x - k) \, dx - (e^{\frac{k}{2m}} + 1)^{-i\xi} \right]. \]

Then, by Theorem 1 with \( d = \pi \),

\[ |E(\xi)| \leq 2^{-\frac{m}{2}} C \sum_{k=k_1}^{k_2} |c_{m,k}| e^{-\pi^2 2^m}, \]

for a certain constant \( C \). The proposition holds by taking into account that,

\[ |c_{m,k}| \leq \int_{\mathbb{R}} f(x) |\varphi_{m,k}(x)| \, dx \leq 2^{\frac{m}{2}}, \]

where the last inequality is satisfied since \( f \) is a density function and \( |\varphi_{m,k}(x)| \leq 2^{\frac{m}{2}} \).
Payoff coefficients

To complete the SWIFT pricing formula, compute the payoff coefficients, \( V_{m,k} \),

\[
V_{m,k} = \frac{2^{m/2}}{2^{J-1}} \sum_{j=1}^{2^{J-1}} \left[ \frac{S_0}{N+1} \left( I_{2,j}^j,k(\tilde{x}, b) + I_{0,j}^j,k(\tilde{x}, b) \right) - K I_{0,j}^j,k(\tilde{x}, b) \right],
\]

where \( \tilde{x} = \log \left( \frac{K(N+1)}{S_0} - 1 \right) \) and the functions \( I_{0,j}^j,k \) and \( I_{2,j}^j,k \) are defined by the following integrals

\[
I_{0,j}^j,k(x_1, x_2) := \int_{x_1}^{x_2} \cos \left( C_j (2^m y - k) \right) \, dy,
\]

\[
I_{2,j}^j,k(x_1, x_2) := \int_{x_1}^{x_2} e^y \cos \left( C_j (2^m y - k) \right) \, dy,
\]

with \( C_j = \frac{2j-1}{2^J} \pi \). These integrals are analytically available.
"Greek" coefficients

- We consider $\Delta$ and $\Gamma$. In the context of Asian options under Lévy processes, only the payoff coefficients, $V_{m,k}$, are affected by $S_0$. Thus, by differentiating $V_{m,k}$ with respect to $S_0$, we obtain

$$V_{(1)}_{m,k} = \frac{2^{m/2}}{2^{J-1}} \sum_{j=1}^{2^{J-1}} \left[ l_{2}^{i,k}(\tilde{x}, b) + l_{0}^{i,k}(\tilde{x}, b) \right] + \frac{S_0 \left( \frac{\partial l_{2}^{i,k}(\tilde{x}, b)}{\partial S_0} + \frac{\partial l_{0}^{i,k}(\tilde{x}, b)}{\partial S_0} \right)}{N+1} - K \frac{\partial l_{0}^{i,k}(\tilde{x}, b)}{\partial S_0}.$$

- Applying the chain rule, the partial derivatives of $l_{u}^{i,k}$, $u \in \{0, 2\}$,

$$\frac{\partial l_{u}^{i,k}(\tilde{x}, b)}{\partial S_0} = \frac{\partial l_{u}^{i,k}(\tilde{x}, b)}{\partial \tilde{x}} \frac{\partial \tilde{x}}{\partial S_0}, \quad \frac{\partial l_{u}^{i,k}(a, \tilde{x})}{\partial S_0} = \frac{\partial l_{u}^{i,k}(a, \tilde{x})}{\partial \tilde{x}} \frac{\partial \tilde{x}}{\partial S_0},$$

where

$$\frac{\partial \tilde{x}}{\partial S_0} = -\frac{K(N+1)}{S_0 K(N+1) - S_0^2},$$

and $\frac{\partial l_{u}^{i,k}(\tilde{x}, b)}{\partial \tilde{x}}$ and $\frac{\partial l_{u}^{i,k}(a, \tilde{x})}{\partial \tilde{x}}$ have analytic solution.

- Following the same procedure, a closed-form solution can be similarly derived for the second derivative of $V_{m,k}$. 

Á. Leitao & L. Ortiz-Gracia & E. Wagner
Asian SWIFT method
December 19, 2018 36 / 45
Numerical results

- We compare the SWIFT method against a state-of-the-art method, the well-known COS method, particularly the COS variant for arithmetic Asian option, called ASCOS method [5].

- To the best of our knowledge, the ASCOS method provides the best balance between accuracy and efficiency.

- Arithmetic Asian call option valuation with varying number of monitoring dates, $N = 12$ (monthly), $N = 50$ (weekly) and $N = 250$ (daily), and conceptually different underlying Lévy dynamics: Geometric Brownian motion (GBM) and Normal inverse Gaussian (NIG).

- We assess not only the accuracy in the solution but also the computational performance.

- All the experiments have been conducted in a computer system with the following characteristics: CPU Intel Core i7-4720HQ 2.6GHz and memory of 16GB RAM. The employed software package is Matlab R2017b.
### Table: GBM. The reference values are 11.9049157487 (N = 12), 11.9329382045 (N = 50) and 11.9405631571 (N = 250).
### Reference NIG

<table>
<thead>
<tr>
<th># Decimals</th>
<th>Method</th>
<th>( N = 12 )</th>
<th>( N = 50 )</th>
<th>( N = 250 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ASCOS SWIFT</td>
<td>( N_c = 64, n_q = 100 )</td>
<td>( N_c = 128, n_q = 200 )</td>
<td>( N_c = 128, n_q = 200 )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( m = 5 )</td>
<td>( m = 5 )</td>
<td>( m = 4 )</td>
</tr>
<tr>
<td>2</td>
<td>ASCOS SWIFT</td>
<td>( N_c = 128, n_q = 200 )</td>
<td>( N_c = 128, n_q = 200 )</td>
<td>( N_c = 192, n_q = 300 )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( m = 6 )</td>
<td>( m = 5 )</td>
<td>( m = 5 )</td>
</tr>
<tr>
<td>3</td>
<td>ASCOS SWIFT</td>
<td>( N_c = 128, n_q = 200 )</td>
<td>( N_c = 192, n_q = 300 )</td>
<td>( N_c = 192, n_q = 300 )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( m = 6 )</td>
<td>( m = 5 )</td>
<td>( m = 7 )</td>
</tr>
<tr>
<td>4</td>
<td>ASCOS SWIFT</td>
<td>( N_c = 256, n_q = 400 )</td>
<td>( N_c = 256, n_q = 400 )</td>
<td>( N_c = 512, n_q = 800 )</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( m = 7 )</td>
<td>( m = 8 )</td>
<td>( m = 9 )</td>
</tr>
</tbody>
</table>

**Table:** NIG. The reference values are 1.0135 \((N = 12)\), 1.0377 \((N = 50)\) and 1.0444 \((N = 250)\).
Results on GBM

<table>
<thead>
<tr>
<th></th>
<th>GBM</th>
<th>ASCOS</th>
<th>SWIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N = 12$</td>
<td>$N = 50$</td>
<td>$N = 250$</td>
</tr>
<tr>
<td>$N_c = 64, n_q = 100$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>$3.75 \times 10^{-4}$</td>
<td>$8.34 \times 10^{-4}$</td>
<td>$7.17 \times 10^{-3}$</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.03</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>$N_c = 128, n_q = 200$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>$8.37 \times 10^{-7}$</td>
<td>$7.43 \times 10^{-6}$</td>
<td>$3.82 \times 10^{-5}$</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.03</td>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>$N_c = 256, n_q = 400$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>=</td>
<td>$5.33 \times 10^{-7}$</td>
<td>$1.58 \times 10^{-7}$</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.16</td>
<td>0.12</td>
<td>0.11</td>
</tr>
<tr>
<td>$N_c = 512, n_q = 800$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>=</td>
<td>=</td>
<td>$3.04 \times 10^{-8}$</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>1.96</td>
<td>1.80</td>
<td>1.85</td>
</tr>
<tr>
<td>$N_c = 1024, n_q = 1600$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>SWIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$m = 4$</td>
</tr>
<tr>
<td>Error</td>
<td>$2.70 \times 10^{-4}$</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.01</td>
</tr>
<tr>
<td>$m = 5$</td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>$7.47 \times 10^{-9}$</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.01</td>
</tr>
<tr>
<td>$m = 6$</td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>=</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.02</td>
</tr>
<tr>
<td>$m = 7$</td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>=</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.08</td>
</tr>
<tr>
<td>$m = 8$</td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>=</td>
</tr>
<tr>
<td>Time (sec.)</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Table: SWIFT vs. ASCOS. Setting: GBM, $S_0 = 100$, $r = 0.0367$, $\sigma = 0.17801$, $T = 1$ and $K = 90$. The reference values are 11.9049157487 ($N = 12$), 11.9329382045 ($N = 50$) and 11.9405631571 ($N = 250$).
## Results on NIG

<table>
<thead>
<tr>
<th>NIG</th>
<th>$N = 12$</th>
<th>$N = 50$</th>
<th>$N = 250$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_c = 64, n_q = 100$</td>
<td>Abs error</td>
<td>$7.78 \times 10^{-3}$</td>
<td>$1.71 \times 10^{-1}$</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>$N_c = 128, n_q = 200$</td>
<td>Abs error</td>
<td>$2.60 \times 10^{-4}$</td>
<td>$5.89 \times 10^{-3}$</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>$N_c = 256, n_q = 400$</td>
<td>Abs error</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.19</td>
<td>0.17</td>
</tr>
<tr>
<td>$N_c = 512, n_q = 800$</td>
<td>Abs error</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>1.98</td>
<td>1.96</td>
</tr>
<tr>
<td>$N_c = 1024, n_q = 1600$</td>
<td>Abs error</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>14.38</td>
<td>14.22</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>ASCOS</th>
<th>SWIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m = 4$</td>
<td>Abs error</td>
<td>$9.72 \times 10^{-2}$</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.02</td>
</tr>
<tr>
<td>$m = 5$</td>
<td>Abs error</td>
<td>$5.69 \times 10^{-3}$</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.02</td>
</tr>
<tr>
<td>$m = 6$</td>
<td>Abs error</td>
<td>$2.13 \times 10^{-4}$</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.02</td>
</tr>
<tr>
<td>$m = 7$</td>
<td>Abs error</td>
<td>=</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.13</td>
</tr>
<tr>
<td>$m = 8$</td>
<td>Abs error</td>
<td>=</td>
</tr>
<tr>
<td></td>
<td>CPU time</td>
<td>0.39</td>
</tr>
</tbody>
</table>

**Table:** SWIFT vs. ASCOS. Setting: **NIG**, $S_0 = 100$, $r = 0.0367$, $\sigma = 0.0$, $\alpha = 6.1882$, $\beta = -3.8941$, $\delta = 0.1622$, $T = 1$ and $K = 110$. The reference values are $1.0135$ ($N = 12$), $1.0377$ ($N = 50$) and $1.0444$ ($N = 250$).
### Results Greeks

<table>
<thead>
<tr>
<th>Strike</th>
<th>Method</th>
<th>$K = 80%$</th>
<th>$K = 100%$</th>
<th>$K = 120%$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GBM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\Delta$</td>
<td>SWIFT</td>
<td>0.97573</td>
<td>0.57645</td>
<td>0.05984</td>
</tr>
<tr>
<td>Ref.</td>
<td></td>
<td>0.97519</td>
<td>0.57036</td>
<td>0.05979</td>
</tr>
<tr>
<td>$\Gamma$</td>
<td>SWIFT</td>
<td>0.00182</td>
<td>0.03788</td>
<td>0.01123</td>
</tr>
<tr>
<td>Ref.</td>
<td></td>
<td>0.00181</td>
<td>0.03777</td>
<td>0.01145</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NIG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\Delta$</td>
<td>SWIFT</td>
<td>0.95132</td>
<td>0.67561</td>
<td>0.03562</td>
</tr>
<tr>
<td>Ref.</td>
<td></td>
<td>0.95015</td>
<td>0.67220</td>
<td>0.03503</td>
</tr>
<tr>
<td>$\Gamma$</td>
<td>SWIFT</td>
<td>0.00268</td>
<td>0.03639</td>
<td>0.00716</td>
</tr>
<tr>
<td>Ref.</td>
<td></td>
<td>0.00272</td>
<td>0.03617</td>
<td>0.00733</td>
</tr>
</tbody>
</table>

**Table:** Option sensitivities, Greeks $\Delta$ and $\Gamma$. Strike $K$ as a % of $S_0$. Setting: $N = 12$, $m = 6$. 
Conclusions

- A new Fourier inversion-based technique has been proposed in the framework of discretely monitored Asian options under exponential Lévy processes.
- The application of SWIFT to the Asian pricing problem allows to overcome the main drawbacks attributed to this type of methods.
- Specially, SWIFT allows to avoid the numerical integration in the recovery of the characteristic function.
- SWIFT results in a highly accurate and fast technique, outperforming the competitors in most of the analysed situations.


Acknowledgements & Questions

Thanks to support from MDM-2014-0445

More: leitao@ub.edu and alvaroleitao.github.io

Thank you for your attention