, , , , 1–30

An SFP–FCC Method for Pricing and Hedging Early-exercise Options under Lévy Processes

Tat Lung (Ron) Chan*†

†School of Business, University of East London, Water Lane, Stratford, UK, E15 4LZ

(*Received 00 Month 20XX; in final form 00 Month 20XX*)

This paper extends the singular Fourier–Padé (SFP) method proposed by Chan (2018) for pricing/hedging early-exercise options–Bermudan, American and discrete-monitored barrier options–under a Lévy process. The current SFP method is incorporated with the Filon–Clenshaw–Curtis (FCC) rules invented by Domínguez *et al.* (2011), and we call the new method SFP–FCC. The main purpose of using the SFP–FCC method is to require a small number of terms to yield fast error convergence and to formulate option pricing and option Greek curves rather than individual prices/Greek values. We also numerically show that the SFP–FCC method can retain a global spectral convergence rate in option pricing and hedging when the risk-free probability density function is piecewise smooth. Moreover, the computational complexity of the method is $\mathcal{O}((L-1)(N+1)(\tilde{N}\log\tilde{N}))$ with N, a (small) number of complex Fourier series terms, $\tilde{N}$, a number of Chebyshev series terms and L, the number of early-exercise/monitoring dates. Finally, we compare the accuracy and computational time of our method with those of existing techniques in numerical experiments.
JEL classification: C6, C63

Keywords: Singular Fourier–Padé, Chebyshev Series, Filon–Clenshaw–Curtis rules, early-exercise options, discrete-monitored barrier options, Lévy process

JEL Classification: C6, C63

1. Introduction

In computational finance, various efficient methods for pricing and hedging early-exercise options under Lévy processes have been developed in financial markets. One of the most prominent approaches is to solve an option pricing formula–a partial integro-differential equation (PIDE)–by applying the finite different (FD) method (e.g. Hirsa and Madan 2004, Almendral and Oosterlee 2007, Almendral 2005, Hirsa 2012, Wang *et al.* 2007), the finite element (FE) method (e.g. Matache *et al.* 2005, Rambeerich *et al.* 2009), and the radial basis (RB) method (e.g. Chan and Hubbert 2014, Brummelhuis and Chan 2014, Chan 2016). In addition to PDIE-based approaches, a Fourier-transformation-based method is widely introduced, the most prominent examples of which can be found in Lord *et al.* (2008), Feng and Linetsky (2008), Fang and Oosterlee (2009b), Chan (2019). Simulation-based approaches also play a pivotal part in pricing early-exercise options. The least-squares Monte-Carlo (LSM) approach of Longstaff and Schwartz (2001) is the most popular method among other simulation-based ones, and Glasserman (2003) provides a comprehensive overview of different Monte-Carlo methods in option pricing and risk management. Finally, an analytical approximation approach (e.g. Guo and Li 2016) and a deep/machine-learning approach (e.g. Fu and

*Corresponding author. Email: t.l.chan@uel.ac.uk

Hirsa 2019, Hirsa *et al.* 2019) have recently drawn a substantial amount of researchers' attention due to their rapid computational speed and higher accuracy. In this paper, we focus mainly on improving a Fourier-transformation-based method and propose the singular Fourier-Padé (SFP) method (cf. Driscoll and Fornberg 2001, 2011), which has been previously introduced as a method to price and hedge European options by Chan (2018), to price Bermudan, American and discrete-monitored barrier options.

A Bermudan option can be exercised on predetermined dates before maturity. The option holder receives the exercise payoff when he/she exercises the option on specific dates at the option's maturity. Between two consecutive exercise dates, the valuation process can be regarded as similar to a European option, which can be priced and hedged using the risk-neutral valuation formula (cf. Chan 2018, Chan and Hale 2019).

If we consider $\log S_t := x_t$ driven by a Lévy process and a Bermudan option with strike K and maturity T that can be exercised only on a given number of exercise dates $t = t_0 < t_1 \leq t_2 \leq \ldots t_l \leq t_{l+1} \leq \ldots \leq t_L = T.$, then we can write the risk-neutral Bermudan pricing formula for such an option as

$$V(x_{t_l}, K, t_l) = \begin{cases} U(e^{x_{t_l}}, K, t_l) & l = L,\ t_L = T \\ \max\left(C(x_{t_l}, K, t_l), U(e^{x_{t_l}}, K, t_l)\right) & l = 1, 2, 3, \ldots, L-1\ , \\ C(x_{t_l}, K, t_l) & l = 0 \end{cases} \tag{1}$$

where $U(e^{x_{t_l}}, K, t_l)$ is the payoff function at $t_l.$, i.e., if the payoff function is a call, then $U(e^{x_{t_l}}, K, t_l)$ is transformed into $\max\left(e^{x_{t_l}} - K, 0\right)$. In (1), $C(x_{t_l}, K, t_l)$ at each t_j can be described as a risk-neutral valuation formula:

$$\begin{aligned} C(x_{t_j}, K, t_j) &= e^{-r(t_{j+1}-t_j)}\mathbb{E}\left(V(x_{t_{j+1}}, K, t_{j+1})|x_{t_j}\right) \\ &= e^{-r(t_{j+1}-t_j)}\int_{-\infty}^{+\infty} V(e^{x+\chi-\log K}, t_{j+1})f(\chi)\mathrm{d}\chi, \quad \chi \in X_{t_{j+1}} - X_{t_j}. \end{aligned} \tag{2}$$

Here, $X_{t_{j+1}} - X_{t_j}$ is the Lévy process, r is the risk-neutral interest rate, and $f(\chi)$ is the risk-neutral probability density function (PDF). As (2) is an expectation and integral, a sustainable number of numerical methods are developed for its calculation. Popular methods include, for example, the FFT–QUAD method, a combination of the Fast Fourier Transform (FFT) method and numerical quadrature, suggested by O'Sullivan (2005); the CONV method, an FFT method proposed by Lord *et al.* (2008); a mixture of the FFT method and Gauss transform (e.g. Broadie and Yamamoto 2003) or Hilbert transform (e.g. Feng and Linetsky 2008, Zeng and Kwok 2014); the COS method, a Fourier-cosine series approach suggested by Fang and Oosterlee (2009b); and the SWIFT method, a wavelet series approach (Maree 2015, Maree *et al.* 2017). The advantage of using the FFTs, COS and SWIFT methods for option pricing is that they can achieve a global spectral (exponential) convergence rate and require fewer summation terms as long as the governing PDF is sufficiently smooth. However, when the difference Δt between t_j and t_{j+1} approaches zero in (2), $f(\chi)$ tends to become highly peaked and piecewise continuous (non-smooth)[1] in any Lévy process. Accordingly, $f(\chi)$ contains a highly peaked point–a jump (a singularity) where the first derivative of $f(\chi)$ is discontinuous. Using any type of Fourier series to represent a piecewise continuous function, e.g., a piecewise continuous PDF, with a jump is notoriously fraught and causes the Gibbs phenomenon (cf. Driscoll and Fornberg 2001, 2011). As a result, the Fourier representation of a piecewise continuous function has a pointwise convergence elsewhere at the rate $\mathcal{O}(N^{-1})$ because of a jump acting as a pole[2] in a complex plane. The impact of the Gibbs phenomenon is therefore detrimen-

[1] A function is called piecewise continuous on an interval if the function is made up of a finite number of ν times with differentiable continuous pieces.

[2] A pole occurs where the function f is not well-defined or tends to infinity in a complex plane.

tal because it can lead to inaccurate pricing and hedging and a lack of spectral convergence when the approximate option prices are generated via FFT or Fourier series methods at or around the jumps.

Accordingly, we propose the singular Fourier-Padé (SFP) method (Chan 2018) to circumvent the mentioned problem to allow fewer summation terms and maintain spectral convergence when $f(\chi)$ is piecewise continuous. Why do we choose the SFP method? We exhibit the following characteristics when we use the method to price and hedge European-type options:

(i) a global spectral convergence rate for piecewise continuous PDFs;
(ii) fast error convergence with fewer partial summation terms required;
(iii) accurate pricing of any European-type option with features of deep in/out of the money and very long/short maturities;
(iv) consistent accuracy for approximating large or small option prices throughout.

To obtain the same advantages of using the SFP method, we extend the current method with the help of the Filon–Clenshaw–Curtis (FCC) rules, invented by Domínguez *et al.* (2011), to price Bermuda options and American and discrete-monitored barrier options. We call the new method SFP–FCC. The main advantage of the SFP–FCC method compared with the SFP method alone is that it not only requires fewer summation terms to yield spectral convergence with a (piecewise) continuous PDF but also provides option pricing and an optional Greek formula rather than individual prices/Greek values.

The remainder of this paper is structured as follows. Section 1 provides an introduction. Section 2 describes the SFP method. Section 3 introduces the financial stochastic models that we examine in this paper. Section 4 revises and improves the formulation of the SFP option pricing formulae for the European options proposed in Chan (2018). In Section 5, we propose the SFP–FCC algorithms/formulae to price Bermudan, American (cf. Section 5.1) and discrete-monitored barrier options (cf. Section 5.3) and to find an early-exercise point by using root-finding techniques (cf. Section 5.2). Section 6 describes the derivation of the option Greek formulae and the choice of truncated integration intervals. Section 7 discusses, analyses and compares the numerical results of the SFP–FCC method with the results of other numerical methods. We conclude and discuss possible future developments in Section 8. Finally, the Appendix A shows the algorithm used for computation of the SFP coefficients, and Appendix B discusses the method of locating jumps in PDFs. Appendix C describes the FCC rules, and Appendix D shows the table of cumulants.

2. Singular Fourier–Padé interpretation and correction of the Gibbs phenomenon

If we consider a function f with a formal power series representation $\sum_{k=0}^{\infty} b_k x^k$, and a rational function defined by $R_{N,M} = P_N/Q_M$, where P_N and Q_M are the polynomials of

$$P_N(x) = \sum_{n=0}^{N} p_n x^n \text{ and } Q_M(x) = \sum_{m=0}^{M} q_m x^m, \tag{3}$$

respectively, then we say that $R_{N,M} = P_N/Q_M$ is the *(linear) Padé approximant* of order (N, M) of the formal series that satisfies the condition

$$\left(\sum_{n=0}^{N} p_n x^n\right) - \left(\sum_{m=0}^{M} q_m x^m\right)\left(\sum_{k=0}^{M+N} b_k x^k\right) = \mathcal{O}(x^{N+M+1}). \tag{4}$$

Here, f is approximated by $\sum_{k=0}^{M+N} b_k x^k$,. To obtain the approximant $R(N,M)$, we simply calculate the coefficients of polynomials P_N and Q_M by solving the following system of linear equations:

$$\sum_{j=0}^{M} b_{N-j+k} q_k = 0, \quad k = 1, \ldots, M. \tag{5}$$

$$\sum_{j=0}^{k} b_{k-j} q_j = p_k, \quad k = 1, \ldots, N. \tag{6}$$

For this system to be well determined, we usually employ a normalisation by setting, for example, $q_0 = 1$.

If we now consider any piecewise analytic real function f in a finite interval $[a,b]$ with a set of jump $\{\zeta_s\}_{s=1}^{\mathcal{S}} \in [a,b]$, singularities that cause poles in a complex plane after the first derivative of f being discontinuous, the complex Fourier series (CFS) representation of the function is defined as

$$f(x) = \mathfrak{Re}\left[\sum_{k=-\infty}^{\infty} b_k e^{i\frac{2\pi}{b-a}kx}\right], \text{ with } b_k = \frac{1}{b-a}\int_a^b f(x) e^{-i\frac{2\pi}{b-a}kx} dx. \tag{7}$$

Here, $\mathfrak{Re}$ represents the real part of the function. As we focus on approximating a real function, we can further obtain

$$f(x) = \mathfrak{Re}\left[2\sum_{k=1}^{\infty} b_k e^{i\frac{2\pi}{b-a}kx} + b_0\right]. \tag{8}$$

Based on this representation, we denote z as $\exp\left(i\frac{2\pi}{b-a}x\right)$, and then, we approximate f with a truncated power series of f_1 such that

$$f(x) \approx f_1(z) = \mathfrak{Re}\left[2\sum_{k=1}^{N+M} b_k z^k + b_0\right]. \tag{9}$$

The transformation $z = \exp\left(i\frac{2\pi}{b-a}x\right)$ also suggests that the jumps in f translate into $\varepsilon_1, \ldots, \varepsilon_s, \ldots \varepsilon_{\mathcal{S}} = \exp\left(i\frac{2\pi}{b-a}\zeta_{\mathcal{S}}\right)$. Based on (4), the Fourier-Padé approximation of f_1 comprises the polynomials

$$P_N(z) = Q_M(z) f_1(z) + \mathcal{O}(z^{N+M+1}), \quad z \to 0. \tag{10}$$

However, Driscoll and Fornberg (2001, 2011) note that this approximant (10) does not reproduce very well at/around the jump of the function, which makes the approximation inaccurate. Therefore, they suggest that every jump ε_s can be attributed to a logarithm of the form

$$\log\left(1 - \frac{z}{\varepsilon_s}\right) \tag{11}$$

This logarithmic jump (logarithmic pole) in f_1, which is difficult for the Padé approximant to simulate, can be exploited to enhance the approximation process. This is the rationale behind the SFP method introduced in Driscoll and Fornberg (2001, 2011). We modify the Fourier-Padé

approximant (10) to obtain the following condition:

$$P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z) \log\left(1 - z/\varepsilon_s\right) = f_1(z) Q_M(z) + \mathcal{O}(z^{U+1}), \tag{12}$$

where

$$\begin{array}{ll} P_N(z) = \sum_{n=0}^{N} p_n z^n, & Q_M(z) = \sum_{m=0}^{M} q_m z^m \neq 0, \\ L_{N_s}(z) = \sum_{n_s=0}^{N_s} l_{n_s} z^{n_s}, & s = 1, \ldots, \mathcal{S}, \\ U = N + M + \mathcal{S} + \sum_{s=1}^{\mathcal{S}} N_s, & \varepsilon_s = \exp\left(i \frac{2\pi}{b-a} \zeta_s\right), \\ \zeta_s \text{ is a jump at } s. & \end{array} \tag{13}$$

3. Financial modelling with Lévy processes

We briefly review option pricing theory in Lévy models partly to establish notations. Standard references for this material are Applebaum (2004), Cont and Tankov (2004), and Sato (1999). Throughout this section, we consider that markets are frictionless and have no arbitrage, and we assume that an equivalent martingale measure (EMM) $\mathbb{Q}$ is chosen by the market. Moreover, there is a complete filtered probability space $(\Omega, \mathcal{F}, \{\mathcal{F}\}_{t\geq 0}, \mathbb{Q})$ on which all processes are assumed to live.

We first introduce a stock price process $S = (S_t)_{t\leq 0}$ and assume that it follows an exponential Lévy process:

$$S_t = S_0 e^{L_t}, \quad t \geq 0, \tag{14}$$

where $S_0 \in \mathbb{R}^+ = (0, \infty)$ is the initial stock price taken as a random variable (rv) independent of $(L_t)_{t\leq 0}$. We limit ourselves to derivatives written on a single risky asset with a log return assumed to be modelled by a one-dimensional Lévy process. As usual, we also assume the existence of a risk-free bond earning interest at a constant rate of r and a continuous compounding stock dividend q for all maturities $T > 0$. For a general Lévy process, the market that consists of the risky asset plus the risk-free bond will be an incomplete market[1].

The Lévy-process $(L_t)_{t\geq 0}$ is fully determined by its characteristic function that, according to the Lévy–Khinchine theorem, is of the form $\varphi(u) := \mathbb{E}(e^{iuL_t}) = e^{t\phi(u)}$, with the characteristic exponent $\phi(z)$ given by

$$\phi(u) = i\gamma u - \frac{1}{2}\sigma^2 u^2 + \int_{\mathbb{R}} \left(e^{ixu} - 1 - i\chi u \mathbf{1}_{\{|\chi|\leq 1\}}\right) \nu(\mathrm{d}\chi). \tag{15}$$

Here, γ and σ are real constants with $\sigma \geq 0$, and ν is a positive measure of $\mathbb{R}$, which is called the Lévy measure that satisfies the Lévy-condition $\int_{\mathbb{R}} \min(\chi^2, 1)\, \nu(\mathrm{d}\chi) < \infty$. The probabilistic interpretation of ν is that $\nu(\mathrm{d}\chi)$ gives the expected number of jumps with a size between χ and $\chi+\mathrm{d}\chi$, which the process makes between time 0 and 1. The triplet (γ, σ, ν) is called the characteristic triplet or the Lévy-Khintchine triplet of $(L_t)_{t\geq 0}$.

We also assume that $\mathbb{E}[S_0] \leq 0$ and a recall of (14). Then, we can write

$$\mathbb{E}[S_t] = \mathbb{E}[S_0]\mathbb{E}[e^{L_t}] = \mathbb{E}[S_0] e^{t\phi(1)}, \tag{16}$$

[1]Markets are complete when the Lévy process is a Brownian motion - the classical Black and Scholes model - or if it is a Poisson process

where $\phi(1)$ is assumed to be finite. For any EMM, $\mathbb{Q}$ is a risk-neutral (no-arbitrage) pricing, and the discounted stock price process, $(e^{-(r-q)t}S_t)_{t\geq 0}$, in an equilibrium, with either a complete or an incomplete market, must constitute a martingale. In addition, under the EMM $\mathbb{Q}$ measure, the growth rate $\phi(1)$ of the stock price equals the risk-free rate $r > 0$ and $q > 0$.

4. Pricing formulae for European type options

In this section, we derive an SFP European option pricing formula. The technique demonstrated is slightly different from the approach in Chan (2018), as we provide an option pricing curve rather than an individual value.

A European option can be exercised at maturity T of the option. By providing the current log price $x := \log S$, the strike price of K and the probability density function (PDF) f of a stochastic process, we can express the option price $V(x, K, t)$ starting at time t with its contingent claim that pays out $U(S_T)$ is as follows:

$$\begin{aligned} V(x,K,t) &= e^{-r(T-t)}\mathbb{E}(U(S_T,K,T)|S_t = e^x) \\ &= e^{-r(T-t)}\mathbb{E}(U(S_t e^{X_T - X_t},K,T))) \\ &= e^{-r(T-t)}\int_{-\infty}^{+\infty} G(e^{x+\chi-\log K})f(\chi)d\chi, \quad \chi \in X_T - X_t, \end{aligned} \tag{17}$$

where $U(S_t e^{X_T - X_t}, K, T) = G(e^{x+\chi-\log K})$. By replacing $x + \chi - \log K$ with y, we have

$$V(x,K,t) = e^{-r(T-t)}\int_{-\infty}^{+\infty} G(e^y) f\left(y - x + \log K\right) \mathrm{d}y \tag{18}$$

$$= e^{-r(T-t)}\int_{-\infty}^{+\infty} G(e^y) f^R\left(\tilde{x} - y\right) \mathrm{d}y, \tag{19}$$

where $\tilde{x} = x - \log K$, $G(e^y)$ is the pay-off in the log-price coordinates and $f^R(\tilde{x}) := f(-\tilde{x})$ is the reflected function. The expression of (18) is indeed a cross-correlation integral; however, since we introduce the idea of the reflected function $f^R(\tilde{x}) := f(-\tilde{x})$, we can instead turn (18) into a convolution integral (19).

If we consider approximating $V(x, K, t)$ in a finite interval $[c, d]$ rather than in $[-\infty, \infty]$, such that the choice of $[c, d]$ satisfies the condition

$$\int_c^d f(\chi)e^{iu\chi}\mathrm{d}\chi \approx \int_{-\infty}^{+\infty} \mathrm{f}(\chi)\mathrm{e}^{\mathrm{iu}\chi}\mathrm{d}\chi = \mathbb{E}[\mathrm{e}^{\mathrm{iu}(\mathrm{X_T}-\mathrm{X_t})}] := \varphi(\mathrm{u}), \tag{20}$$

where $\varphi(u)$ is a characteristic function of $X_T - X_t$., then (19) becomes

$$V(x,K,t) \approx e^{-r(T-t)}\int_c^d G(e^y) f^R(\tilde{x} - y)\mathrm{d}y. \tag{21}$$

By using the Fourier transform shift theorem and the CFS expansion shown in (8), we express $f^R\left(\tilde{x} - y\right)$ as

$$\mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} b_k e^{-i\frac{2\pi}{b-a}ky}\right], \tag{22}$$

where

$$b_k = \frac{1}{d-c}\int_c^d f(y)e^{-i\frac{2\pi}{d-c}ky}\mathrm{d}y\left(e^{i\frac{2\pi}{d-c}k\tilde{x}}\right) \quad \text{and} \quad b_0 = \frac{1}{d-c}\int_c^d f(y)\mathrm{d}y. \tag{23}$$

Through substitution, we obtain

$$V(x,K,t) = e^{-r(T-t)}\mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} b_k g_k e^{i\frac{2\pi}{d-c}k\tilde{x}}\right], \tag{24}$$

where,

$$b_k = \frac{1}{d-c}\int_c^d f^R(y)e^{-i\frac{2\pi}{d-c}ky}\mathrm{d}y \quad \text{and} \quad b_0 = \frac{1}{d-c}\int_c^d f^R(y)\mathrm{d}y. \tag{25}$$

$$g_k = \int_c^d G(e^y)e^{-i\frac{2\pi}{d-c}ky}\mathrm{d}y \quad \text{and} \quad g_0 = \int_c^d G(e^y)\mathrm{d}y. \tag{26}$$

Because of condition (20), we can approximate b_k and b_0 as

$$\widehat{B}_k := \frac{1}{d-c}\varphi\left(\frac{2\pi}{d-c}k\right) \quad \text{and} \quad \frac{1}{d-c}\widehat{B}_0 := \varphi(0) = 1, \tag{27}$$

respectively. Furthermore, since we only consider a vanilla call/put in this paper, their payoffs are formulated as

$$U(S_t,K,T) = \begin{cases} \max\left(e^{x+\chi}-K,0\right) = K\max\left(e^{x+\chi-\log K}-1,0\right): & \text{(call)} \\ \max\left(K-e^{x+\chi},0\right) = K\max\left(1-e^{x+\chi-\log K},0\right): & \text{(put)} \end{cases}. \tag{28}$$

By considering $y := x+\chi-\log K$ and applying basis calculus, we obtain

$$\begin{aligned}\widehat{G}_k &= \int_c^d \max\left(e^y-1,0\right)e^{-i\frac{2\pi}{d-c}ky}\mathrm{d}y \\ &= \left(\frac{d-c}{d-c-i2\pi k}\left(e^{(1-i\frac{2\pi}{d-c}k)d}-1\right)+\frac{d-c}{i2\pi k}\left(e^{-i\frac{2\pi}{d-c}kd}-1\right)\right)\end{aligned} \tag{29}$$

for a call, and similarly, we obtain

$$\begin{aligned}\widehat{G}_k &= \int_c^d \max\left(1-e^y,0\right)e^{-i\frac{2\pi}{d-c}ky}\mathrm{d}y \\ &= \left(\frac{d-c}{d-c-i2\pi k}\left(e^{(1-i\frac{2\pi}{d-c}k)c}-1\right)+\frac{d-c}{i2\pi k}\left(e^{-i\frac{2\pi}{d-c}kc}-1\right)\right)\end{aligned} \tag{30}$$

for a put. Accordingly, we replace b_k with KG_k, and the new CFS representation of (24) becomes

$$V(x,K,t) := e^{-r(T-t)}K\mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty}\widehat{B}_k\widehat{G}_k e^{i\frac{2\pi}{d-c}k\tilde{x}}\right]. \tag{31}$$

To express our final pricing formula with the SFP representation, as we know the pricing formula

is a real function, we can transform (31) into

$$V(x,K,t) := e^{-r(T-t)}K\mathfrak{Re}\left[2\sum_{k=1}^{\infty}\widehat{B}_k\widehat{G}_k e^{i\frac{2\pi}{d-c}k\tilde{x}} + \widehat{B}_0\widehat{G}_0\right]. \tag{32}$$

We set $\exp\left(i\frac{2\pi}{d-c}\tilde{x}\right)$ equal to z. The transformation $z = \exp\left(i\frac{2\pi}{d-c}\tilde{x}\right)$ maps the interval $[c,d]$ onto the unit circle in z. This change also transforms the jumps ζ along f into z with the form of $\varepsilon = \exp\left(i\frac{2\pi}{d-c}\zeta\right)$. Finally, by expressing (32) with a new variable of z, we have

$$2\sum_{k=1}^{\infty}\widehat{B}_k\widehat{G}_k z^k + \widehat{B}_0\widehat{G}_0. \tag{33}$$

By substituting the equation above with $f_1(z)$ in (12), we obtain the approximant given by

$$P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z)\log\left(1 - z/\varepsilon_s\right) = \left(2\sum_{k=1}^{U}\widehat{B}_k\widehat{G}_k z^k + \widehat{B}_0\widehat{G}_0\right)Q_M(z) + \mathcal{O}(z^{U+1}) \tag{34}$$

$$\begin{array}{rr} P_N(z) = \sum_{n=0}^{N} p_n z^n, & Q_M(z) = \sum_{m=0}^{M} q_m z^m \neq 0, \\ L_{N_s}(z) = \sum_{n_s=0}^{N_s} l_{n_s} z^{n_s}, & s = 1,\ldots,\mathcal{S}, \\ \varepsilon_s = e^{i\frac{2\pi}{d-c}\zeta_s}, & U = N + M + \sum_{s=1}^{\mathcal{S}} N_s. \end{array} \tag{35}$$

Once we can determine the unknown coefficients of $\{p_n\}_{n=0}^{N}$, $\{q_m\}_{m=0}^{M}$ and $\{l_{n_s}\}_{n_s=0}^{N_s}$ in (34) via the algorithm shown in Appendix A and replace

$$2\sum_{k=1}^{\infty}\widehat{B}_k\widehat{G}_k e^{i\frac{2\pi}{d-c}k\tilde{x}} + \widehat{B}_0\widehat{G}_0$$

with

$$\frac{P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z)\log\left(1 - z/\varepsilon_s\right)}{Q_M(z)}, \quad z = \exp\left(i\frac{2\pi}{d-c}\tilde{x}\right), \quad \tilde{x} = x - \log K$$

in (24), we reach our first SFP representation of a European vanilla option such that

$$V(x,K,t) := e^{-r(T-t)}K\mathfrak{Re}\left(\frac{P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z)\log\left(1 - z/\varepsilon_s\right)}{Q_M(z)}\right). \tag{36}$$

The above pricing formula can only be applied to compute the option prices with a value of K and a range of S_t. However, in the financial markets, option price quotes always appear with a value of S_t and a range of K. To fit in this financial phenomenon, we modify (36) by using $K = Se^{-\tilde{x}} = e^{x-\tilde{x}}$ so that we obtain the new pricing formula of

$$V(x,K,t) := e^{-r(T-t)+x-\tilde{x}}\mathfrak{Re}\left(\frac{P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z)\log\left(1 - z/\varepsilon_s\right)}{Q_M(z)}\right). \tag{37}$$

5. Pricing early-exercise options with the SFP–FCC method

In this section, we derive option pricing/hedging formulas for early-exercise options by using the SFP–FCC method. We formulate a Bermudan option pricing curve as the first illustration. Then, in the same fashion, we derive the SFP–FCC pricing formulas for the American and discrete-monitored barrier options and their hedging formulas.

The general idea of the SFP–FCC method is first to discretise the lifespan of the options in an equal time step. Then, starting backwards from maturity to the initial time of the option, we present the option pricing/hedging curve that applies the CFS method at each time step. The accuracy of the CFS method can only be guaranteed by implementing the FCC rules. Finally, once we reach the initial time of the option, the pricing/hedging formula of the option can be constructed by applying the SFP method.

5.1. *Pricing formulae for Bermudan and American options*

We consider $\log S_t := x_t$ driven by a Lévy process and a Bermudan option with strike K and maturity T that can be exercised only on a given number of exercise dates $t = t_0 < t_1 \leq t_2 \leq \ldots t_l \leq t_{l+1} \leq \ldots \leq t_L = T$. By assuming that the difference between t_l and its successive t_{l+1} are the same, we can write the Bermudan pricing formula for such an option as

$$V(x_{t_l}, K, t_l) = \begin{cases} U(e^{x_{t_l}}, K, t_l) & l = L,\, t_L = T \\ \max\left(C(x_{t_l}, K, t_l), U(e^{x_{t_l}}, K, t_l)\right) & l = 1, 2, 3, \ldots, L-1\,, \\ C(x_{t_l}, K, t_l) & l = 0 \end{cases} \tag{38}$$

where $U(e^{x_{t_l}}, K, t_l)$ is the payoff function at t_l. For example, if the payoff function is a call, then $U(e^{x_{t_l}}, K, t_l)$ is transformed into $\max\left(e^{x_{t_l}} - K, 0\right)$. In (38), $C(x_{t_l}, K, t_l)$ at each t_l can be defined as

$$C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)}\mathbb{E}\left(V(x_{t_{l+1}}, K, t_{l+1})|x_{t_l}\right). \tag{39}$$

$$= e^{-r(t_{l+1}-t_l)} \int_{-\infty}^{+\infty} V(x_{t_l} + \chi - \log K, t_{l+1}) f(\chi)\mathrm{d}\chi, \quad \chi \in X_{t_{l+1}} - X_{t_l}. \tag{40}$$

Following the algorithm of pricing European options in Section 4, we set $\tilde{x}_{t_l} = x_{t_l} - \log K$, replace $\tilde{x}_{t_l} + \chi$ with y_{t_l} and choose $[c, d]$ to satisfy (20). We can transform the equation above as a convolution integral, i.e.,

$$C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)} \int_c^d V(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\mathrm{d}y_{t_l}. \tag{41}$$

Due to the early-exercise feature of the option, $V(y_{t_l}, t_{l+1})$ is equal to $\max\left(C(y_{t_l}, t_{l+1}), U(e^{y_{t_l}}, t_{l+1})\right)$. Then, the integral of $C(x_{t_l}, K, t_l)$ in (41) can be split into two parts when we know the *early-exercise point*, $x^*_{t_l}$ at t_l. By supposing that we know $x^*_{t_l}$ (we discuss the techniques of finding $x^*_{t_l}$ in Section 5.2), we can split the integral that defines $C(x_{t_l}, K, t_l)$, into two parts: one on the interval $[c, x^*_{t_l}]$, and the second on $[x^*_{t_l}, d]$, i.e.,

$$C(x_{t_l}, K, t_l) = \begin{cases} \int_c^{x^*_{t_l}} C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\mathrm{d}y_{t_l} + \int_{x^*_{t_l}}^d U(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\mathrm{d}y_{t_l} : & \text{(call)} \\ \int_c^{x^*_{t_l}} U(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\mathrm{d}y_{t_l} + \int_{x^*_{t_l}}^d C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\mathrm{d}y_{t_l} : & \text{(put)} \end{cases}. \tag{42}$$

In (42), the integral of

$$\int U(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\, \mathrm{d}y_{t_l}$$

is clearly the CFS presentation of a European vanilla call or put on $[x^*_{t_l}, d]$ or $[c, x^*_{t_l}]$, respectively, because $U(y_{t_l}, t_{l+1})$ is a payoff, and the CFS representation of $f^R(\tilde{x}_{t_l} - y_{t_l})$, which is equivalent to (22), is defined as

$$f^R(\tilde{x}_{t_l} - y_{t_l}) = \mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k e^{i\frac{2\pi}{d-c}k(-y_{t_l}+\tilde{x}_{t_l})}\right], \tag{43}$$

where $\widehat{B}_k$ is the same as (27). Accordingly, by using the idea of deriving the CFS European option pricing formula in Section 4 and the result of (29) and (30), we can show that

$$\begin{aligned}
&\int_{x^*_{t_l}}^{d} U(y_{t_l}, t_{l+1}) \mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k e^{i\frac{2\pi}{d-c}k(-y_{t_l}+\tilde{x}_{t_l})}\right] \\
&\mathrm{d}y_{t_l} = K\mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{G}_k[x^*_{t_l}, d] e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_l}}\right] : \text{ (call)},
\end{aligned} \tag{44}$$

$$\begin{aligned}
&\int_{c}^{x^*_{t_l}} U(y_{t_l}, t_{l+1}) \mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k e^{i\frac{2\pi}{d-c}k(-y_{t_l}+\tilde{x}_{t_l})}\right] \\
&\mathrm{d}y_{t_l} = K\mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{G}_k[c, x^*_{t_l}] e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_l}}\right] : \text{ (put)},
\end{aligned} \tag{45}$$

where, $\widehat{G}_k[x^*_{t_l}, d]$ and $\widehat{G}_k[c, x^*_{t_l}]$ are the closed-form Fourier integrals on $[x^*_{t_l}, d]$ and $[c, x^*_{t_l}]$, respectively.

When we compute

$$\int C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l}, \tag{46}$$

it is not a straightforward case, as $C(y_{t_l}, t_{l+1})$ does not have a closed-form expression at t_{l+1}. To solve the integral and also yield a higher accuracy of the SFP-FCC method, we first approximate $C(y_{t_l}, t_{l+1})$ with a Chebyshev series since it has a CFS representation in the previous time step. Therefore,

$$C(y_{t_l}, t_{l+1}) = C_{cheb}(y_{t_l}, t_{l+1}) := \begin{cases} K \sum_{n=1}^{\infty} \alpha_n T_n \circ \psi_{[c, x^*_{t_l}]}(y_{t_l}) : & \text{(call)} \\ K \sum_{n=1}^{\infty} \alpha_n T_n \circ \psi_{[x^*_{t_l}, d]}(y_{t_l}) : & \text{(put)} \end{cases}. \tag{47}$$

Here, α_n is the n^{th} coefficient, and we also define the composition of $T_k \circ \psi_{[y_k, y_{k+1}]}$, where $\psi_{[y_k, y_{k+1}]}(y_{t_l}) = (2y_{t_l} - (y_{k+1} + y_k))/(y_{k+1} - y_k)$ is the linear mapping from $[y_k, y_{k+1}]$ to $[-1, 1]$. By

substituting (47) into (46) and expanding the integral (46), we have

$$\begin{aligned}
&\int_c^{x_{t_l}^*} C_{cheb}(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l} \\
&= K \sum_{k=-\infty}^{+\infty} \sum_{n=1}^{\infty} \widehat{B}_k \alpha_n \left(\int_c^{x_{t_l}^*} T_n \circ \psi_{[c, x_{t_l}^*]}(y_{t_l}) e^{-i\frac{2\pi}{d-c} k y_{t_l}} \mathrm{d}y_{t_l} \right) e^{i\frac{2\pi}{d-c} k \tilde{x}_{t_l}} : \quad \text{(call)}, \qquad (48) \\
&\int_{x_{t_l}^*}^d C_{cheb}(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l} \\
&= K \sum_{k=-\infty}^{+\infty} \sum_{n=1}^{\infty} \widehat{B}_k \alpha_n \left(\int_{x_{t_l}^*}^d T_n \circ \psi_{[x_{t_l}^*, d]}(y_{t_l}) e^{-i\frac{2\pi}{d-c} k y_{t_l}} \mathrm{d}y_{t_l} \right) e^{i\frac{2\pi}{d-c} k \tilde{x}_{t_l}} : \quad \text{(put)}. \qquad (49)
\end{aligned}$$

In the equations above, both integrals of

$$\int_c^{x_{t_l}^*} T_n \circ \psi_{[c, x_{t_l}^*]}(y_{t_l}) e^{-i\frac{2\pi}{d-c} k y_{t_l}} \mathrm{d}y_{t_l}, \text{ and } \int_{x_{t_l}^*}^d T_n \circ \psi_{[x_{t_l}^*, d]}(y_{t_l}) e^{-i\frac{2\pi}{d-c} k y_{t_l}} \mathrm{d}y_{t_l} \qquad (50)$$

can be simplified to

$$\widehat{T}_{n,k}[c, x_{t_l}^*] := \frac{x_{t_l}^* - c}{2} e^{-i\frac{d-c}{x_{t_l}^* - c} k\pi} \int_{-1}^{+1} T_n(s) \exp\left(i \left(-\frac{k(x_{t_l}^* - c)\pi}{d-c} \right) s \right) \mathrm{d}s : \quad \text{(call)} \qquad (51)$$

and

$$\widehat{T}_{n,k}[x_{t_l}^*, d] := \frac{d - x_{t_l}^*}{2} e^{-i\frac{d-c}{d - x_{t_l}^*} k\pi} \int_{-1}^{+1} T_n(s) \exp\left(i \left(-\frac{k(d - x_{t_l}^*)\pi}{d-c} \right) s \right) \mathrm{d}s : \quad \text{(put)}, \qquad (52)$$

respectively. We denote $\tilde{k}$ to be equal to either $-\frac{k(x_{t_l}^* - c)\pi}{d-c}$ or $-\frac{k(d - x_{t_l}^*)\pi}{d-c}$ to simplify the mathematical notation in the equations above. Therefore, we have

$$\int_{-1}^{+1} T_n(s) \exp(i\tilde{k}s) \mathrm{d}s, \quad n \geq 0. \qquad (53)$$

This integral is not easy to solve numerically because it is highly oscillatory (e.g., Domínguez *et al.* 2011). To yield higher accuracy, we apply the FCC rules stated in Appendix C to compute the integral. By using the final numerical result of (53), we can further transform (48) and (49) as

$$\int_c^{x_{t_l}^*} C_{cheb}(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l} = K \sum_{k=-\infty}^{+\infty} \sum_{n=1}^{\infty} \widehat{B}_k \alpha_n \widehat{T}_{n,k}[c, x_{t_l}^*] e^{i\frac{2\pi}{d-c} k \tilde{x}_{t_l}} : \quad \text{(call)} \qquad (54)$$

$$\int_{x_{t_l}^*}^d C_{cheb}(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l} = K \sum_{k=-\infty}^{+\infty} \sum_{n=1}^{\infty} \widehat{B}_k \alpha_n \widehat{T}_{n,k}[x_{t_l}^*, d] e^{i\frac{2\pi}{d-c} k \tilde{x}_{t_l}} : \quad \text{(put)}, \qquad (55)$$

respectively. By substituting (44), (45), (54), and (55) back into (42), we can obtain a CFS repre-

sentation of $C(x_{t_l}, K, t_l)$ such that

$$C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)}K \begin{cases} \sum_{k=-\infty}^{+\infty} \widehat{B}_k \left(\widehat{G}_k(x^*_{t_l}, d) + \sum_{n=1}^{\infty} \alpha_n \widehat{T}_{n,k}[c, x^*_{t_l}] \right) e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_l}} : & \text{(call)} \\ \sum_{k=-\infty}^{+\infty} \widehat{B}_k \left(\widehat{G}_k(c, x^*_{t_l}) + \sum_{n=1}^{\infty} \alpha_n \widehat{T}_{n,k}[x^*_{t_l}, d] \right) e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_l}} : & \text{(put)} \end{cases}. \tag{56}$$

We should note that the CFS representation above is working at each time step from t and t_{L-2}. However, at t_{L-1}, since $t_L = T$ and $V(y_T, T) = U(y_T, T)$, is a payoff function in (41), we simply have a CFS European pricing formula on $[c, d]$, i.e.,

$$C(x_{t_{L-1}}, K, t_{L-1}) = e^{-r(T-t_{L-1})}K \begin{cases} \sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{G}_k[0, d]\, e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_{L-1}}} : & \text{(call)} \\ \sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{G}_k[c, 0]\, e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_{L-1}}} : & \text{(put)} \end{cases}. \tag{57}$$

Finally, to seek an SFP representation of $C(x_t, K, t)$ at time t, we first denote

$$\widehat{\mathcal{G}}_k = \begin{cases} \widehat{G}_k(x^*_{t_l}, d) + \sum_{n=1}^{\infty} \alpha_n \widehat{T}_{n,k}[c, x^*_{t_l}] : & \text{(call)} \\ \widehat{G}_k(c, x^*_{t_l}) + \sum_{n=1}^{\infty} \alpha_n \widehat{T}_{n,k}[x^*_{t_l}, d] : & \text{(put)} \end{cases}. \tag{58}$$

By starting from T using (57) and then working backwards and recursively using (56) until t, we can reach

$$V(x_t, K, t) = C(x_t, K, t) = e^{-r(t_1-t)}K \left(2 \sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{\mathcal{G}}_k e^{i\frac{2\pi}{d-c}k\tilde{x}_t} \right). \tag{59}$$

Then, by following the step proposed in (32), we can further infer that

$$V(x_t, K, t) = e^{-r(t_1-t)}K \left(2 \sum_{k=1}^{\infty} \widehat{B}_k \widehat{\mathcal{G}}_k e^{i\frac{2\pi}{d-c}k\tilde{x}_t} + \widehat{B}_0 \widehat{\mathcal{G}}_0 \right). \tag{60}$$

Based on the equation above, we apply all the steps from (33) to (36); then, we can reach

$$V(x_t, K, t) = e^{-r(t_1-t)}K\mathfrak{Re} \left(\frac{P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z) \log\left(1 - z/\varepsilon_s\right)}{Q_M(z)} \right), \tag{61}$$

where $z = \exp\left(i\frac{2\pi}{d-c}\tilde{x}_t\right)$ and $\tilde{x}_t = x_t - \log K$.

To evaluate American options, one simple approach is to approximate an American option by a Bermudan option with many exercise opportunities L that go to infinity (cf. Fang and Oosterlee 2009b). An alternative approach is to use a Richardson extrapolation on a series of Bermudan options with an increasing number of exercise opportunities. This method is first introduced in Geske and Johnson (1984) and then described in detail in Chang *et al.* (2007). The CONV (Lord *et al.* 2008), COS (Fang and Oosterlee 2009b) and QUAD (Andricopoulos *et al.* 2003) methods all apply the same technique to price American options. In this paper, we adapt these two approaches to demonstrate the efficiency of our method. When we use the Richardson extrapolation, we im-

plement the 4-point Richardson extrapolation scheme proposed by Fang and Oosterlee (2009b). Accordingly, we have the American option price given by

$$V_{Amer}(L) = \frac{1}{21}\left(64V(2^{L+3}) - 56V(2^{L+2}) + 14V(2^{L+1}) - V(2^{L})\right), \tag{62}$$

where $V_{Amer}(L)$ denotes the approximated value of the American option.

Remark 1 We approximate $C(y_t, t_{l+1})$ with a Chebyshev series in (47) due to the achievement of a lower computational cost. According to Mason and Handscomb (2002), the complexity of calculating coefficient α_n of a Chebyshev series is $\mathcal{O}(\tilde{N}\log\tilde{N})$, where $\tilde{N}$ is the total number of Chebyshev terms. Moreover, as we also apply the FFC rules in (54) and (55), the complexity of the rules is also $\mathcal{O}(\tilde{N}\log\tilde{N})$ for each complex Fourier term k up to N (cf. Domínguez *et al.* 2011).

5.2. *Early-exercise point using root-finding techniques and a computational algorithm for the Bermudan option*

In this short section, we combine the SFP–FCC method with root-finding techniques, mainly Newton's method, to find early-exercise points. Newton's method is first proposed in Fang and Oosterlee (2009b) to find an early-exercise point. This technique can be used when one solves the following equality:

$$C(y_{t_l}, t_{l+1}) = U(y_{t_l}, t_{l+1}), \tag{63}$$

which appears in (42). Therefore, to find $x^*_{t_l}$, we can implement different root-finding techniques, such as the secant method. In this paper, as suggested in Fang and Oosterlee (2009b), we instead implement Newton's method (also known as the Newton-Raphson method). The process of this method is repeated as

$$x_{j+1} = x_j - \frac{U(y_{t_l}, t_{l+1}) - C(y_{t_l}, t_{l+1})}{\frac{\partial}{\partial y_{t_l}}U(y_{t_l}, t_{l+1}) - \frac{\partial}{\partial y_{t_l}}C(y_{t_l}, t_{l+1})} \tag{64}$$

over x_j for $j = 1, 2, \ldots$ until a sufficiently accurate value is reached. As we only determine whether $x^*_{t_l}$ lies on $[c, d]$, if not, we set $x^*_{t_l}$ to be equal to the nearest boundary point. In the equation, we start with x_0 equal to $x^*_{t_{l+1}}$, the exercise point in the exercise date at t_{l+1}, and we also know that at maturity, T, x^*_T is equal to 0. In (64),

$$C(y_{t_l}, t_{l+1}) = e^{-r(t_{l+2}-t_{l+1})}K\left(\mathfrak{Re}\left[2\sum_{k=1}^{\infty}\widehat{B}_k\widehat{\mathcal{G}}_k e^{i\frac{2\pi}{d-c}ky_{t_l}}\right]\right), \tag{65}$$

$$\frac{\partial C(y_{t_l}, t_{l+1})}{\partial y_{t_l}} = e^{-r(t_{l+2}-t_{l+1})}K\left(\mathfrak{Re}\left[2\sum_{k=1}^{\infty}\left(i\frac{2\pi}{d-c}k\right)\widehat{B}_k\widehat{\mathcal{G}}_k e^{i\frac{2\pi}{d-c}ky_{t_l}}\right]\right). \tag{66}$$

Since $C(y_{t_l}, t_{l+1})$ may suffer from the Gibbs phenomenon due to a piecewise continuous PDF, to avoid the phenomenon and achieve a higher accuracy of finding $x^*_{t_l}$, we apply the SFP method to (65) and (66). To obtain our SFP representation, we first let $z = \exp\left(i\frac{2\pi}{d-c}y_{t_l}\right)$ and then transform all the jumps ζ into $\varepsilon = \exp\left(i\frac{2\pi}{d-c}\zeta\right)$ in (65) and (66). Accordingly, this process transforms the

CFS representation into the form

$$f_1(z) = \begin{cases} 2\sum_{k=1}^{U} \widehat{B}_k \widehat{\mathcal{G}}_k z^k + \widehat{B}_0 \widehat{\mathcal{G}}_0, \\ 2\sum_{k=1}^{U} \left(i\frac{2\pi}{d-c} k \right) \widehat{B}_k \widehat{\mathcal{G}}_k z^k. \end{cases} \tag{67}$$

based on the equation above, by using (12), we can eventually obtain the SFP approximant given by

$$P_N(z) \sum_{s=1}^{\mathcal{S}} L_{N_s}(z) \log\left(1 - z/\varepsilon_s\right) = f_1(z) Q_M(z) + \mathcal{O}(z^{U+1}). \tag{68}$$

By applying the approximation algorithm in Appendix A to determine the coefficients of P_N, Q_M, and L_{N_s}, we can obtain the SPF formula for $C(y_{t_l}, t_{l+1})$ and $\frac{\partial}{\partial y_{t_l}} C(y_{t_l}, t_{l+1})$ with the form

$$e^{-r(t_1-t)-x_t} K \mathfrak{Re} \left(\frac{P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z) \log\left(1 - z/\varepsilon_s\right)}{Q_M(z)} \right). \tag{69}$$

By combining the root-finding techniques above and summarising Section 5.1, we present the pseudo-code of our algorithm that computes Bermudan option prices in Algorithm 1.

Finally, we draw our attention to the performance or complexity of the algorithm, $\mathcal{O}$, of the SFP–FCC method. At each time step t_l, since we adopt Chebfun (Trefethen *et al.* 2014) to calculate α_n without applying an adaptive process in (47), the complexity is $\mathcal{O}(\tilde{N} \log \tilde{N})$, where $\tilde{N}$ is the total number of Chebyshev terms because Chebfun employs the fast Fourier transfer (FFT) technique, which originated in Mason and Handscomb (2002), to calculate α_n. Furthermore, we apply the FFC rules in (54) and (55), so according to Domínguez *et al.* (2011), the complexity of the rules is also $\mathcal{O}(\tilde{N} \log \tilde{N})$ for each complex Fourier term k up to N. Combining the computational complexities above and considering L exercising dates, the total complexity of the SFP–FCC method is $\mathcal{O}((L-1)(N+1)(\tilde{N} \log \tilde{N}))$.

Remark 2 In (46), we can directly integrate both C and f^R together because they both have a CFS representation with a complex Fourier basis function $e^{-i\frac{2\pi}{d-c}k y_{t_l}}$; however, unfortunately, if we integrate them, our numerical results suggest that less accuracy can be obtained in the SFP framework.

5.3. *Pricing formulae for discretely monitored Barrier options*

A barrier option is an early-exercise option with payoff that depends on the stock price crossing a pre-set barrier level during the option's lifetime. We call the option an up-and-out, knock-out, or down-and-out option when the option's existence fades out after crossing the barrier level. Like European vanilla options, these options can all be written as either put or call contracts that have a pre-determined strike price on an expiration date. In this paper, we only investigate two basic types of barrier options: down-and-out barrier (DO) options and up-and-out barrier (UO) options for illustrations of our method.

(i) *Down-and-out barrier (DO) option*: A down-and-out barrier option is an option that can be exercised at a pre-set strike price on an expiration date as long as the stock price that drives the option does not go below a pre-set barrier level during the option's lifetime. As an illustration, if the stock price falls below the barrier, the option is "knocked-out" and immediately carries no value.

Result: Bermudan option price $V(x_t, K, t)$ at time t
initialisation;
discretise $[t, T]$ into timesteps $t = t_0, t_1, \ldots, t_l, \ldots, t_L = T$;
$t_l = t_{L-1}$;
compute $C(x_{t_{L-1}}, K, t_{L-1}) = e^{-r(T-t_{L-1})} K \mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{G}_k e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_{L-1}}}\right]$ stated in (57);
while $t_l \neq t$ **do**
  express $C(x_{t_l}, K, t_l)$ in the form of (42);
  find $\tilde{x}^*_{t_l}$ by using the root-finding technique in Section 5.2;
  compute $\int U(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\, \mathrm{d}y_{t_l}$ by using the steps from (42) to (45);
  compute $\int C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l})\, \mathrm{d}y_{t_l}$ by using the steps from (46) to (55);
  express $C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)} K \mathfrak{Re}\left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{\mathcal{G}}_k e^{i\frac{2\pi}{d-c}k\tilde{x}_{t_l}}\right]$ stated in (56);
  next t_l;
end
express $C(x_t, K, t) = V(x_t, K, t) = e^{-r(t_1-t)} K \mathfrak{Re}\left(\frac{P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z) \log(1 - z/\varepsilon_s)}{Q_M(z)}\right)$, where
$z = \exp\left(i\frac{2\pi}{d-c}\tilde{x}_t\right)$ and $\tilde{x}_t = x_t - \log K$, by using the steps from (60) to (61);

Algorithm 1: Algorithm for computing Bermudan option price $V(x_t, K, t)$ at t by using the SFP-FCC method.

(ii) *Up-and-out barrier (UO) option*: Similar to a down-and-out barrier option, an up-and-out barrier option will be knocked out when the stock price rises above the barrier level during the option's lifetime. Once it is knocked out, the option cannot be exercised at a predetermined strike price on an expiration date.

The structure of discretely monitored barrier options is the same as the structure of Bermudan options. Instead of having a pre-set exercise date and an early-exercise point like Bermudan options, barrier options have a pre-set monitored date and a barrier level. In the case of Bermudan options, when the stock price goes across the early-exercise point, a payoff occurs, and the option expires immediately. In the same manner, a barrier option is immediately knocked out when the barrier level is crossed. The barrier level acts exactly the same as the exercise point in Bermudan options. However, in the case of a barrier option without a rebate, no payoff occurs when the barrier level is reached; otherwise, a rebate occurs when a barrier option is knocked out.

In this paper, we only focus on a barrier option without a rebate and use a DO option to illustrate the SFP–FCC method to approximate discretely monitored barrier option prices. Suppose that we have a DO option driven by S_t with a barrier B, a strike K and a series of monitoring dates L: $t = t_0 < \ldots < t_l < \ldots < t_L = T$; the option formulae can be described as

$$V(x_{t_l}, K, t_l) = \begin{cases} U(e^{x_{t_l}}, K, t_l)\mathbb{1}_{x_{t_l} > \log B} & l = L,\ t_L = T \\ C(x_{t_l}, K, t_l)\mathbb{1}_{x_{t_l} > \log B} & l = 1, \ldots, L-1\ , \\ C(x_{t_l}, K, t_l) & l = 0 \end{cases} \tag{70}$$

where, $\mathbb{1}$ is an indicator function, $U(e^{x_{t_l}}, K, t_l)$ is again either a call or put payoff and

$$C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)} \int_c^d V(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l}. \tag{71}$$

We follow the steps from (41) and (42) in Section 5.1 and replace the exercise point $\tilde{x}^*_{t_l}$ with a

scaled log barrier, $\tilde{B} = \log B - \log K$. Accordingly, we can expand the equation into

$$C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)} \left(\int_{\tilde{B}}^{d} C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l} \right). \tag{72}$$

To compute $\int C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l}$, we follow the steps from (46) to (55) in Section 5.1. Therefore, we first approximate $C(y_{t_l}, t_{l+1})$ with a Chebyshev series $C_{cheb}(y_{t_l}, t_{l+1})$, such that

$$\int_{\tilde{B}}^{d} C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \mathrm{d}y_{t_l} = K \sum_{k=-\infty}^{+\infty} \sum_{n=1}^{\infty} \widehat{B}_k \alpha_n \widehat{T}_{n,k}[\tilde{B}, d] e^{i \frac{2\pi}{d-c} k \tilde{x}_{t_l}}. \tag{73}$$

By substituting (73) into (72), the CFS representation of $C(x_{t_l}, K, t_l)$ can be formulated as

$$C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)} K \sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{\mathcal{G}}_k \, e^{i \frac{2\pi}{d-c} k \tilde{x}_{t_l}}, \tag{74}$$

where $\widehat{\mathcal{G}}_k = \sum_{n=1}^{\infty} \alpha_n \widehat{T}_{n,k}[\tilde{B}, d]$. We have a different expression of $\widehat{\mathcal{G}}_k$ in $C(x_{t_{L-1}}, K, t_{L-1})$ at t_{L-1} as we do not apply the FCC rules to approximate a payoff function $U(e^{x_{t_L}}, K, t_L)$; therefore, we have

$$C(x_{t_{L-1}}, K, t_{L-1}) = e^{-r(t_{l+1}-t_l)} K \sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{\mathcal{G}}_k \, e^{i \frac{2\pi}{d-c} k \tilde{x}_{t_l}}, \tag{75}$$

where $\widehat{\mathcal{G}}_k = \widehat{G}_k$ and $\widehat{G}_k$ is either the Fourier transform of a call payoff on $[\tilde{B}, d]$ (cf. [29]) or a put payoff on $[\tilde{B}, 0]$ (cf. [30]). Finally, to obtain the SFP–FCC pricing formula of the DO barrier option, we work backwards and recursively from T to t by using (74) and (75) and then approximate $C(x_t, K, t)$ with the SFP approximant at t by applying the steps of (60) and (61) in Section 5.1. We present the pseudo-code of our algorithm computing DO option prices in Algorithm 2.

Result: discretely monitored barrier option price $V(x_t, K, t)$ at time t
initialisation;
discretise $[t, T]$ into timesteps $t = t_0, t_1, \ldots, t_l, \ldots, t_L = T$;
compute $C(x_{t_{L-1}}, K, t_{L-1}) = e^{-r(T-t_{L-1})} K \mathfrak{Re} \left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{\mathcal{G}}_k e^{i \frac{2\pi}{d-c} k \tilde{x}_{t_{L-1}}} \right]$ stated in (75);
while $t_l \neq t$ **do**
  express $C(x_{t_l}, K, t_l)$ in the form of (74);
  compute $\int C(y_{t_l}, t_{l+1}) f^R(\tilde{x}_{t_l} - y_{t_l}) \, \mathrm{d}y_{t_l}$ as stated in (72);
  express $C(x_{t_l}, K, t_l) = e^{-r(t_{l+1}-t_l)} K \mathfrak{Re} \left[\sum_{k=-\infty}^{+\infty} \widehat{B}_k \widehat{\mathcal{G}}_k e^{i \frac{2\pi}{d-c} k \tilde{x}_{t_l}} \right]$ as stated in (74);
  next t_l;
end
express $C(x_t, K, t) = V(x_t, K, t) = e^{-r(t_1-t)} K \mathfrak{Re} \left(\frac{P_N(z) + \sum_{s=1}^{S} L_{N_s}(z) \log(1 - z/\varepsilon_s)}{Q_M(z)} \right)$, where
$z = \exp\left(i \frac{2\pi}{d-c} \tilde{x}_t \right)$ and $\tilde{x}_t = x_t - \log K$, using the steps from (60) to (61);

Algorithm 2: Algorithm for computing discretely monitored DO barrier option price $V(x_t, K, t)$ at time t by using the SFP–FCC method.

For the UO barrier options, we can modify Algorithm 2 to compute their prices, but we consider

the condition of the option knocked out when the stock price rises above B, i.e.,

$$V(x_{t_l}, K, t_l) = \begin{cases} U(e^{x_{t_l}}, K, t_l)\mathbb{1}_{x_{t_l}<\log B} & l = L,\ t_L = T \\ C(x_{t_l}, K, t_l)\mathbb{1}_{x_{t_l}<\log B} & l = 1, \ldots, L-1\ . \\ C(x_{t_l}, K, t_l) & l = 0 \end{cases} \tag{76}$$

6. Option Greek hedging and choice of truncated intervals

This section is divided into two parts: calculating the option Greeks and choosing truncated intervals. As we have previously mentioned in Chan (2018), we repeat the derivation of only two option Greeks—Delta and Gamma. Other Greeks, such as Theta, can be derived in a similar fashion; however, depending on the characteristic function, the derivation expression might be rather lengthy. We omit them here, as many terms are repeated. We use the Bermudan option defined in (60) as an illustration to derive the Greeks since the derivation for other option Greeks is equivalent.

Delta is the first derivative of the value of V of the option with respect to the underlying instrument price S. Therefore, differentiating the CFS expansion of V (60) with respect to S, we have

$$\begin{aligned} \Delta_t = \frac{\partial V(x_t, K, t)}{\partial S} &= \frac{\partial V(x_t, K, t)}{\partial x}\frac{\partial x}{\partial S} \\ &= e^{-r(t_1-t)-x_t} K \left(\mathfrak{Re}\left[2\sum_{k=1}^{\infty} \left(i\frac{2\pi}{d-c}k \right) \widehat{B}_k \widehat{\mathcal{G}}_k e^{i\frac{2\pi}{d-c}k\tilde{x}_t} \right] \right). \end{aligned} \tag{77}$$

where $\tilde{x}_t = x_t - \log K$. Similarly, we can obtain Γ_t by differentiating Δ_t with respect to S such that

$$\Gamma_t = \frac{\partial^2 V(x_t, K, t)}{\partial S^2} = \frac{\partial \Delta_t}{\partial S} = \frac{\partial \Delta_t}{\partial x_t}\frac{\partial x_t}{\partial S}, \tag{78}$$

and eventually,

$$\Gamma_t = e^{-r(t_1-t)-2x_t} K \mathfrak{Re}\left[2\sum_{k=1}^{\infty} \left(i\frac{2\pi}{d-c}k \right) \left(i\frac{2\pi}{d-c}k - 1 \right) \widehat{B}_k \widehat{\mathcal{G}}_k e^{i\frac{2\pi}{d-c}k\tilde{x}_t} \right].$$

To obtain our first SFP representation of Δ, we first let $z = \exp\left(i\frac{2\pi}{d-c}\tilde{x}_t\right)$ and then transform every jump ζ_s in Δ_t into $\varepsilon_s = \exp\left(i\frac{2\pi}{d-c}\zeta_s\right)$ in (77). Accordingly, this transforms the CFS representation in (77) into the form

$$f_1(z) = 2\sum_{k=1}^{U} \left(i\frac{2\pi}{d-c}k \right) \widehat{B}_k \widehat{\mathcal{G}}_k z^k. \tag{79}$$

and based on the equation above, by using (12), we can eventually obtain the SFP approximant given by

$$P_N(z)\sum_{s=1}^{\mathcal{S}} L_{N_s}(z)\log\left(1 - z/\varepsilon_s\right) = f_1(z)Q_M(z) + \mathcal{O}(z^{U+1}). \tag{80}$$

By applying the approximation algorithm in Appendix A to determine the coefficients of P_N, Q_M, and L_{N_s}, we can obtain the SPF formula for Δ_t with the form

$$e^{-r(t_1-t)-x_t} K \mathfrak{Re} \left(\frac{P_N(z) + \sum_{s=1}^{\mathcal{S}} L_{N_s}(z) \log\left(1 - z/\varepsilon_s\right)}{Q_M(z)} \right). \tag{81}$$

To determine the SFP approximant of Γ_t, we follow the same idea of approximating Δ_t but replace $f_1(z)$ with

$$2 \sum_{k=1}^{U} \left(i \frac{2\pi}{d-c} k \right) \left(i \frac{2\pi}{d-c} k - 1 \right) \widehat{B}_k \widehat{\mathcal{G}}_k z^k. \tag{82}$$

We now draw our attention to wisely choosing a good truncated interval. The choice of the interval $[c, d]$ plays a crucial role in the accuracy of the SFP–FCC method. A minimum and substantial interval $[c, d]$ can capture most of the mass of a PDF such that our algorithm can, in turn, produce a sensible global spectral convergence rate. We adopt the ideas of Fang and Oosterlee (2009a) and Chan (2018) to choose the interval $[c, d]$. In this short section, we show how to construct an interval related to the closed-form formulas of stochastic process cumulants. The idea of using the cumulants was first proposed by Fang and Oosterlee (2009a) to construct the definite interval $[c, d]$ in (20). Based on their ideas, we have the following expression for $[c, d]$:

$$\begin{aligned} d &= \left| c_1 + \tilde{L} \sqrt{c_2 + \sqrt{c_4}} \right| \\ c &= -d, \end{aligned} \tag{83}$$

where c_1, c_2, and c_4 are the first, second and fourth cumulants, respectively, of the stochastic process and $\tilde{L} \in [8, 12]$. For simple and less-complicated financial models, we also obtain closed-form formulas for c_1, c_2, and c_4, which are shown in Table D1 of Appendix D.

7. Numerical results

The main purpose of this section is to test the accuracy and efficiency of the SFP–FCC method through various numerical tests. This process involves evaluating the ability of the method to price any early-exercise options and to exhibit good accuracy even when the PDF is smooth/non-smooth. A number of popular numerical methods are implemented to compare the algorithm in terms of the error convergence and computational time. These methods include the COS method (a Fourier COS series method, Fang and Oosterlee 2009a), the filter-COS method (a COS method with an exponential filter to resolve the Gibbs phenomenon; see Ruijter *et al.* 2015), the CONV method (an FFT method, Lord *et al.* 2008), the FFT–QUAD (a combination of the quadrature and CONV methods; see O'Sullivan 2005), and the SWIFT methods (wavelet-based methods; see Ortiz-Gracia and Oosterlee 2013, Maree 2015, Ortiz-Gracia and Oosterlee 2016, Maree *et al.* 2017). When we implement CONV, we use Simpson's rule for the Fourier integrals to achieve fourth-order accuracy. In the filter-COS method, we use an exponential filter and set the accuracy parameter to 10, as Ruijter *et al.* (2015) report that this filter provides better algebraic convergence than other options. We also set the damping factors of the CONV to 0 for pricing European options.

As the SFP method requests approximating jumps in logarithmic series, we only consider and apply the endpoints c and d as our two known jumps for all non-smooth/smooth PDFs. In all numerical experiments, we use parameter U to denote the number of terms of the SFP–FCC method, $\tilde{N}$ to denote the number terms of the Chebyshev polynomials and N to denote the number

of terms/grid points of the other variables. When we measure the approximation errors of the numerical methods, we use absolute errors, the infinity norm errors R_∞ and the L_2 norm errors R_2 as the measurement units. A MacBook Pro with a 2.8 GHz Intel Core i7 CPU and two 8 GB DDR SDRAM (cache memory) is used for all experiments. Finally, the code is written in MATLAB, and the codes to implement the COS method and the FFT method, such as the CONV method and the like, are retrieved from von Sydow *et al.* (2015). In terms of computing the Chebyshev polynomials, we use Chebfun (Trefethen *et al.* 2014) to generate non-adaptive Chebyshev polynomials.

We consider four different test cases based on the following PDFs and other parameters:

$$\begin{aligned}
\textbf{VG1}: & \ S = 80 - 120, K = 90, \sigma = 0.12, \theta = -0.14, \nu = 0.2, \\
& \ T = 0.1, r = 0.1, q = 0. && (84)\\
\textbf{CGMY1}: & \ S = 0.5 - 1.5, K = 1, C = 1, G = 5, M = 5, Y = 0.5, \\
& \ T = 1, r = 0.1, q = 0.0. && (85)\\
\textbf{CGMY2}: & \ S = 80 - 120, K = 100, C = 4, G = 50, M = 60, Y = 0.7, \\
& \ T = 1, r = 0.05, q = 0.02. && (86)\\
\textbf{NIG1}: & \ S = 100, K = 80 - 120, \alpha = 15, \beta = -5, \delta = 0.5, T = 1, \\
& \ r = 0.05, q = 0.02. && (87)
\end{aligned}$$

In each set of parameters, VG denotes the variance gamma model (e.g. Madan *et al.* 1998, Madan and Milne 1991), CGMY stands for the Carr-German-Madan-Yor model (Carr *et al.* 2002), and NIG is short for the normal inverse Gaussian process (Barndorff-Nielsen 1991).

Throughout all the numerical tests in this paper, we set $\tilde{L} = 8$ in (83) to obtain an accurate truncated interval for the (filter-)COS, SFP–FCC and SWIFT methods. In the first test, we discuss the behaviour of the error and the stability of the SFP–FCC method if M, the number of early-exercise dates, goes to infinity. We also check how the Bermudan option prices converge to their American option counterparts. When M approaches infinity, this leads to Δt going to zero and to the eventual formation of a highly peaked PDF. The **VG1** is chosen for the test because relatively slow convergence was reported for the CONV method for very short maturities in Lord *et al.* (2008). In the test, the Bermudan call options without paying dividends have the same values as their European counterparts, and the European call reference prices are generated by using the SFP method (Chan 2018). In Fig. 1, the left-hand side of the graph shows highly peaked PDFs with $\Delta t = 0.1$ and $\Delta t = 1e^{-05}$, and the right-hand side of the graph demonstrates the logarithm absolute error of the SFP–FCC method. As we gradually increase M from 100 to 10000 (equivalent to a decrease in Δt from 0.001 to 1^{-05}) and keep both $U = 32$ and $\tilde{N} = 128$ fixed, the logarithm absolute error remains almost equivalent throughout on the right-hand side of the graph. This phenomenon indicates that the SFP–FCC method works stably to steadily converge Bermudan option prices to their American option counterparts and yields a spectral convergence rate apart from the jump (the highly peaked point). In the next test shown in Fig. 2, we compare the filter–COS, CONV, and FFT–QUAD methods with the SFP–FCC method for pricing a Bermudan call option with the same input parameters,**VG1**. In the SFP–FCC method, we set L to 1000 (equivalent to $\Delta t = 1^{-04}$) and gradually increase U in a sequence of 8 (blue), 16 (red) and 32 (yellow), and $\tilde{N}$ is set to 128 for the SFP–FCC method. For the remaining three methods, N is ascended in a sequence of 128 (blue), 256 (red) and 512 (yellow). We compute 401 Bermudan call option prices in the range of S from 80 to 120 and $K = 90$. Compared with the other methods, we observe that the SFP–FCC method can retain spectral convergence apart from the jump, the highly peaked point, and yield a higher accuracy than the other methods with fewer summation terms required.

In Table 1, we compare the accuracy of the SFP–FCC method with the COS method in pricing an American put option under the CGMY model after applying the Richardson extrapolation technique (62) to them. We use **CGMY1** retrieved from Fang and Oosterlee (2009b) for the

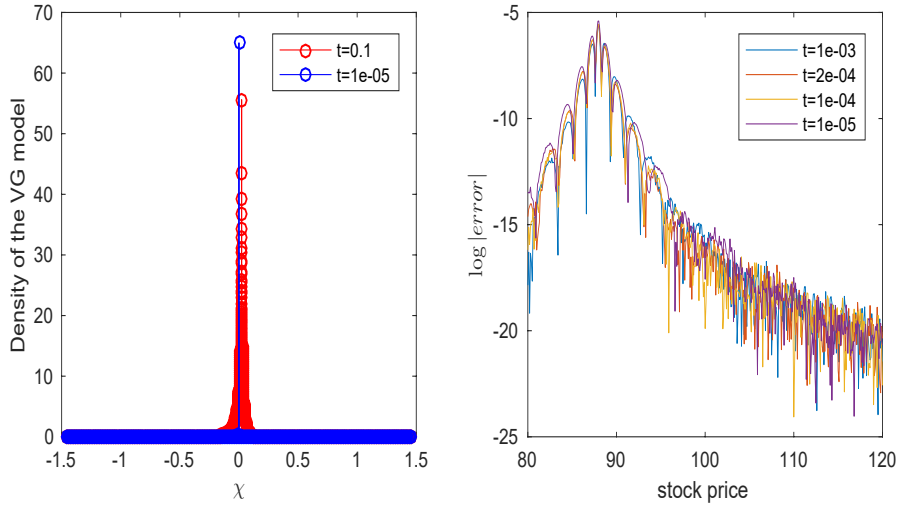

Figure 1. Density functions (left) of the VG model and the logarithm absolute errors (right) of the SFP–FCC method with parameters taken from **VG1**. L is gradually increased in a sequence of 100 ($\Delta t = 1^{-03}$), 500 ($\Delta t = 2^{-04}$), 1000 ($\Delta t = 1^{-04}$) and 10000 ($\Delta t = 1^{-05}$), and both U and $\tilde{N}$ are equal to 32 and 128, respectively. $\tilde{L} = 8$. Four hundreds-one Bermudan call option prices are computed in the range of S from 80 to 120, and K is equal to 90.

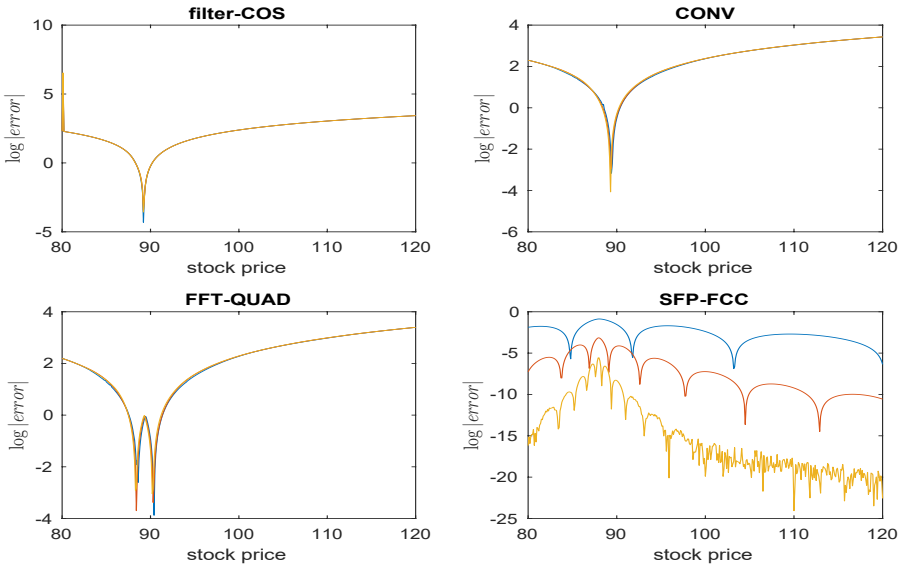

Figure 2. Comparison of the filter–COS, CONV, FFT–QUAD and SFP–FCC methods for pricing a Bermudan call option under the VG model with parameters taken from **VG1**. L is set to 1000 (equivalent to $\Delta t = 1^{-04}$). U is gradually increased in a sequence of 8 (blue), 16 (red) and 32 (yellow), and $\tilde{N}$ is set to 128 for the SFP–FCC method. N is ascended in a sequence of 128 (blue), 256 (red) and 512 (yellow) for the other three methods. 401 Bermudan call option prices are computed in the range of S from 80 to 120, and K is equal to 90. Apart from the jump, a singularity causing a pole in a complex plane, spectral convergence is observed in the SFP–FCC method.

test. The test itself is a replicate of the same test in Fang and Oosterlee (2009b, Table 3). 14 reference values are computed by using the CONV method with $N = 4096$ and applying the same extrapolation technique to a range of S from 0.5 to 1.5, and K equals 1. In Table 1, we increase L from 0 to 3, and we can infer that the SFP–FCC method can achieve relatively better accuracy than the COS method with a reduced total number of $U = 256$ and $\tilde{N} = 128$ than $N = 512$ required. By using the same input parameters of **CGMY1**, we examine the stability of the SFP–FCC method when $\tilde{N}$ increases in Table 2. We increase $\tilde{N}$ twice from 64 to 512 and keep $U = 256$ and $L = 2$ the same, and both R_∞ and R_2 errors first decrease and then level off.

Table 1. Comparison of the COS and SFP–FCC methods for pricing an American put option under the CGMY model with parameters taken from **CGMY1**; Fourteen option prices are computed for both methods in a range of S from 0.5 to 1.5, and K is equal to 1.

L in Eq. (62)	**COS**				**SFP–FCC**				
	N	R_∞	R_2	Time (sec.)	U	$\tilde{N}$	R_∞	R_2	Time (sec.)
0	512	4.182e-02	2.717e-01	0.896	256	128	3.180e-02	1.797e-01	0.731
1	512	1.123e-03	9.034e-03	1.528	256	128	1.580e-03	9.614e-03	1.430
2	512	2.629e-04	2.011e-03	3.066	256	128	1.659e-05	1.011e-04	3.021
3	512	2.667e-05	2.021e-04	6.164	256	128	1.670e-05	1.021e-04	6.182

Table 2. Comparison of the R_∞ and R_2 errors of the SFP–FCC method for pricing an American put option under the CGMY model with parameters taken from **CGMY1** when $\tilde{N}$ increases and L and U are kept the same. Fourteen option prices are computed in a range of S from 0.5 to 1.5, and K is equal to 1.

L in Eq. (62)	**SFP–FCC**				
	U	$\tilde{N}$	R_∞	R_2	Time (sec.)
2	256	64	3.180e-03	1.114e-02	1.530
2	256	128	1.659e-05	1.011e-04	3.021
2	256	256	1.670e-05	1.021e-04	5.282
2	256	512	1.670e-05	1.021e-04	10.082

In the final two tests, we focus on the comparison of the SFP–FCC method with the SWIFT and COS methods in pricing the UO and DO barrier options, respectively. We set L equal to 12, and both **CGMY2** and **NIG1** are taken from Fang and Oosterlee (2009b). All the reference values are generated by using the CONLeg method–the Convolution of Legendre Series (Chan and Hale 2019). In Tables 3 and 4, the difference in computational time across methods is not large. In Table 3, we first compare the accuracy of the SFP–FCC method with the SWIFT method under the CGMY model. In the table, we can see that both methods can reach spectral convergence and are comparable to each other when we compare 41 UO option prices in the range of S from 80 to 120, K equal to 100, and setting the barrier level B to 120. Finally, when pricing the DO barrier options shown in Table 4 under the NIG model, both methods–COS and SFP–FFC–can obtain spectral convergence when we compare 80 option prices in the range of K from 80 to 120, $S = 100$ and $B = 80$. However, the SFP–FCC method can obtain much lower R_∞ and R_2 errors than the COS method when both N and U are doubled. This result indicates that the SFP–FCC method is superior to the COS method.

Table 3. Comparison of the SWIFT and SFP–FCC methods for pricing daily-monitored ($L = 12$) UO call and UO put under the CGMY model with parameters taken from **CGMY2**. Forty-one option prices are computed for both methods in the range of S from 80 to 120, and K is equal to 100. The barrier level B is equal to 120. Spectral convergence is observed in both methods.

	SWIFT				**SFP–FCC**				
	scale	R_∞	R_2	Time (sec.)	U	$\tilde{N}$	R_∞	R_2	Time (sec.)
UO Call	2	6.419e-01	2.522	0.208	8	128	3.439e-01	8.022e-01	0.512
	3	3.344e-02	1.391e-01	0.256	16	128	6.114e-02	2.398e-01	0.856
	4	6.710e-04	3.231e-03	0.324	32	128	1.220e-04	4.568e-04	0.882
	5	1.287e-07	4.560e-06	0.451	64	128	3.187e-09	1.260e-08	0.911
	6	1.561e-12	4.850e-12	0.761	128	128	1.769e-12	5.050e-12	1.071
UO Put	2	1.313	7.307	0.206	8	128	3.353e-01	9.707e-01	0.123
	3	2.115e-02	5.742e-02	0.264	16	128	1.185e-02	4.842e-02	0.251
	4	5.613e-03	2.964e-02	0.336	32	128	4.663e-05	1.964e-04	0.321
	5	7.178e-07	3.721e-06	0.472	64	128	6.078e-11	2.724e-10	0.425
	6	2.021e-12	8.234e-12	0.761	128	128	1.825e-13	7.825e-13	0.543

Table 4. Comparison of the COS and SFP–FCC methods for pricing daily-monitored ($L = 12$) DO call and DO put under the NIG model with parameters taken from **NIG1**. Eighty option prices are computed for both methods in the range of K from 80 to 120, and S is equal to 100. The barrier level B is equal to 80. Spectral convergence is observed in both methods.

	COS				**SFP–FCC**				
	N	R_∞	R_2	Time (sec.)	U	$\tilde{N}$	R_∞	R_2	Time (sec.)
DO Call	64	1.965e-02	5.741e-02	0.691	64	256	2.837e-03	1.382e-02	0.551
	128	1.571e-03	4.244e-03	0.876	128	256	2.905e-05	1.364e-04	0.651
	256	1.532e-05	4.138e-05	1.181	256	256	6.871e-08	1.418e-07	0.761
	512	3.29e-09	7.867e-09	1.591	512	256	5.351e-10	3.285e-09	1.282
DO Put	64	4.212e-02	1.246e-01	0.681	64	256	3.104e-04	1.179e-03	0.701
	128	2.632e-03	7.166e-03	0.712	128	256	1.479e-05	8.387e-05	0.822
	256	2.811e-05	7.358e-05	1.060	256	256	2.566e-09	1.469e-08	0.981
	512	5.705e-09	1.326e-08	1.460	512	256	6.377e-13	9.154e-13	1.350

8. Conclusions

We have generalised the SFP option pricing method, based on a singular Fourier–Padé series to price and hedge early-exercise options–Bermudan, American and discretely-monitored barrier options. We call the new method SFP–FCC, as we incorporate the SFP method with the Filon–Clenshaw–Curtis (FCC) rules. The main advantages of the SFP–FCC method are its ability to return the price and Greeks as a function defined on a prescribed interval rather than just point values and its ability to retain spectral convergence under any process with a (piecewise) continuous PDF. The complexity of the new method is $\mathcal{O}((L-1)(N+1)(\tilde{N}\log\tilde{N}))$, and the method itself is shown to be either comparable or favourable to existing popular techniques in all numerical experiments.

Future research on the method will aim to theoretically validate the spectral convergence for early-exercise options and extend the method to price options with path-dependant features under the (time-changed) Lévy process. Research in this direction is already underway and will be presented in a forthcoming manuscript.

Appendix A: Computation of the singular Fourier-Padé coefficients

The approach to computing the polynomial coefficients needed in the SFP method is fairly straightforward. To demonstrate the algorithm, we focus on a simple case where the option pricing and Greek formulae are infinitely smooth apart from the jumps located at the endpoints c and d. As we consider $z = \exp\left(i\frac{2\pi}{d-c}\tilde{x}\right)$ in either the option pricing formula or the Greek formula, the jump of c and d in the z-plane is -1. For the sake of simplicity, we denote $f_1(z)$ as the CFS representation of any European-style pricing formula or its option Greek formula. With some superscripts dropped for clarity and knowing that $s = 1$, in (12), we have

$$P_N(z) + L_{N_1}(z)\log\left(1 - \frac{z}{\varepsilon_1}\right) = f_1(z)Q_M(z) + \mathcal{O}(z^{U+1}), \tag{A1}$$

where $N + M + N_1 = U$. Both L_{N_1} and $f_1(z)$ have Taylor series and CFS expansions, respectively, to determine U; therefore, their expansions are

$$\log\left(1 - \frac{z}{\varepsilon_s}\right) = \sum_{k=1}^{U} -\frac{z^k}{\varepsilon_1^k} + 0 \tag{A2}$$

$$f_1(z) = 2\sum_{k=1}^{U} \widehat{B}_k \widehat{G}_k z^k + \widehat{B}_0 \widehat{G}_0. \tag{A3}$$

Our goal is to derive a linear system for the unknown polynomial coefficients. Note that $Q_M(z)$ and $L_{N_1}(z)$ are determined only by terms of an order greater than N. Accordingly, we seek a linear solution to

$$\left[\widehat{B}\widehat{G} - L\right] \begin{bmatrix} \mathbf{q} \\ \mathbf{l} \end{bmatrix} = \mathbf{0}. \tag{A4}$$

Here, $\widehat{B}\widehat{G}$ is the $(M + N_1 + 1) \times (M + 1)$ Toeplitz matrix

$$\begin{bmatrix} \widehat{B}_{\frac{U}{2}+1}\widehat{G}_{\frac{U}{2}+1} & \widehat{B}_{\frac{U}{2}}\widehat{G}_{\frac{U}{2}} & \cdots & \widehat{B}_1\widehat{G}_1 \\ \widehat{B}_{\frac{U}{2}+2}\widehat{G}_{\frac{U}{2}+2} & \widehat{B}_{\frac{U}{2}+1}\widehat{G}_{\frac{U}{2}+1} & \ddots & \widehat{B}_2\widehat{G}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{B}_U\widehat{G}_U & \widehat{B}_{U-1}\widehat{G}_{U-1} & \cdots & \widehat{B}_{\frac{U}{2}}\widehat{G}_{\frac{U}{2}}, \end{bmatrix} \tag{A5}$$

and L is the $(M + N_1 + 1) \times (N_1 + 1)$ matrix defined similarly by using the Taylor coefficients of log(1+z). The vectors $\mathbf{q} = \{q_m\}_{m=0}^{M}$ and $\mathbf{l} = \{l_{n_1}\}_{n_1=0}^{N_1}$ hold the unknown polynomial coefficients in order of increasing degree. As the column dimension of the matrix in (A4) is one greater than its row dimension, we can conclude that there is one nonzero solution to (A4). In many cases, this can be made into a square system by choosing, for example, $q_0 = 1$. However, if one does not want to assume that any particular coefficient is nonzero, one can solve (A4) by a singular value decomposition. Finally, the unknown coefficients of $\mathbf{p} = \{p_n\}_{n=1}^{N}$ can be obtained by multiplication through the following matrix system:

$$\mathbf{p} = \begin{bmatrix} \widehat{B}_0\widehat{G}_0 & & & \\ \widehat{B}_1\widehat{G}_1 & \widehat{B}_0\widehat{G}_0 & & \\ \vdots & \ddots & \ddots & \\ \widehat{B}_{\frac{U}{2}}\widehat{G}_{\frac{U}{2}} & \cdots & \cdots & \widehat{B}_0\widehat{G}_0 \end{bmatrix} \mathbf{q} - \begin{bmatrix} l_0 & & & \\ l_1 & l_0 & & \\ \vdots & \ddots & \ddots & \\ l_{\frac{U}{2}} & \cdots & \cdots & l_0 \end{bmatrix} \mathbf{l}. \tag{A6}$$

If there is more than one jump in the option pricing/Greek curve (A1), this suggests the following modification of the equation:

$$P_N(z) + L_{N_1}(z)\log\left(1 - \frac{z}{\varepsilon_1}\right) + \ldots + L_{N_s}(z)\log\left(1 - \frac{z}{\varepsilon_{\mathcal{S}}}\right) = f_1(z)Q_M(z) + \mathcal{O}(z^{U+1}). \tag{A7}$$

Accordingly, we must modify (A4) to produce a new L matrix and a vector of coefficients for each location to reflect the changes. According to Driscoll and Fornberg (2001, 2011), there is no rigorous optimal formula for choosing the degrees M, N, and N_1,…,N_s. Because the denominator polynomial Q_M is shared, we allow M to be the largest, with the others being equal as much as

possible. For the case of just one jump, taking N at roughly 40% of the total available degrees of freedom seems to work well. Experiments suggest that these choices can affect the observed accuracy, occasionally by as much as an order of magnitude, but on average, there is little variation within a broad range of choices.

Appendix B: Locating jumps in probability density functions

Many PDFs (cf. Fig. B1) of interest are not smooth but piecewise smooth. If the locations of all jumps are not known in advance in the PDFs, we can also use Fourier-Padé ideas (cf. Driscoll and Fornberg 2011, Chan 2018) to estimate the locations of jumps sufficiently well to allow good reconstruction nearly everywhere in the interval $[c, d]$.

Here, g is approximated by $\sum_{k=0}^{M+N} b_k x^k$,. To obtain the approximant $R(N, M)$, we simply calculate the coefficients of polynomials P_N and Q_M by solving a system of linear equations. To obtain $\{q_m\}_{m=0}^{M}$, we first normalise $q_0 = 1$ to ensure that the system is well determined and has a unique solution in (4). Then, we consider the coefficients for $x^{N+1}, \ldots, x^{M+N}$, and we can yield a Toeplitz*[1] linear system:

$$\begin{bmatrix} b_{N+1} & b_N & b_{N-1} & \cdots & b_{N+1-M} \\ b_{N+2} & b_{N+1} & b_N & \ddots & b_{N+2-M} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ b_{N+M} & \cdots & b_{N+2} & b_{N+1} & b_N \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_M \end{bmatrix} = 0. \tag{B1}$$

Once $\{q_m\}_{m=0}^{M}$ is known, $\{p_n\}_{n=0}^{N}$ is found through the terms of order N and less in (4). This yields $\underline{p} = B\underline{q}$, where $b_{ij} = b_{i-j}$. For example, if $N = M$, one obtains

$$\begin{bmatrix} p_0 \\ p_1 \\ \vdots \\ p_N \end{bmatrix} = \begin{bmatrix} b_0 & & & \\ b_1 & b_0 & & \\ \vdots & \ddots & \ddots & \\ b_N & \cdots & b_1 & b_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_M \end{bmatrix}. \tag{B2}$$

Now, assuming g is a PDF, to find the jumps in g and to express g in a Fourier-Padé series, we first express g with the CFS representation:

$$\mathfrak{Re}\left[2\sum_{k=1}^{\infty} \varphi\left(\frac{2\pi}{d-c}k\right) e^{-i\frac{2\pi}{d-c}kx} + \varphi(0)\right]. \tag{B3}$$

Then, we can differentiate (B3) with respect to x to obtain

$$\mathfrak{Re}\left[2\sum_{k=1}^{\infty} -\left(i\frac{2\pi}{d-c}k\right) \varphi\left(\frac{2\pi}{d-c}k\right) e^{-i\frac{2\pi}{d-c}kx}\right]. \tag{B4}$$

Finally, we let $z = \exp\left(i\frac{2\pi}{d-c}x\right)$ in the two equations above, and they are ready for the Fourier-Padé approximation. In general, when the PDF has a jump, the sharp-peaked jump point will have an

[1] A Toeplitz matrix or diagonal-constant matrix is an invertible matrix in which each descending diagonal from left to right is constant.

enormously large value after differentiation. In other words, Fig. B1 is a graphical illustration of the outlooks of the PDF (left) and the first derivative (right) of the VG model after the Fourier-Padé approximation. In the figure, we can see that the non-smooth PDF with a jump can produce a value of 10×10^{11} at the jump point after the first derivative.

Appendix C: Accurate computation of the weights

We adopt Domínguez *et al.* (2011)s' algorithm to compute

$$w_n(\tilde{k}) := \int_{-1}^{+1} T_n(s) \exp(i\tilde{k}s)\mathrm{d}s, \quad n \geq 0. \tag{C1}$$

For the sake of clear mathematical notations, we finally assume the total number of a Chebyshev series as described in (C1), which is N in this section.

C.1. *Algorithm: for $n \leq N \leq \tilde{k}$ (first phase)*

First, based on the idea of $U_n = 1/(n+1)T'_{n+1}$ (cf. Abramowitz and Stegun 1965, Eq. (22.5.8)), where U_n is the nth Chebyshev polynomial of the second kind, we can see that

$$\rho_n(\tilde{k}) := \int_{-1}^{+1} U_{n-1}(s) \exp(i\tilde{k}s)\mathrm{d}s = \frac{1}{n}\int_{-1}^{+1} T'_n(s) \exp(i\tilde{k}s)\mathrm{d}s. \tag{C2}$$

Then, according to Domínguez *et al.* (2011, Section 4), their computation algorithm leads to

$$w_n(\tilde{k}) := \gamma_n(\tilde{k}) - \frac{n}{i\tilde{k}}\rho_n(\tilde{k}), \quad n \geq 1, \quad w_0(\tilde{k}) := \gamma_0(\tilde{k}). \tag{C3}$$

Here,

$$\gamma_n(\tilde{k}) = \begin{cases} \frac{2\sin\tilde{k}}{\tilde{k}} & \text{for even } n \\ \frac{2\cos\tilde{k}}{\tilde{k}} & \text{for odd } n \end{cases}, \quad \gamma_0(\tilde{k}) = \frac{1}{i\tilde{k}}\left(\exp(i\tilde{k}) - \exp(-i\tilde{k})\right), \tag{C4}$$

and $\rho_n(\tilde{k})$ can be determined based on the recurrence relationship,

$$2\gamma_n(\tilde{k}) - \frac{2n}{i\tilde{k}}\rho_n(\tilde{k}) = \rho_{n+1}(\tilde{k}) - \rho_{n-1}(\tilde{k}), \quad n \geq 2, \tag{C5}$$

with

$$\rho_0(\tilde{k}) := \gamma_0(\tilde{k}) \text{ and } \rho_2(\tilde{k}) := 2\gamma_1(\tilde{k}) - \frac{2}{i\tilde{k}}\gamma_0(\tilde{k}), \tag{C6}$$

If $n \leq N \leq \tilde{k}$, by using (C4) for computing $\gamma_n(\tilde{k})$ and (C5) and (C6) as a forward recurrence for $\rho_n(\tilde{k})$, we can stably obtain a vector of $\{w_n(\tilde{k})\}_{n=0}^{N}$. We summarise the computation in Algorithm 3. According to Domínguez *et al.* (2011, Theorem 5.1 and Corollary 5.2), the stability for $n \leq N \leq \tilde{k}$ is proved. However, the algorithm becomes unstable when $n \geq \tilde{k}$ and $n \leq \tilde{k} \leq N$.

1: Compute

$$\rho_1(\tilde{k}) := \gamma_0(\tilde{k}), \tag{C7}$$

$$\rho_2(\tilde{k}) := 2\gamma_1(\tilde{k}) - \frac{2}{i\tilde{k}}\gamma_0(\tilde{k}), \tag{C8}$$

$$\rho_{n+1}(\tilde{k}) := 2\gamma_n(\tilde{k}) - \frac{2}{i\tilde{k}}\gamma_n(\tilde{k}) + \rho_{n-1}(\tilde{k}), \quad n = 2, \ldots, N-1, \quad N \leq \tilde{k}. \tag{C9}$$

2: Set

$$w_n(\tilde{k}) := \gamma_n(\tilde{k}) - \frac{n}{ik}\rho_n(\tilde{k}), \quad w_0(\tilde{k}) := \gamma_0(\tilde{k}), \quad n = 1, 2, \ldots, N, \quad N \leq \tilde{k} \tag{C10}$$

Algorithm 3: Algorithm: for $n \leq N \leq \tilde{k}$ (first phase)

C.2. *Algorithm: for $n \leq \tilde{k} < N$ (second phase)*

According to Domínguez *et al.* (2011)s' algorithm, if $n \leq \tilde{k} < N$, we must modify Algorithm 3. In this case, we introduce the integers $n_0 = \lceil \tilde{k} \rceil$, the ceiling function mapping $\tilde{k}$ to the least integer greater than or equal to $\tilde{k}$, and $M \geq n_0$, the tridiagonal matrix and the right-hand side vector

$$A_M(\tilde{k})\boldsymbol{\rho}_M(\tilde{k}) = \mathbf{b}_M(\tilde{k}), \tag{C11}$$

where

$$A_M(\tilde{k}) = \begin{bmatrix} \frac{2n_0}{i\tilde{k}} & 1 & & & \\ -1 & \frac{2(n_0+1)}{i\tilde{k}} & 1 & & \\ & -1 & \frac{2(n_0+2)}{i\tilde{k}} & 1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & \frac{2(2M-1)}{i\tilde{k}} \end{bmatrix}, \mathbf{b}_M(\tilde{k}) := \begin{bmatrix} 2\gamma_{n_0}(\tilde{k}) + \rho_{n_0-1}(\tilde{k}) \\ 2\gamma_{n_0+1}(\tilde{k}) \\ 2\gamma_{n_0+2}(\tilde{k}) \\ \vdots \\ 2\gamma_{2M-1}(\tilde{k}) + \rho_{2M}(\tilde{k}) \end{bmatrix} \tag{C12}$$

$$\boldsymbol{\rho}_M(\tilde{k}) := \begin{bmatrix} \rho_{n_0}(\tilde{k}) & \rho_{n_0+1}(\tilde{k}) & \rho_{n_0+2}(\tilde{k}) & \cdots & \rho_{2M-1}(\tilde{k}) \end{bmatrix}^{\mathrm{T}}. \tag{C13}$$

Since $A_M(\tilde{k})$ is a tridiagonal matrix, we can use Oliver's algorithm (Oliver 1967), proposed by Domínguez *et al.* (2011), to solve (C11) to obtain $\boldsymbol{\rho}_M(\tilde{k})$. The coefficients $\gamma_n(\tilde{k})$ and $\rho_{n_0-1}(\tilde{k})$ can be obtained by Algorithm 3. The value of $\rho_{2M}(\tilde{k})$ is a priori unknown, but if we take $2M$ as sufficiently large, we can approximate it accurately by using an asymptotic expansion as shown in the next algorithm.

C.3. *Algorithm: for $\tilde{k} < n < N$ (thrid phase)*

According to Domínguez *et al.* (2011, Theorem 3.1), if M is sufficiently large, then we can compute the asymptotic expansion of $\rho_{2M}(\tilde{k})$ with the formula

$$2i\left(\sum_{r=0}^{J}(-1)^r p_{2r}(0)\sin\tilde{k} + \sum_{r=0}^{J}(-1)^r p_{2r+1}(0)\cos\tilde{k}\right) + R_J(M,k), \tag{C14}$$

where the coefficients are defined as

$$p_0(\theta) := \frac{1}{(2M - \tilde{k}\sin\theta)}, \quad p_r(\theta) := p_0(\theta)\frac{\mathrm{d}}{\mathrm{d}\theta}p_{r-1}(\theta), \quad r = 1, 2, \ldots, \tag{C15}$$

and $|R_J(M, K)| \leq C_J \tilde{k} M^{-2J-4}$, and C_J is independent of M and $\tilde{k}$. If $\theta = 0$, the first four coefficients can be formulated as follows:

$$p_0(0) := \frac{1}{2M}, \quad p_1(0) := \frac{\tilde{k}}{(2M)^3}, \quad p_2(0) := \frac{3\tilde{k}^2}{(2M)^5}, \quad p_2(0) := \frac{(15\tilde{k}^2 - 4M^2)\tilde{k}}{(2M)^7}. \tag{C16}$$

We summarise the ideas above in Algorithm 4.

1: Set $n_0 = \lceil \tilde{k} \rceil$;
2: Take $M \geq \max(n_0/2, N/2)$ as sufficiently large and compute $\rho_{2M}(\tilde{k})$ using (C14);
3: Construct $A_M(\tilde{k})$, $b_M(\tilde{k})$ as in and solve a linear system of equations:

$$A_M(\tilde{k})\boldsymbol{\rho}_M(\tilde{k}) = \mathbf{b}_M(\tilde{k})$$

to obtain a vector of $\boldsymbol{\rho}_M(\tilde{k})$;
4: Set $w_n(\tilde{k}) := \gamma_n(\tilde{k}) - \frac{n}{i\tilde{k}}\rho_n(\tilde{k}), \quad n = n_0, \ldots, N.$

Algorithm 4: Algorithm: for $\tilde{k} < n < N$ (second phase)

Remark 3 Based on all the algorithms proposed by Domínguez *et al.* (2011), the FCC rule applied to solve (C1) only requires $\mathcal{O}(N \log N)$ operations.

Appendix D: Table of cumulants

In Table D1, we show the first c_1, second c_2, and fourth c_4 cumulants of the GB model, NIG model, VG model and CGMY model. In the CGMY model, we only present the cumulants when $Y \in (0, 2)/\{1\}$ because when $Y = 1$, it becomes the VG model. Given the characteristic functions, the cumulants can be generally computed by using

$$c_k = \frac{1}{i^k}\frac{\partial^k \log\varphi(z)}{\partial z^n}\bigg|_{z=0}.$$

Acknowledgement

We thank Professor Bengt Fornberg, Department of Applied Mathematics, University of Colorado, for teaching the singular Fourier–Padé method and Professor Victor Domínguez, Department of Mathematics, University of Navarra, for help and advice on using the Filon–Clenshaw–Curtis rules.

References

Abramowitz, M. and Stegun, I.A., *Handbook of Mathematical Formulas, Graphs, and Mathematical Tables*, 1965 (Dover Publications, Inc.: New York, NY, USA).

Almendral, A., Numerical valuation of American options under the CGMY process. Available at `https://www.nr.no/directdownload/4289/Almendral_Vazquez_-_Numerical_Valuation_of_American_Options_Under_the.pdf`, 2005.

Almendral, A. and Oosterlee, C.W., On American Options Under the Variance Gamma Process. *Applied Mathematical Finance*, 2007, **14**, 131–152.

Andricopoulos, A.D., Widdicks, M., Duck, P.W. and Newton, D.P., Universal Option Valuation Using Quadrature Methods. *J. Financ. Econ.*, 2003, **67**, 447–471.

Applebaum, D., *Lévy Processes and Stochastic Calculus*, 2004 (Cambridge University Press: Cambridge).

Barndorff-Nielsen, O.E., Normal inverse Gaussian distributions and stochastic volatility modelling. *Scandinavian Journal of Statistics*, 1991, **24**, 1–13.

Broadie, M. and Yamamoto, Y., Application of the fast Gauss transform to option pricing. *Management Science*, 2003, **49**, 1071–1088.

Brummelhuis, R. and Chan, R.T.L., A Radial Basis Function Scheme for Option Pricing in Exponential Lévy Models. *Applied Mathematical Finance*, 2014, **21**, 238–269.

Carr, P., Geman, H., Madan, D.B. and Yor, M., The Fine Structure of Asset Returns: An Empirical Investigation. *Journal of Business*, 2002, **75**, 305–332.

Chan, R.T.L., Adaptive Radial Basis Function Methods for Pricing Options Under Jump-Diffusion Models. *Computational Economics*, 2016, **47**, 623–643.

Chan, R.T.L., Hedging and pricing early-exercise options with complex Fourier series expansion. *The North American Journal of Economics and Finance*, 2019 Available at `https://www.sciencedirect.com/science/article/abs/pii/S1062940818304194`.

Chan, T.L.R., Singular Fourier–Padé series expansion of European option prices. *Quantitative Finance*, 2018, **18**, 1149–1171.

Chan, T.L.R. and Hale, N., Hedging and Pricing European-type, Early-Exercise and Discrete Barrier Options using an Algorithm for the Convolution of Legendre Series. Available at `https://arxiv.org/abs/1811.09257`, 2019.

Chan, T.L.R. and Hubbert, S., Options pricing under the one-dimensional jump-diffusion model using the radial basis function interpolation scheme. *Review of Derivatives Research*, 2014, **17**, 161–189.

Chang, C.C., Chung, S.L. and Stapleton, R.C., Richardson extrapolation techniques for the pricing of American-style options. *Journal of Futures Markets*, 2007, **27**, 791–817.

Cont, R. and Tankov, P., *Financial Modelling With Jump Processes*, Chapman & Hall/CRC Financial Mathematics Series, 2004 (Chapman & Hall/CRC: Boca Raton, Fla., London).

Domínguez, V., Graham, I.G. and Smyshlyaev, V.P., Stability and error estimates for Filon–Clenshaw–Curtis rules for highly oscillatory integrals. *IMA Journal of Numerical Analysis*, 2011, **31**, 1253–1280.

Driscoll, T.A. and Fornberg, B., A Padé-based Algorithm for Overcoming the Gibbs phenomenon. *Numerical Algorithms*, 2001, **26**, 77–92.

Driscoll, T.A. and Fornberg, B., *The Gibbs Phenomenon in Various Representations and Applications*, 2011 (Sampling Publishing: Potsdam, NY).

Fang, F. and Oosterlee, C.W., A Novel Pricing Method For European Options Based On Fourier-Cosine Series Expansions. *SIAM Journal on Scientific Computing*, 2009a, **31**, 826–848.

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

Feng, L. and Linetsky, V., Pricing discretely monitored barrier options and defaultable bonds in Lévy process models: A fast Hilbert transform approach. *Mathematical Finance*, 2008, **18**, 337–384.

Fu, W. and Hirsa, A., A fast method for pricing American options under the Variance Gamma model. Available at `https://arxiv.org/abs/1903.07519`, 2019.

Geske, R. and Johnson, H.E., The American put option valued analytically. *Journal of Finance*, 1984, **39**, 1511–1524.

Glasserman, P., *Monte Carlo Methods in Financial Engineering*, Vol. 53, , 2003, Springer Science & Business Media.

Guo, X. and Li, Y., Valuation of American options under the CGMY model. *Quantitative Finance*, 2016, **16**, 1529–1539.

Hirsa, A., *Computational Methods in Finance*, Chapman & Hall/CRC Financial Mathematics Series, 2012 (Chapman & Hall/CRC: Boca Raton, Fla., London).

Hirsa, A., Karatas, T. and Oskoui, A., Supervised Deep Neural Networks (DNNs) for Pricing/Calibration of Vanilla/Exotic Options Under Various Different Processes. Available at `https://arxiv.org/abs/1902.05810`, 2019.

Hirsa, A. and Madan, D.B., Pricing American Options Under Variance Gamma. *Journal of Computational Finance*, 2004, **7**, 63–80.

Longstaff, F.A. and Schwartz, E.S., Valuing American Options by Simulation: A Simple Least-Squares Approach. *The Review of Financial Studies*, 2001, **14**, 113–147.

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

Madan, D.B., Carr, P. and Chang, E.C., The Variance Gamma Process and Option Pricing. *European Finance Review*, 1998, **2**, 79–105.

Madan, D.B. and Milne, F., Option Pricing with V. G. Martingale Components. *Mathematical Finance*, 1991, **1**, 39–55.

Maree, S.C., Numerical pricing of Bermudan options using Shannon wavelet expansions. Master's thesis, Delft Institute of Applied Mathematics, Delft University of Technology, Delft, The Netherlands, 2015.

Maree, S.C., Ortiz-Gracia, L. and Oosterlee, C.W., Pricing early-exercise and discrete barrier options by Shannon wavelet expansions. *Numerische Mathematik*, 2017, **136**, 1035–1070.

Mason, J.C. and Handscomb, D., *Chebyshev Polynomials*, 2002 (CRC Press: Florida, FL, USA).

Matache, A.M., Nitsche, P.A. and Schwab, C., Wavelet Galerkin pricing of American options on Lévy driven assets. *Quantitative Finance*, 2005, **5**, 403–424.

Oliver, J., Relative error propagation in the recursive solution of linear recurrence relations. *Numerische Mathematik*, 1967, **9**, 323–340.

Ortiz-Gracia, L. and Oosterlee, C.W., Robust Pricing Of European Options With Wavelets And The Characteristic Function. *SIAM Journal on Scientific Computing*, 2013, **35**, B1055–B1084.

Ortiz-Gracia, L. and Oosterlee, C.W., A Highly Efficient SHannon Wavelet Inverse FOurier Technique For Pricing EUropean Options. *SIAM Journal on Scientific Computing*, 2016, **38**, B118–B143.

O'Sullivan, C., Path dependent option pricing under Lévy processes. *EFA 2005 Moscow Meetings Paper*, 2005 Available at `http://ssrn.com/abstract=673424`.

Rambeerich, N., Tangman, D., Gopaul, A. and Bhuruth, M., Exponential time integration for fast finite element solutions of some financial engineering problems. *Journal of Computational and Applied Mathematics*, 2009, **224**, 668 – 678.

Ruijter, M., Versteegh, M. and Oosterlee, C., On The Application Of Spectral Filters In A FOurier Option Pricing Technique. *Journal of Computational Finance*, 2015, **19**, 75–106.

Sato, K.I., *Lévy Processes and Infinitely Divisible Distributions*, 1999 (Cambridge University Press: Cambridge, U.K., New York).

Trefethen, L.N., Driscoll, T.A. and Hale, N., *Chebfun Guide*, 2014 (Pafnuty Publications: Oxford), See `http://www.chebfun.org/`.

von Sydow, L., Höök, L.J., Larsson, E., Lindström, E., Milovanović, S., Persson, J., Shcherbakov, V., Shpolyanskiy, Y., Sirén, S., Toivanen, J., Waldén, J., Wiktorsson, M., Levesley, J., Li, J., Oosterlee, C.W., Ruijter, M.J., Toropov, A. and Zhao, Y., BENCHOP–The BENCHmarking project in option pricing. *International Journal of Computer Mathematics*, 2015, **92**, 2361–2379.

Wang, I.R., Wan, J.W.L. and Forsyth, P.A., Robust numerical valuation of European and American options under the CGMY process. *Journal of Computational Finance*, 2007, **10**, 31–69.

Zeng, P. and Kwok, Y.K., Pricing barrier and Bermudan style options under time-changed Lévy processes: Fast Hilbert transform approach. *SIAM Journal on Scientific Computing*, 2014, **36**, B450–B485.

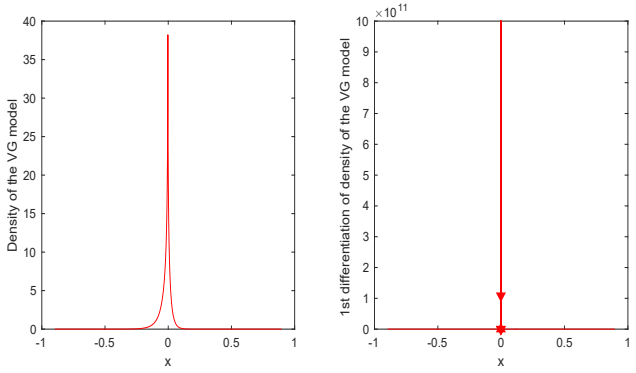

Figure B1. Density functions (left) of the VG model and its first derivative (right). The parameters are taken from **VG1**.

Table D1. The first c_1, second c_2, and fourth c_4 cumulants of various models.

Lévy models	
BS	$c_1 = (r - q + \omega)t \quad c_2 = \sigma^2 t, \quad c_4 = 0, \omega = -0.5\sigma^2$
NIG	$c_1 = (r - q + \omega)t + \delta t\beta/\sqrt{\alpha^2 - \beta^2}$ $c_2 = \delta t\alpha^2(\alpha^2 - \beta^2)^{-3/2}$ $c_4 = \delta t\alpha^2(\alpha^2 + 4\beta^2)^{-3/2}(\alpha^2 - \beta^2)^{-7/2}$ $\omega = -0.5\sigma^2 - \delta(\sqrt{\alpha^2 - \beta^2} - \sqrt{\alpha^2 - (\beta + 1)^2})$
VG	$c_1 = (r - q + \theta + \omega)t$ $c_2 = (\sigma^2 + \upsilon\theta^2)t$ $c_4 = 3(\sigma^4\upsilon + 2\theta^4\upsilon^3 + 4\sigma^2\theta^2\upsilon^2)t$ $\omega = 1/\upsilon \log(1 - \theta\upsilon - \sigma^2\upsilon/2)$
CGMY	$c_1 = (r - q + \omega)t$ $c_2 = (C\Gamma(2 - Y)(M^{Y-2} + G^{Y-2})t$ $c_4 = (C\Gamma(4 - Y)(M^{Y-4} + G^{Y-4})t$ $\omega = \left(C\Gamma(-Y)G^Y\left(\left(1 + \frac{1}{G}\right)^Y - 1 - \frac{Y}{G}\right) + C\Gamma(-Y)M^Y\left(\left(1 - \frac{1}{M}\right)^Y - 1 + \frac{Y}{M}\right)\right)$