TOPICS ON OPTION VALUATION AND MODEL CALIBRATION

BY

ZISHENG CHEN

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Industrial Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2013

Urbana, Illinois

Doctoral Committee:

Associate Professor Liming Feng, Chair
Associate Professor Ramavarapu S. Sreenivas
Professor Renming Song
Assistant Professor Enlu Zhou

Abstract

This dissertation is devoted to high performance numerical methods for option valuation and model calibration in Lévy process and stochastic volatility models. In the first part, a numerical scheme for simulating from an analytic characteristic function is developed. Theoretically, error bounds for bias are explicitly given. Practically, different types of options in commonly used Lévy process models could be priced through this method fast and accurately. Also, sensitivity analysis could be conducted through this approach effectively. Numerical results show that the schemes are effective for both options valuation and sensitivity analysis in Lévy process models. In the second part, a numerical scheme for Asian option pricing in jump-diffusion models is analyzed. Approximation errors are shown to decay exponentially. Numerical results show the speed and accuracy of the scheme. In the third part, for calibration purpose, certain numerical schemes are studied to price European and American options. For European options, error bounds are explicitly given. For American contracts, multiple options with different strikes and maturities could be priced simultaneously. Numerical results show that the combination of the above schemes with state-of-the-art optimization schemes makes efficient calibration of option pricing models possible.

To the son of man, for where transgression abounded, grace abounded much more.

Acknowledgments

I would like to express profound gratitude to my advisor Dr. Liming Feng, a respectable mentor, caring elder brother and beneficial friend. Without his detailed instruction and endless support, I could not have completed my doctoral study. He has not only provided me with academic knowledge, but also shown me his noble personal characters. Leading by example, he has taught me the value of diligence, self-motivation and self-sufficiency. I enjoyed greatly the inspiring discussions with him and have always rejoiced in being his student.

I would like to thank my doctoral committee members, professor Ramavarapu S. Sreenivas, Renming Song and Enlu Zhou for their precious time, effort and suggestions. With their selfless help, my last year in University of Illinois working on my dissertation was truly a wonderful journey.

I am very fortunate to be a member of Industrial and Enterprise Systems Engineering department family. I am delighted to know dedicated faculties, such as Jiming Peng, Uday Shanbhag, Enlu Zhou, Xin Chen, Jong-shi Pang and many others. Their professional knowledge, brilliant intelligence as well as enthusiasm for their research have been a great encouragement for me. For my six years in IsE, I was also blessed by professional staff including Holly Michelle Kizer, Donna Eiskamp, Randall Elkins, Amy Summers and many others.

I am very thankful for having excellent peer students around, growing and learning together, such as Ao Chen, Fan Ye, Fuyuan Wang, Hao Jiang, Jing Li, Jingnan Chen, Limeng Pan, Ping Liu, Peng Hu, Rui Yang, Shuoyuan He, Tao Zhu, Xiao Li, Ying Xiao, Yiqun Wang, Yu-Ching Lee, Yuhan Zhang, Zhi Yin and many others. Especially, I would like to thank Xiong Lin and Qiang Zeng, who have provided me with many good insights in research. Also, I am grateful to know brilliant friends in Quant group such as Rui Yang, Yufei Liu, Quan Geng and Chi Wang.

With great appreciation, I thank my church friends for their years of prayers and love, Xiaokang and Ping, Xiaofeng and Sarah, Yejun and Jinzhi, Jiajia and Jane, Han and Lixuan, as well as Kaiyan, Hui Liu, Mingyuan, Daniel, Hui Wen, Xiaolei, Yuan, Joseph, Lian, Qiong, Sherry and many others. Especially, I want to give thanks to Elder Wei-Laung Hu, who treats me as his spiritual son. And I

want to thank Xiaoyun, my girlfriend, for her faithful support.

In the end, I am greatly indebted to my parents, for their endless love and uncountable sacrifice. I know no matter rain or shine, they will be there.

Table of Contents

List of Abbreviations

$\mathbb{R}$: set of real numbers;

$\mathbb{C}$: set of complex numbers;

$\Re$: real part of complex variable;

$\Im$: imaginary part of complex variable;

$p.v.$: principle value;

T: option maturity;

K: option strike price;

S_t: underlying asset price at time t;

r: risk free interest rate;

q: dividend of underlying asset;

Chapter 1

Introduction

Derivative securities have become standard tools in today's financial world. According to a recent report by Bank of International Settlement, total notional amounts outstanding of over-the-counter (OTC) derivatives reached $638.9 trillion at the end of June 2012. Accurate valuation and effective risk management of derivative securities are thus essential for maintaining the stability of the global financial industry. Options are among the most actively traded derivative securities.

The first breakthrough of option pricing was the publication of the works by Fischer Black, Myron Scholes and Robert C. Merton in 1970s. Their classic Black-Scholes-Merton (BSM) model [11; 56] has become popular since then. However, although widely used in the financial industry, the BSM model is known to have severe limitations. The main drawbacks of the BSM model include failing to capture fat-tail behaviors of asset returns, omitting the possibility of jumps in asset prices, as well as assuming constant volatilities. Many alternative models have been proposed. The most popular ones among them include: affine jump-diffusion models proposed by Duffie, Pan and Singleton in [30], Heston's model [47], Merton's jump diffusion model [57], Kou's double exponential jump diffusion model [49], Normal inverse Gaussian (NIG) process model proposed by Barndorff Nielsen [7] and the CGMY model proposed by Carr, Geman, Madan and Yor [14]. The first two belong to the class of stochastic volatility models, and the rest belong to Lévy process models, which includes the BSM model as a special case.

Lévy process models and stochastic volatility models became popular because they allow jumps and stochastic volatility in asset prices, and are hence able to capture the fat tail structure. Moreover, empirical studies show that these models agree with market data well. Moreover, these models often admit explicit characteristic functions. This makes such models computationally tractable.

Some of the most important tasks that are associated with options include: option pricing, sensitivity estimation and model calibration. When we switch to the above more realistic models, closed-form solutions are not available any more, and efficient numerical methods are needed. Commonly used numerical methods include transform methods, numerical solution of partial integro-differential

equations and variational inequalities and Monte Carlo simulation. Among these methods, Monte Carlo simulation is very attractive because it handles multi-dimensional problems better. Also, it is easier to adjust for different payoff structures of different contracts. Transform methods are also often applicable when the characteristic functions of the stochastic processes underlying the alternative models are known explicitly. This is the case for many affine stochastic volatility models and for most of Lévy process models. In particular, transform methods become highly efficient when the analyticity of the characteristic functions are explored. However, transform methods are usually only applicable for low-dimensional problems. It is thus interesting to investigate whether Monte Carlo simulation can be combined with transform methods to handle more difficult problems. Moreover, it is interesting to study how Monte Carlo simulation can be conducted for models where only characteristic functions are known.

In this dissertation, we study transform methods as well as Monte Carlo simulation from distributions where only characteristic functions are given. The dissertation is divided into three main parts.

In the first part, we study Monte Carlo simulation for options pricing and sensitivity estimation in Lévy process models. When the transition density function or the cumulative distribution function (CDF) of the Lévy process is known explicitly, simulation could be conducted through traditional methods easily. Otherwise, simulating a Lévy process is not trivial. Generally, there could be infinitely many jumps in an arbitrary finite time horizon. One way to solve this issue is to either discard small jumps or replace small jumps with a Brownian motion and hence approximate the Lévy process by a jump diffusion process [5; 22]. However, in these methods, it is difficult to determine the threshold for small jumps for a given level of estimation bias. Another method is to simulate from infinite series representations of a Lévy process [12; 63]. Also, if a Lévy process has certain subordination structure, we only need to simulate the subordinator and a Brownian motion [66]. However, these approaches require the process to have certain structures, and hence are not applicable to general Lévy processes.

In our applications, due to the structures of the financial contracts, we are only interested in the values of a Lévy process on an evenly spaced time grid. Because a Lévy process admits independent and stationary increments, we only need to simulate from a Lévy increment. However, the probability density and CDF of the Lévy increment are usually not known explicitly. On the other hand, due to the Lévy -Khintchine formula for infinitely divisible distributions, the characteristic function of a Lévy process is often known explicitly. We thus need to simulate from the characteristic function

of the Lévy increment. One way to simulate from a characteristic function is shown in [64] where a connection between a quantile function and a characteristic function is established. The quantile then solves a nonlinear integro-differential equation. Another method is proposed by Devroye in [29], where a bound for the density function is obtained from a characteristic function, and then the acceptance-rejection method is applied. The idea of tabulating the CDF was also discussed in [29]. However, this was not pursued further. Glasserman and Liu [44; 45] study the inverse transform method with tabulated CDF for simulating and sensitivity estimation from a characteristic function. They use the inverse Laplace transform to calculate the CDF from a characteristic function [1]. They also demonstrate that the interpolation error incurred by using the tabulated CDF is of $O(\eta^2)$, where η is the fineness of the table. Also, the bias introduced by the numerical Laplace inversion is of $O(e^{-c/h})$, where c is a positive constant and h is the discretizaiton step size used for discretizing the inverse Laplace transform. In order to get the desired accuracy, they begin with initial guess of the parameters η and h. Then, they adjust the parameters in a way such that the above two types of errors decrease proportionally. Convergence is achieved by running the simulation several times.

In this dissertation, we use the Hilbert transform to calculate the CDF. Due to a powerful tool based on the sinc expansion of analytic functions [38], when the characteristic function belongs to a certain analytic class, the Hilbert transform method becomes remarkably accurate. When the trapezoidal rule with step size h is used, the discretization error is explicitly given and decays exponentially in $1/h$. Moreover, when the characteristic function has certain tail behavior, the truncation error is also explicitly given and decays very fast. Based on this, we are able to derive an explicit bound for the estimation bias when the inverse transform method based on tabulated CDF is used to simulate from a given characteristic function. For any given bias tolerance level, we can choose the numerical parameters in a way such that the total bias is less than the given tolerance level. In particular, for one dimensional problems we only need to run the simulation once.

Advantages of our method are also reflected in the following aspects: the inverse transform method does not require specific structures of the Lévy process; variance reduction techniques and quasi Monte Carlo schemes could be easily incorporated. We also compare our method with alternative schemes for processes with specific structures to test the numerical performance. For Kou's double exponential jump diffusion model [49], we compare our method with the direct implementation, which simulates the jump diffusion process by simulating a Brownian motion, a Poisson process and double exponential jump sizes separately. For the Normal inverse Gaussian (NIG) process model [7], we compare with the methods in [66] and [62], where the Brownian subordination structure of

the NIG process and inverse Gaussian bridge are used. The method in [66] is implemented, which simulates Brownian motion and inverse Gaussian subordinator; when multi dimensional problem is considered, the method in [66] is implemented together with inverse Gaussian bridge [62]. Numerical results show that, even for processes with special structures, the inverse transform method remains competitive.

This method could also be easily extended to perform sensitivity analysis. Sensitivity measures the relative rate of change between two variables. It plays a significant role in financial risk management. For example, to conduct delta hedging, one must compute the rate of change in the derivative price when the underlying asset price changes. Monte Carlos simulation is a very useful tool to estimate sensitivities [18; 43; 45]. We consider the likelihood ratio method (LRM) in particular. Again, the sinc based methods are applicable and allow us to derive explicit bounds for the estimation bias.

Discrete Asian options are priced using Monte Carlo simulation in the first part. To obtain accurate benchmark prices, in the second part, we study a transform based scheme for pricing Asian options. Asian options are actively traded in the natural resource derivatives market, especially on crude oil and heating oil. Asian options can be categorized by sampling frequency into continuously monitored and discretely monitored ones, or by how the average is calculated, into arithmetic Asian and geometric Asian options. We focus on discretely monitored arithmetic Asian options. Continuously monitored Asian option prices can be approximated by increasing the monitoring frequency. Geometric Asian options can usually be priced easily and used as control variates. Various numerical methods have been proposed for pricing Asian options. Methods based on numerical solution of PIDEs are studied in [3; 69]. One problem with these methods is that, the discrete sampling structure will impair the convergence rate in time horizon. Another type of methods relies on the Fourier transform and the so called Carverhill-Clewlow or Steward-Hodges factorization, which was first applied in pricing Asian options by Carverhill and Clewlow in [17]. Since then, Benhamou [8] proposed a re-centering scheme to alleviate the density shifting issue in backward induction, and hence reduced the computational cost. Fusai and Meucci [42] further improved their method using Gaussian quadrature and provided bounds for errors. Cerny and Kyriakou [68] reversed the backward induction and analyzed the approximation error.

In the above Fourier transform based approaches, only polynomial convergence was achieved. In this dissertation, we study a Fourier transform based approach for pricing discrete Asian options in jump-diffusion models. In particular, exponential convergence is established theoretically and illustrated numerically. In the theoretical part, we revisit results on the relationship between the

analyticity of a function and the tail behavior of its Fourier transform [34; 54; 61]. Then, we take advantage of the powerful approximation theory for analytic functions [38; 65]. Numerical results show that Asian options could be priced with very high accuracy. We are then able to use these as benchmark prices when Monte Carlo simulation is used to price Asian options.

So far, we have assumed that the parameters of the option pricing models are given. In the third part, we calibrate Lévy process models from both European and American options. The calibration problem could be divided into two sub-problems: the forward problem, which is the option pricing problem; and the inverse problem, which is an optimization problem. The calibration problem is usually numerically very demanding, since the forward problem itself can only be solved numerically. In the forward problem, a large number of options should be priced repeatedly for the purpose of model calibration. Unless the pricing scheme is sufficiently fast and is able to price multiple options simultaneously, the computational cost is usually too large for practical use. In this regard, Dupire [31] type structures are particularly useful, where options with different strikes and maturities could be priced simultaneously. Dupire type structures were initially studied for European options in local volatility model [27; 28]. See also [25] and [50]. In this dissertation, we utilize a Dupire type structure for American options in Lévy process models.

The inverse problem is also computationally challenging. When the least square formulation is used, the objective function is nonlinear and non-convex. We consider global optimization methods as well as regularization techniques. Popular regularization methods include Tikhonov regularization and relative entropy regularization. The former one was used in calibrating local volatility models in [26] and [50]; the latter was applied in non-parametric calibration of jump diffusion models in [22], [23] and [24]. For more works on regularization techniques for model calibration problems, see [32; 35; 46; 48].

In this dissertation, for the forward problem, we study a sinc based scheme for pricing European options [65], which is fast and accurate. For American options, we revise the algorithm by Feng and Lin in [39] for pricing Bermudan options. More specifically, we obtain multiple option prices with different strike prices and maturities by one run of backward induction. For the inverse problem, we use a two-step procedure, combining the global search for the original problem and the local search for the problem with Tikhonov regularization. Finally, we carry out empirical studies to exhibit the effectiveness of the calibration scheme.

Chapter 2

Analytic Characteristic Function

In this part, we first introduce an analytic class. For functions in the analytic class, very powerful numerical approximation result could be applied to integration type of calculation for those functions. Corresponding theory is shown in detail in [65]. When the function is characteristic function, we could take advantage of the result in obtaining either cumulative distribution function (CDF) [38] or probability density function (pdf) fast and accurately. Moreover, analytic characteristic function also implies some interesting tail behavior of corresponding CDF, as shown in [54]. Furthermore, in [34] and [61], they talk about Fourier transform theory and provide some relative result on relationship between tail behavior of a function and the analyticity of its (inverse) Fourier transform. Besides quote and moderate revise of the known results, new findings are also added in this part to set up the theoretical framework serving later application purpose.

2.1 An Analytic Class Of Functions

For $d_\pm$ satisfying $-\infty < d_- < 0 < d_+ < +\infty$, we consider the horizontal strip in the complex plane $\mathcal{D}_{(d_-,d_+)} = \{z \in \mathbb{C} : \Im(z) \in (d_-, d_+)\}$, where $\Im(z)$ is the imaginary part of complex variable z.

Definition 2.1.1. *ψ is in $H(\mathcal{D}_{(d_-,d_+)})$ if it is analytic in the horizontal strip $\mathcal{D}_{(d_-,d_+)}$, absolutely integrable on $\mathbb{R}$, and satisfies:*

$$\int_{d_-}^{d_+} |\psi(\xi + iy)|dy \to 0, \quad \xi \to \pm\infty, \tag{2.1a}$$

$$||\psi||^{\pm} := \lim_{\epsilon \to 0+} \int_{\mathbb{R}} |\psi(\xi + i(d_\pm \mp \epsilon))|d\xi < +\infty. \tag{2.1b}$$

For the integration and transformation of functions in this analytic class, there is a powerful approximation theory that could be applied. Here we quote three results from [38]. Readers may refer to [38] for more details.

Lemma 2.1.2. *(Trapezoidal rule, Theorem 2.3 in [38]) Suppose $\psi \in H(\mathcal{D}_{(d_-,d_+)})$. Then*

$$|E_h^T(\psi, a=0)| := \Big| \int_{-\infty}^{\infty} \psi(x)dx - \sum_{l=-\infty}^{\infty} \psi(lh)h \Big| \\ \leq \frac{e^{-2\pi(-d_-)/h}}{1-e^{-2\pi(-d_-)/h}} \|\psi\|^- + \frac{e^{-2\pi d_+/h}}{1-e^{-2\pi d_+/h}} \|\psi\|^+.$$

Lemma 2.1.3. *(Fourier transform, Corollary 2.4 in [38]) Suppose $\psi \in H(\mathcal{D}_{(d_-,d_+)})$. Then*

$$|E_h^F(\psi, a=0)(x)| := \Big| \int_{\mathbb{R}} e^{ixy}\psi(y)dy - \sum_{l=-\infty}^{\infty} e^{ixlh}\psi(lh)h \Big| \\ \leq \frac{e^{-2\pi(-d_-)/h}}{1-e^{-2\pi(-d_-)/h}} e^{-xd_-} \|\psi\|^- + \frac{e^{-2\pi d_+/h}}{1-e^{-2\pi d_+/h}} e^{-xd_+} \|\psi\|^+.$$

Lemma 2.1.4. *(Hilbert transform, Corollary 2.9 in [38]) Suppose $\psi \in H(\mathcal{D}_{(d_-,d_+)})$. Then*

$$|E_h^H(\psi, a=0)(0)| = \Big| \frac{1}{\pi} p.v. \int_{\mathbb{R}} \frac{\psi(x)}{0-x}dx - \sum_{l=-\infty, l\neq 0}^{\infty} \psi(lh) \frac{1-(-1)^l}{-l\pi} \Big| \\ \leq \frac{e^{-\pi(-d_-)/h}}{\pi(-d_-)(1-e^{-\pi(-d_-)/h})} \|\psi\|^- + \frac{e^{-\pi d_+/h}}{\pi d_+(1-e^{-\pi d_+/h})} \|\psi\|^+.$$

As we can easily observe from the previous lemmas, the discretization error bounds are explicitly given and decay exponentially in terms of $1/h$, by using simple numerical schemes. These results would be used throughout the thesis. Also, when the tail behavior of the function is known, it is also possible to obtain error bounds for truncation errors. When combining these two types of error bounds, we are able to compute the integrals and transforms fast and accurately using simple numerical schemes with explicit approximation error estimates.

2.2 Analyticity and Tail Behavior

The analyticity of a function is connected to the tail behavior of its (inverse) Fourier transform. Related results have been obtained for a long time and shown in different forms in many publications. Here we summarize some of the results and make certain modifications of the original proofs to serve our purpose. We begin with showing a lemma, which is an interesting and useful result: the regularity of characteristic functions is equivalent to the finiteness of exponential moments of the corresponding distributions. Notice that, a characteristic function $\phi(\xi)$, $\xi \in \mathbb{R}$, is said to be analytic if $\phi(z)$, $z \in \mathbb{C}$, is analytic in some region, and $\phi(z)$ and $\phi(\xi)$ agree on the real line.

Lemma 2.2.1. *The characteristic function $\phi(z)$ of a random variable X is analytic in the strip $\{z \in \mathbb{C}, -\alpha < \Im(z) < \beta\}$, $\alpha, \beta > 0$, if and only if $\mathbb{E}[e^{-yX}]$ exists and is finite for any $-\alpha < y < \beta$.*

Proof. "$\Longrightarrow$": By analyticity, $\phi(z) = \int_{-\infty}^{\infty} e^{izx} dF(x)$ exists and is finite at any point in the strip $\{z \in \mathbb{C}, -\alpha < \Im(z) < \beta\}$. Let $z = iy$, where $-\alpha < y < \beta$. Then

$$\int_{-\infty}^{\infty} e^{izx} dF(x) = \int_{-\infty}^{\infty} e^{-yx} dF(x) = \mathbb{E}[e^{-yX}]$$

exists and is finite.

"$\Longleftarrow$": Suppose $\mathbb{E}[e^{-yX}]$ exists and is finite for any $-\alpha < y < \beta$. For any $z = t + iy, t \in \mathbb{R}, -\alpha < y < \beta$,

$$\Big| \int_{-\infty}^{\infty} e^{izx} dF(x) \Big| = \Big| \int_{-\infty}^{\infty} e^{i(t+iy)x} dF(x) \Big| \leq \int_{-\infty}^{\infty} e^{-yx} dF(x) = \mathbb{E}[e^{-yX}].$$

Hence, $\phi(z) = \int_{-\infty}^{\infty} e^{izx} dF(x)$ exists and is finite for $-\alpha < \Im(z) < \beta$. In order to show $\phi(z)$ is analytic in the strip, we only need to show it is differentiable. Note that

$$\lim_{z \to z_0} \frac{\phi(z) - \phi(z_0)}{z - z_0} = \lim_{z \to z_0} \int_{-\infty}^{\infty} \frac{e^{izx} - e^{iz_0 x}}{z - z_0} dF(x),$$

where z_0 is an arbitrary point in the analytic strip and z is in a small enough neighborhood of z_0. To change the order of limit and integration, we need to use the dominated convergence theorem. And here we need to take advantage of the complex mean-value theorem as shown in [36], which states that, suppose Ω is an open convex set in $\mathbb{C}$, then for holomophic function $f : \Omega \to \mathbb{C}$, suppose a and b are distinct points in Ω . Then there exist points z_1 and z_2 on the straight line connecting a and b not containing the endpoints, such that

$$\Re(\frac{f(b) - f(a)}{b - a}) = \Re(f'(z_1)), \quad \Im(\frac{f(b) - f(a)}{b - a}) = \Im(f'(z_2)).$$

In our case, $f(z) = e^{izx}$, and if we define $z = u + iv$, then

$$\begin{aligned} f'(z) = ixe^{izx} &= ixe^{i(u+iv)x} = ix[\cos(ux) + i\sin(ux)]e^{-vx} \\ &= -x\sin(ux)e^{-vx} + ix\cos(ux)e^{-vx}. \end{aligned}$$

And

$$\frac{e^{izx} - e^{iz_0 x}}{z - z_0} = \Re(\frac{e^{izx} - e^{iz_0 x}}{z - z_0}) + i\Im(\frac{e^{izx} - e^{iz_0 x}}{z - z_0}) = -x\sin(u_1 x)e^{-v_1 x} + ix\cos(u_2 x)e^{-v_2 x},$$

for some $z_1 = u_1 + iv_1$ and $z_2 = u_2 + iv_2$ in the line segment of z and z_0. Hence,

$$|\frac{\phi(z) - \phi(z_0)}{z - z_0}| \leq \int_{-\infty}^{\infty} |\frac{e^{izx} - e^{iz_0x}}{z - z_0}| dF(x) \leq \int_{-\infty}^{\infty} (|x|e^{-v_1x} + |x|e^{-v_2x}) dF(x) < \infty.$$

Now we explain why the last inequality holds. For any z_0 fixed, if we let $y_0 = \Im(z_0)$, since $-\alpha < v_1, v_2 < \beta$, we can choose a $\delta > 0$ small enough, for example, $\delta = \min(|\alpha - y_0|, |\beta - y_0|)/2$, such that when z is in the δ neighborhood of z_0, we know $v_1 - \delta$, $v_1 + \delta$, $v_2 - \delta$, $v_2 + \delta$ are still in $(-\alpha, \beta)$. Then we can use $e^{\pm\delta x}$ to control $|x|$ while the rest part is still be controlled by $F(x)$. To be more specific, consider $\int_0^{\infty} |x|e^{-v_1x} dF(x)$. We divide e^{-v_1x} by two parts: $e^{-\delta x}$ is used to control $|x|$, and $e^{-(v_1-\delta)x}$ is controlled by $F(x)$. Then, the integral $\int_0^{\infty} |x|e^{-v_1x} dF(x)$ is finite. We can deal with other three cases in the same manner. Then we know $\phi(z)$ is differentiable and hence represents an analytic function in the strip $-\alpha < \Im(z) < \beta$. □

Now we states the first theorem, which shows the relationship between the tail behavior of a CDF and the analyticity of the corresponding characteristic function. It is originally from Theorem 7.2.1 in [54]. Here we follow their proof and extend it to more general situation by allowing asymmetric analytic strip.

Theorem 2.2.2. *The characteristic function $\phi(\xi)$ of a distribution function $F(x)$ is an analytic characteristic function if and only if there exists two positive constants α and β such that*

$$(i) \quad 1 - F(x) = o(e^{-rx}) \quad as \quad x \to \infty$$

holds for all $0 < r < \alpha$, and

$$(ii) \quad F(-x) = o(e^{-rx}) \quad as \quad x \to \infty.$$

holds for all $0 < r < \beta$. The strip of regularity of $\phi(z)$ contains then the strip $-\alpha < \Im(z) < \beta$.

Remark 2.2.3. *The condition is equivalent to*

$$(i) \quad 1 - F(x) = O(e^{-(\alpha-\epsilon)x}) \quad as \quad x \to \infty;$$
$$(ii) \quad F(-x) = O(e^{-(\beta-\epsilon)x}) \quad as \quad x \to \infty.$$

for any positive ϵ.

Proof. We first prove the *sufficient* condition. Let y, r in $\mathbb{R}$ such that $-\alpha < -r < y < 0$, and let

$k \geq 1$ be a positive integer. By condition (i) we know $1 - F(k-1) = o(e^{-r(k-1)})$ as $k \to \infty$. Hence, there exists a constant C and K large enough, such that $1 - F(k-1) \leq Ce^{-rk}$ for any $k \geq K$. Then

$$\int_{k-1}^{k} e^{-yx} dF(x) \leq e^{-yk}[1 - F(k-1)] \leq Ce^{-k(r+y)} \quad \text{for} \quad k \geq K.$$

We choose a real number $x_0 \geq K$. Then for any $M > 0$,

$$\int_{x_0}^{x_0+M} e^{-yx} dF(x) \leq \sum_{k=K}^{\infty} \int_{k-1}^{k} e^{-yx} dF(x) \leq C \sum_{k=K}^{\infty} e^{-k(r+y)} = \frac{Ce^{-(r+y)K}}{1 - e^{-(r+y)}}.$$

The last term can be made arbitrarily small when K is sufficiently large. Therefore, the integral

$$\int_{0}^{\infty} e^{-yx} dF(x)$$

exists and is finite for $-\alpha < y < 0$. And it is obvious that the integral exists and is finite for all $y \geq 0$. To sum up, for all $y > -\alpha$, we have the integral exists and is finite.
Similarly, for any $0 < y < \beta$, we have:

$$\int_{-k}^{-k+1} e^{-yx} dF(x) \leq e^{yk} F(-k+1),$$

And for $y < r < \beta$, from (ii) we know that for large enough C' and K', when $k > K'$, we have $F(-k+1) \leq C'e^{-rk}$. Then for $x_0' > K'$ and $M' > 0$,

$$\int_{-x_0'-M'}^{-x_0'} e^{-yx} dF(x) \leq \frac{C'e^{-(r-y)K'}}{1 - e^{-(r-y)}}.$$

By the same argument, we know the integral

$$\int_{-\infty}^{0} e^{-yx} dF(x)$$

exists and is finite for $0 < y < \beta$. Also, it is obvious that the integral exists and is finite for any $y \leq 0$. Hence, it also exists and is finite for all y such that $y < \beta$.
Combining this with the earlier result, we know $\int_{-\infty}^{\infty} e^{-yx} dF(x)$ exists and is finite for any y such that $-\alpha < y < \beta$. Hence, by lemma 2.2.1, $\phi(z)$ represents a regular analytic function.

Then we prove the *necessary* condition. Suppose the characteristic function $\phi(z) = \int_{-\infty}^{\infty} e^{izx} dF(x)$

is a characteristic function analytic in the strip $-\alpha < \Im(z) < \beta$. Let $x > 0$, then by lemma 2.2.1 the two integrals

$$\int_x^\infty e^{-yu} dF(u) \quad \text{and} \quad \int_{-\infty}^{-x} e^{-yu} dF(u)$$

exist and are finite for all $-\alpha < y < \beta$. Moreover, the first integral also exists and is finite for all $y > 0$ and hence is finite for all $y > -\alpha$ (or $-y < \alpha$). The second integral also exists and is finite for all $y < 0$ and hence is finite for all $y < \beta$. If we choose $r, \delta > 0$, such that $0 < r < r + \delta < \alpha$. Then $\exists C > 0$ such that:

$$C > \int_x^\infty e^{(r+\delta)u} dF(u) \geq e^{(r+\delta)x}[1 - F(x)] \geq 0,$$

which implies $1 - F(x) \leq Ce^{-(r+\delta)x}$. Hence

$$0 \leq [1 - F(x)]e^{rx} \leq Ce^{-\delta x}.$$

The last expression goes to zero as $x \to \infty$, so that $1 - F(x) = o(e^{-rx})$ as $x \to \infty$. Hence, (i) holds. Similarly, we can prove (ii) also holds. Take $0 < r < r + \delta' < \beta$, then

$$C' > \int_{-\infty}^{-x} e^{-(r+\delta')u} dF(u) \geq e^{(r+\delta')x} F(-x) \geq 0$$

Hence

$$0 \leq F(-x)e^{rx} \leq C'e^{-\delta' x}.$$

so, $F(-x) = o(e^{-rx})$ as $x \to \infty$, for any $0 < r < \beta$.

□

Not only distribution functions but also probability density functions have close relationship with their characteristic functions. And in a more general setting, here we introduce the tail behavior and analyticity relationship between functions and their Fourier transforms. Following we quote partial results and proof of theorem 26 in [34] without modification.

Theorem 2.2.4. *Let $z = x + iy$, $f(z)$ be an analytic function, regular for $-a < y < b$, where $a > 0$, $b > 0$. If $f(z)$ satisfies the following*

$$f(z) = \begin{cases} O(e^{-(\lambda-\epsilon)x}) & as \quad x \to \infty; \\ O(e^{(\mu-\epsilon)x}) & as \quad x \to -\infty \end{cases}$$

for any $-a < y < b$ and every positive ϵ, λ and μ. Then $\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x)e^{ix\omega}dx$ satisfies conditions similar to those imposed on $f(z)$, with a, b, λ, μ replaced by λ, μ, b, a.

Proof. Let $\omega = u+iv$, then the integral $\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x)e^{ix\omega}dx$ converges uniformly for $-\lambda < v < \mu$. Hence $\hat{f}(\omega)$ is analytic in this strip. By Cauchy's theorem, the integral on real axis could be replaced by corresponding integral on any parallel line. Then:

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x+iy)e^{i(x+iy)(u+iv)}dx = O(e^{-yu}),$$

and then take y close to $-a$ or b, we know the tail behavior of $\hat{f}(\omega)$. □

Remark 2.2.5. *Notice that in order to show $\hat{f}(\omega)$ is analytic in a strip, exponentially decaying tail behavior of $f(z)$ on real axis would be sufficient. And the proof can be easily derived following same argument used in Lemma 2.2.1.*

The following theorem shows that the analyticity of functions could imply exponentially decaying tail behavior of their inverse Fourier transforms if some minor integrability conditions are posed.

Theorem 2.2.6. *Let $\omega = u + iv$, if (i) $\hat{f}(\omega)$ is analytic in $-\alpha < v < \beta$; (ii) $\hat{f}(\cdot + iv) \in L^1(\mathbb{R})$ in the analytic strip; (iii)* $\lim_{u\to\pm\infty} \int_{-\alpha}^{\beta} |\hat{f}(u+iv)|dv = 0$. *Then*

$$f(x) = \begin{cases} O(e^{-(\alpha-\epsilon)x}) & as \quad x \to \infty; \\ O(e^{(\beta-\epsilon)x}) & as \quad x \to -\infty \end{cases}$$

Proof. Since $\hat{f}(\omega)$ is analytic in $-\alpha < v < \beta$, by Cauchy's theorem and the fact that

$$\lim_{u\to\pm\infty} \int_{-\alpha}^{\beta} |\hat{f}(u+iv)|dv = 0$$

we have:

$$\begin{aligned} f(x) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(u)e^{-iux}du = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(u+iv)e^{-i(u+iv)x}du \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(u+iv)e^{-iux}e^{vx}du = O(e^{vx}) \end{aligned}$$

The last equality is true because $\|\hat{f}(\cdot + iv)\|_1 < \infty$. If we let $x \to \infty$ and take $v \to -\alpha$, then we have $f(x) = O(e^{-(\alpha-\epsilon)x})$; if we let $x \to -\infty$ and take $v \to \beta$, then we have $f(x) = O(e^{(\beta-\epsilon)x})$. □

Remark 2.2.7. *Since*

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(u+iv)e^{-iux}e^{vx}du = \frac{1}{2\pi}\Big(\int_{-\infty}^{\infty} \hat{f}(u+iv)e^{-iux}du\Big)\cdot e^{vx},$$

we know

$$|f(x)| \le \frac{1}{2\pi}\Big(\int_{-\infty}^{\infty} |\hat{f}\big(u+i(-\alpha+\epsilon)\big)|du\Big)\cdot e^{(-\alpha+\epsilon)x}$$

and

$$|f(x)| \le \frac{1}{2\pi}\Big(\int_{-\infty}^{\infty} |\hat{f}\big(u+i(\beta-\epsilon)\big)|du\Big)\cdot e^{(\beta-\epsilon)x}$$

are satisfied for $\forall \epsilon > 0$. *Take* $\epsilon \to 0$, *we have* $|f(x)| \le \frac{1}{2\pi}\|\hat{f}\|^{-}e^{-\alpha x}$ *and* $|f(x)| \le \frac{1}{2\pi}\|\hat{f}\|^{+}e^{\beta x}$.

Remark 2.2.8. *In above deductions, we only require* $\hat{f}(\cdot+iv) \in L^1(\mathbb{R})$ *for* v *close to* β *and* $-\alpha$. *In this case, the conditions posed on* $\hat{f}(\omega)$ *is essentially* $\hat{f}(\omega) \in H(\mathcal{D}_{(-\alpha,\beta)})$.

If we add stronger condition by further assuming $\hat{f}(\cdot+iv)$ has exponentially decaying tail behavior near its analytic boundary, then we can obtain more general result on tail behavior of $f(z)$, where $z = x + iy$ is a complex variable, as shown in the following proposition.

Proposition 2.2.9. *Let* $\omega = u + iv$, *then if (i)* $\hat{f}(\omega)$ *is analytic in* $-\alpha < v < \beta$; *(ii) in the analytic strip (close to its boundary), we have*

$$\hat{f}(u+iv) = \begin{cases} O(e^{-(a-\epsilon)u}) & as \quad u \to \infty; \\ O(e^{(b-\epsilon)u}) & as \quad u \to -\infty; \end{cases}$$

(iii) $\lim_{u\to\pm\infty}\int_{-\alpha}^{\beta}|\hat{f}(u+iv)|dv = 0$. *Then, for any* $z = x + iy$, *where* $-a < y < b$, $a, b > 0$, *we have:*

$$f(z) = \begin{cases} O(e^{-(\alpha-\epsilon)x}) & as \quad x \to \infty; \\ O(e^{(\beta-\epsilon)x}) & as \quad x \to -\infty. \end{cases}$$

Proof. Since $\hat{f}(\omega)$ is analytic in $-\alpha < v < \beta$, by Cauchy's theorem, we have:

$$\begin{aligned} f(z) &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\hat{f}(u)e^{-iuz}du = \frac{1}{2\pi}\int_{-\infty}^{\infty}\hat{f}(u+iv)e^{-i(u+iv)(x+iy)}du \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\hat{f}(u+iv)e^{uy}e^{-iux}e^{-ivy}e^{vx}du = O(e^{vx}) \end{aligned}$$

The last equation holds because the fact that $\hat{f}(u+iv)e^{uy}$ is absolutely integrable for any $-a < y < b$ (when v close to $-\alpha$ or β). If we let $x \to \infty$ when $v \to -\alpha$, then we have $f(z) = O(e^{-(\alpha-\epsilon)x})$; if we

let $x \to -\infty$ when $v \to \beta$, then we have $f(z) = O(e^{(\beta-\epsilon)x})$ are satisfied for any $-a < y < b$. $\square$

To summarize, the results we presented here are essentially about functions which are analytic in a strip in the complex plane. The analyticity condition of those functions is usually equivalent to the exponentially decaying tail behavior of their (inverse) Fourier transforms. If stronger condition for those functions is provided, for example, the original functions may have compact support, then the resulting Fourier transforms would be entire in the complex plane. The related theory is called Schwartz's Paley-Wiener theorem and is mentioned in [61].

Chapter 3

Simulation From Analytic Characteristic Functions

As well as its application in science and engineering, Monte Carlo (MC) simulation plays an important role in computational finance these years. Comparing with other popular methods like PDE or transform methods, MC simulation can be applied in solving high dimensional problems. Also, in its application on derivative pricing, only minor modifications is required to fit different type of contracts. In contrast, large amount of extra work is inevitable if other methods are used. On the other hand, the main drawback of MC simulation is its computational speed, which is due to its slow convergence rate. Monte Carlo simulation is known to converge in a rate of square root, which is very slow comparing with the second order convergence of PDE methods and at least polynomial or even exponential convergence rate of transform methods. However, there are certain techniques could be applied to accelerate the computation of MC simulation, for example, variance reduction techniques or quasi-Monte Carlo.

In this chapter, we will propose a method by which we could effectively simulate from analytic characteristic function. We will show that this method could be easily combined with various variance reduction techniques to reduce computational cost. Moreover, a theoretical framework will be developed for error estimation, by which explicit error bound for the bias can be derived. Then, we illustrate our method by pricing different type of options in Lévy process models, where comparison with other simulation methods is demonstrated.

3.1 The Inverse Transform Method

Options in derivative market can usually be categorized by vanilla type options like European option as well as exotic ones. Path-dependent options are even more difficult to price among their exotic peers. As the name shows, the path-dependent option prices depend not only on the underlying asset price at maturity, but also on the whole path of price trajectory. However, in real world market, it is not possible to generate and keep track of continuous price path, hence instead, only

prices at discrete monitoring dates are taking into account. In order to do option pricing, we are trying to estimate the following type of expectation:

$$\mathbb{E}[f(X_{t_1}, ..., X_{t_d})] \tag{3.1}$$

where $t_1, ..., t_d$ are d monitoring dates. And X_t is a stochastic process. Here, for illustration purpose, we assume it follows Lévy process. European option is then a trivial example where $d = 1$ and $t_d = T$ the option maturity.

Monte Carlo simulation takes advantage of the LLN (law of large number) by simulating N paths of the underlying asset price. Hence, N sample of $f(X^i_{t_1}, ..., X^i_{t_d})$ are generated. And the approximation to the expectation (3.1) is given by:

$$\mathbb{E}[f(X_{t_1}, ..., X_{t_d})] \approx \frac{1}{N} \sum_{i=1}^{N} f(X^i_{t_1}, ..., X^i_{t_d})$$

For Lévy process, since it has independent increments property, the path dependent problem could be easily reduced to simulate individual Lévy increments. To be more specific, since Lévy process X_t begins at $X_0 = X_{t_0} = 0$, if we define $\Delta X_{t_k} = X_{t_k} - X_{t_{k-1}}$, then, we only need to simulate Lévy increments ΔX_{t_k}, k is from 1 to d.

3.1.1 Simulating From Tabulated CDF

First, we introduce the inverse transform method for simulation, which is a classic method has long been used. Assume random variable X has distribution function $F(x)$. Then, in order to simulate from X, one only need to simulate standard uniform random variable $U_i \sim [0, 1]$. Then, $X_i = F^{-1}(U_i)$ follows $F(x)$. Here then the difficulty comes from the derivation of F^{-1}, which is often unavailable. We will then follow the same approach by Glasserman and Liu in [44; 45]. They utilized a traditional method by storing the CDF value in the following type of table:

$$\begin{pmatrix} x_0 & x_1 & \cdots & x_K \\ \hat{F}_0 & \hat{F}_1 & \cdots & \hat{F}_K \end{pmatrix}. \tag{3.2}$$

Here is the procedure of doing that: first, the domain of the CDF is truncated to the close interval $[x_0, x_K]$. Then, this finite domain is divided into K equally spaced subintervals $[x_k, x_{k+1}]$, $k = 0, 1, ...K - 1$. Each interval is $\eta = (x_K - x_0)/K$ in length. Then, the CDF value at those $K + 1$

points are estimated by numerical methods. And the approximated CDF value $\hat{F}_k$ is stored in the table coupled with the corresponding x_k value, for $k = 0, 1, ...K$.

Since F^{-1} and/or F is usually unknown, instead of applying inverse transform method directly on distribution F, we consider using an approximation distribution $\hat{F}$ instead. Here $\hat{F}$ is derived from the stored table (3.2):

$$\hat{F}(x) = \begin{cases} 0, & x < x_0 \\ \hat{F}_{k-1} + \frac{\hat{F}_k - \hat{F}_{k-1}}{\eta}(x - x_{k-1}), & x_{k-1} \leq x < x_k, 1 \leq k \leq K \\ 1, & x \geq x_K \end{cases} . \tag{3.3}$$

We first generate random sample from standard uniform random variable $U_i \sim [0, 1]$. Then we search in the increasing probability sequence $\hat{F}_k$, $k = 0, 1, ..., K$ to find a fitting for U_i such that $\hat{F}_k \leq U_i < \hat{F}_{k+1}$. Once the fitting interval is found, the random sample X_i could be generated by

$$X_i = x_k + \frac{x_{k+1} - x_k}{\hat{F}_{k+1} - \hat{F}_k}(U_i - \hat{F}_k). \tag{3.4}$$

For the special cases when $0 \leq U_i < \hat{F}_0$, we set $X_i = x_0$; when $\hat{F}_K \leq U_i < 1$, we set $X_i = x_K$. In this case, X_i follows distribution $\hat{F}$ instead of F. It is apparent that bias has been introduced during this process. And there are three types of error:

1. The truncation error introduced from truncating the domain of the CDF, which is related to x_0 and x_K;

2. The interpolation error introduced when we use linear interpolation scheme (3.4), which depends on η (or K), the subinterval length;

3. the approximation error introduced when using $\hat{F}_k$ to approximate $F(x_k)$, which depends on the numerical scheme used.

In the next section, we will introduce the Hilbert representation we used to approximate the probability values. And we will further discuss the bias estimation later.

Remark 3.1.1.

- Binary search could be used to find the fitting interval, and the computational complexity is approximately $\log_2(K)$.

- When we are trying to estimate $\mathbb{E}[f(X)]$ in applications, some of the function f would have the following property: $f(x) = 0$ for $x \leq z_0$ or $x \geq z_0$. In this case, we can safely take $x_0 = z_0$ or $x_K = z_0$. For example, in option pricing, for European options, $z_0 = \ln(K/S_0)$.

3.1.2 Hilbert Representation for CDF

In our application, the CDF $F(x)$ is usually unknown, not even mention its inverse function F^{-1}. However, thanks to the dedicated Lévy -Khinchine formula for infinitely divisible distributions, the characteristic function is usually explicitly given. There are many existing method to obtain CDF values by inverting characteristic function. Here we use the one proposed by Feng and Lin in [38]. Their method is based on the following Hilbert representation for CDF:

$$F(x) = \frac{1}{2} - \frac{i}{2}\mathcal{H}(e^{-i\xi x}\phi(\xi))(0). \tag{3.5}$$

And the representation could be approximated by using Trapezoidal's rule with discretization step h and truncation level M:

$$F_{h,M}(x) = \frac{1}{2} + \frac{i}{2}\sum_{m=-M}^{M} e^{-ix(m-1/2)h}\frac{\phi((m-1/2)h)}{(m-1/2)\pi}. \tag{3.6}$$

The most important advantage of this scheme is that its numerical error decays very fast under certain conditions. When the corresponding characteristic function ϕ falls in the analytic class $H(\mathcal{D}_{(d_-,d_+)})$ as we introduced in section 2.1, its discretization error decays exponentially in $1/h$. And its truncation error depends on the tail behavior of ϕ. Usually, there are two typical tail behaviors that are imposed on ϕ in our application, the exponential tail:

$$|\phi(\xi)| \leq \kappa \exp(-c|\xi|^\nu), \quad \xi \in \mathbb{R}. \tag{3.7}$$

or the polynomial tail:

$$|\phi(\xi)| \leq \frac{\kappa}{|\xi|^\nu}, \quad \xi \in \mathbb{R}. \tag{3.8}$$

Here c, ν, κ are all positive constants. For the former type, the truncation error is of order $O(e^{-c(Mh)^\nu})$; and for the latter type, the truncation error is of order $O((Mh)^{-\nu})$. We summarize and end this section by quoting the following theorem in [20], where it was presented without proof. And the idea of the theorem was originally from [38], with a little modification.

Theorem 3.1.2. *Let $F(x)$ and $\phi(\xi)$ be the cdf and the characteristic function of a continuous distribution. Suppose that $\phi \in H(\mathcal{D}_{(d_-,d_+)})$. Then*

$$F(x) = \frac{1}{2} - \frac{i}{2}\mathcal{H}(e^{-i\xi x}\phi(\xi))(0). \tag{3.9}$$

For any $a \in (d_-, 0)$,

$$1 - F(x) = e^{ax}\int_{\mathbb{R}} \frac{e^{-ix\xi}\phi(\xi + ia)}{2\pi i(\xi + ia)}d\xi. \tag{3.10}$$

For any $a \in (0, d_+)$,

$$F(x) = e^{ax}\int_{\mathbb{R}} \frac{e^{-ix\xi}\phi(\xi + ia)}{-2\pi i(\xi + ia)}d\xi. \tag{3.11}$$

If ϕ satisfies (3.7) for some $c, \nu, \kappa > 0$, then for any $h > 0$ and integer $M \geq 1$,

$$\begin{aligned} |F(x) - F_{h,M}(x)| \quad \leq \quad & \frac{e^{-2\pi|d_-|/h + xd_-}}{2\pi|d_-|(1 - e^{-2\pi|d_-|/h})}||\phi||^- + \frac{e^{-2\pi d_+/h + xd_+}}{2\pi d_+(1 - e^{-2\pi d_+/h})}||\phi||^+ \\ & + \frac{\kappa}{2\pi}\Big(\frac{1}{M} + \frac{2}{\nu c(Mh)^\nu}\Big)e^{-c(Mh)^\nu}. \end{aligned} \tag{3.12}$$

If ϕ satisfies (3.8) for some $\nu, \kappa > 0$, then for any $h > 0$ and integer $M \geq 1$,

$$\begin{aligned} |F(x) - F_{h,M}(x)| \quad \leq \quad & \frac{e^{-2\pi|d_-|/h + xd_-}}{2\pi|d_-|(1 - e^{-2\pi|d_-|/h})}||\phi||^- + \frac{e^{-2\pi d_+/h + xd_+}}{2\pi d_+(1 - e^{-2\pi d_+/h})}||\phi||^+ \\ & + \frac{\kappa}{\pi}\Big(\frac{1}{M} + \frac{2}{\nu}\Big)\frac{1}{(Mh)^\nu}. \end{aligned} \tag{3.13}$$

Proof. Representations (3.9)-(3.11) follow from Theorem 3.1 in [38]. The discretization error in (3.12) and (3.13) follows from the proof of Corollary 2.9 in [38]. And the truncation error bound in (3.12) can be derived through the following:

$$\begin{aligned} |F_{h,\infty}(x) - F_{h,M}(x)| \quad &= \quad \Big(\sum_{m=M+1}^{\infty} + \sum_{m=-\infty}^{-M-1}\Big)\frac{|\phi((m - 1/2)h)|}{\pi|m - 1/2|} \\ &\leq \quad \sum_{m=M+1}^{\infty}\frac{\kappa e^{-c((m-1)h)^\nu}}{(m-1)\pi} + \sum_{m=-\infty}^{-M-1}\frac{\kappa e^{-c(-mh)^\nu}}{-m\pi} \\ &= \quad \frac{\kappa e^{-c(-mh)^\nu}}{M\pi} + \sum_{m\geq M+1}^{\infty}\frac{2\kappa e^{-c(-mh)^\nu}}{m\pi} \\ &\leq \quad \frac{\kappa}{\pi}\Big(\frac{1}{M} + \frac{2}{\nu c(Mh)^\nu}\Big)e^{-c(Mh)^\nu}. \end{aligned}$$

Then, the truncation error in (3.13) is obtained by:

$$\begin{aligned}
|F_{h,\infty}(x) - F_{h,M}(x)| &= \Big(\sum_{m=M+1}^{\infty} + \sum_{m=-\infty}^{-M-1} \Big) \frac{|\phi((m-1/2)h)|}{\pi|m-1/2|} \\
&\leq \sum_{m=M+1}^{\infty} \frac{\kappa}{\pi(m-1)((m-1)h)^{\nu}} + \sum_{m=-\infty}^{-M-1} \frac{\kappa}{-\pi m(-mh)^{\nu}} \\
&= \frac{\kappa h}{\pi(Mh)^{\nu+1}} + \sum_{m=M+1}^{\infty} \frac{2\kappa h}{\pi(mh)^{\nu+1}} \\
&\leq \frac{\kappa h}{\pi(Mh)^{\nu+1}} + \frac{2\kappa}{\pi} \int_{Mh}^{\infty} \frac{1}{x^{\nu+1}} dx \quad = \quad \frac{\kappa}{\pi}\Big(\frac{1}{M} + \frac{2}{\nu}\Big) \frac{1}{(Mh)^{\nu}}.
\end{aligned}$$

We thus obtain the results in the theorem. □

Then, with this result, we are able to quantify the third one among those three type of errors introduced before. And even more interesting, the error decays very fast and is explicitly given. Hence, we can control the error to be less than any given tolerance level by choosing h and M properly.

3.1.3 Randomized quasi Monte Carlo

In our problem, in order to estimate (3.1), we apply Monte Carlo simulation in the following way:

$$\mathbb{E}[f(X_{t_1}, ..., X_{t_d})] \approx \frac{1}{N} \sum_{i=1}^{N} f(X^i_{t_1}, ..., X^i_{t_d}) \approx \frac{1}{N} \sum_{i=1}^{N} g_i(U^i_1, ..., U^i_d).$$

Here for each sample of g_i, d standard uniform random variables $U^i_1, ..., U^i_d$ need to be generated. Instead of using pseudo-random number sequence, quasi Monte Carlo sequences could be applied. In fact, both quasi and ordinary Monte Carlo methods try to estimate the integral of a function by an average function value:

$$\int_{[0,1]^d} f(u)du \approx \frac{1}{N} \sum_{i=1}^{N} f(u_i).$$

Hence, more evenly u_is are scattering in the $[0,1]^d$ space, better the approximation result should be. Since the quasi Monte Carlo is predetermined, usually it is designed in such a way that it has lower "discrepancy": a mathematical term used to estimate how evenly the sequence lying in the target space. To be more specific, as shown in [43]:

$$D(u_1, ..., u_N; A) = \sup_{A \in \mathcal{A}} |\frac{\#\{u_i \in A\}}{n} - vol(A)|,$$

where $\#\{u_i \in A\}$ denotes the number of sample u_i contained in A. And $vol(A)$ is the measure of A. This explains the fact that quasi Monte Carlo sequence is usually called low discrepancy sequence. According to [4], one of the most important advantage of low discrepancy sequence is that it has a convergence rate close to $O(1/N)$, much faster than $O(1/\sqrt{N})$ for ordinary Monte Carlo.

On the other hand, quasi Monte Carlo also has certain drawbacks, for example, the variance is hard to estimate since the sequence is predetermined. In this case, randomized quasi Monte Carlo method could be applied to alleviate this problem. And in this view of point, quasi Monte Carlo methods could be taken as a variance reduction technique [43]. Among methods in randomized quasi Monte Carlo, we choose digital shift method in our implementation [51; 52]. The idea of digital shift method is very simple, for a generated low discrepancy sequence $\{u_1^i, ..., u_d^i\}$, we generate an i.i.d standard uniform random variable sequence $\{v_1^j, ..., v_d^j\}$ by ordinary method. And then, the exclusive-or operation $\oplus$ is conducted between each (u, v) pair to generate the following new sequence:

$$\{u_1^i \oplus v_1^j, \cdots, u_d^i \oplus v_d^j\}, \quad 1 \leq i \leq N_0.$$

For each j fixed, we obtain one estimation by taking average of N_0 samples. And we repeat this process for L times, that is $j = 1, 2, ..., L$. Then the standard error could be estimated from the L estimations. And to sum up, $N = N_0 L$ samples in total are generated to estimate the expectation (3.1). Here we would like to mention that Sobol's sequence is used as our low discrepancy sequence to generate $\{u_1^i, ..., u_d^i\}$ for $1 \leq i \leq N_0$. And it is verified in [43] the good performance of Sobol's sequence in financial applications.

3.2 Bias Estimation

In this part, we further discuss the error introduced when we replace F with $\hat{F}$. First we provide a error bound for the total bias, which depends on those three error sources we discussed before in section (3.1.1): (i) the truncation level x_0 and x_K; (ii) the number of subinterval K or the subinterval length η, which could be also explained as the fineness of the grid; (iii) the estimation error caused by using $\hat{F}_k$ to approximate $F(x_k)$. After that, we discuss the specific situation where the Hilbert representation is used to calculate the CDF values. In this case, we are able to provide an explicit and computable error bound for the bias, which will be applied in our further discussion of numerical results. We will also consider the situation in high dimensional setting.

3.2.1 The total error bound

The main result is provided in the following theorem, which was developed by Chen, Feng and Lin in [20]. Before that, we introduce the notations used in the theorem. The approximation of distribution function F is $\hat{F}$, which was defined in (3.3). Then the corresponding probability density function has the following form:

$$\hat{p}(x) = \hat{F}_0 \cdot \delta(x - x_0) + (1 - \hat{F}_K) \cdot \delta(x - x_K) + \frac{1}{\eta} \sum_{k=1}^{K} (\hat{F}_k - \hat{F}_{k-1}) \cdot \mathbf{1}_{(x_{k-1}, x_k)}(x). \tag{3.14}$$

And we use the following notations:

$$|\mathcal{X}| = x_K - x_0, \quad ||f||_{\mathcal{X}} = \sup_{x \in \mathcal{X}} |f(x)|,$$

$$||f'||_{\mathcal{X}} = \operatorname{ess\,sup}_{x \in \mathcal{X}} |f'(x)|, \quad ||p'||_{\mathcal{X}} = \operatorname{ess\,sup}_{x \in \mathcal{X}} |p'(x)|,$$

$$E_{\mathcal{X}} = \sup_{x \in \mathcal{X}} |F(x) - \hat{F}(x)|.$$

The theorem is listed below:

Theorem 3.2.1. *Consider a continuous random variable X with cumulative distribution function $F(x)$ and density $p(x)$. Let $\{x_0, x_1, \cdots, x_K\}$ be a uniform grid with step size $\eta = (x_K - x_0)/K$ for some positive integer K. Let $\hat{X}$ be a random variable with distribution function $\hat{F}(x)$ defined as in (3.3). Suppose $f(x)$ is differentiable in $\bigcup_{i=1}^{K}(x_i, x_{i-1})$ except at n^f points, and $||f'||_{\mathcal{X}} < \infty$, $||f||_{\mathcal{X}} < \infty$. $p(x)$ is bounded on $\mathcal{X}$, differentiable in $\bigcup_{i=1}^{K}(x_i, x_{i-1})$, and $||p'||_{\mathcal{X}} < \infty$. Then*

$$\begin{aligned} |\mathbb{E}[f(X)] - \mathbb{E}[f(\hat{X})]| \;\le\; & |f(x_0)|F(x_0) + |f(x_K)|(1 - F(x_K)) + \Big(\int_{-\infty}^{x_0} + \int_{x_K}^{\infty} \Big) |f(x)|p(x)dx \\ & + \frac{1}{K^2} ||f'||_{\mathcal{X}} ||p'||_{\mathcal{X}} |\mathcal{X}|^3 \\ & + \big(|f(x_0)| + |f(x_K)| + 2(K + n^f)||f||_{\mathcal{X}} + 2||f'||_{\mathcal{X}}|\mathcal{X}| \big) E_{\mathcal{X}}. \end{aligned} \tag{3.15}$$

Proof. The bias is given by

$$\begin{aligned}
\mathbb{E}[f(X)] - \mathbb{E}[f(\hat{X})] &= \int_{\mathbb{R}} f(x)p(x)dx - \int_{\mathbb{R}} f(x)\hat{p}(x)dx \\
&= \int_{\mathbb{R}} f(x)p(x)dx - \int_{x_0}^{x_K} f(x)\hat{p}(x)dx - f(x_0)\hat{F}_0 - f(x_K)(1-\hat{F}_K) \\
&= \sum_{k=1}^{K}\int_{x_{k-1}}^{x_k} f(x)(p(x)-\hat{p}(x))dx + \Big(\int_{-\infty}^{x_0} + \int_{x_K}^{\infty}\Big) f(x)p(x)dx \\
&\quad -f(x_0)\hat{F}_0 - f(x_K)(1-\hat{F}_K). \qquad (3.16)
\end{aligned}$$

If $f(x)$ is differentiable in (x_{k-1}, x_k), then for any $x \in (x_{k-1}, x_k)$, by the mean value theorem, there exists $\xi_k(x) \in (x_{k-1}, x)$ such that

$$\begin{aligned}
\Big|\int_{x_{k-1}}^{x_k} f(x)(p(x)-\hat{p}(x))dx\Big| &= \Big|\int_{x_{k-1}}^{x_k} (f(x_{k-1}+) + f'(\xi_k(x))(x-x_{k-1}))(p(x)-\hat{p}(x))dx\Big| \\
&\le ||f||_{\mathcal{X}} \cdot |F_k - \hat{F}_k - (F_{k-1} - \hat{F}_{k-1})| \\
&+ ||f'||_{\mathcal{X}} \cdot \eta \cdot \int_{x_{k-1}}^{x_k} |p(x)-\hat{p}(x)|dx \\
&\le 2 \cdot ||f||_{\mathcal{X}} \cdot E_{\mathcal{X}} + ||f'||_{\mathcal{X}} \cdot \eta \cdot \int_{x_{k-1}}^{x_k} |p(x)-\hat{p}(x)|dx.
\end{aligned}$$

Here $f(x_{k-1}+)$ is the right limit of f at x_{k-1}, which is finite by the assumptions. In general, if $f(x)$ is not differentiable at n_k^f points in (x_{k-1}, x_k), where $\sum_{k=1}^{K} n_k^f = n^f$, it can be shown in the same way as above that

$$\Big|\int_{x_{k-1}}^{x_k} f(x)(p(x)-\hat{p}(x))dx\Big| \le 2(n_k^f+1) \cdot ||f||_{\mathcal{X}} \cdot E_{\mathcal{X}} + ||f'||_{\mathcal{X}} \cdot \eta \cdot \int_{x_{k-1}}^{x_k} |p(x)-\hat{p}(x)|dx.$$

Note that

$$\begin{aligned}
\int_{x_{k-1}}^{x_k} |p(x)-\hat{p}(x)|dx &= \int_{x_{k-1}}^{x_k} |p(x) - \frac{F_k - F_{k-1}}{\eta} + \frac{F_k - F_{k-1}}{\eta} - \frac{\hat{F}_k - \hat{F}_{k-1}}{\eta}|dx \\
&\le |F_k - \hat{F}_k - (F_{k-1} - \hat{F}_{k-1})| + \int_{x_{k-1}}^{x_k} |p(x) - \frac{F_k - F_{k-1}}{\eta}|dx \\
&\le 2E_{\mathcal{X}} + \int_{x_{k-1}}^{x_k} |p(x) - \frac{F_k - F_{k-1}}{\eta}|dx.
\end{aligned}$$

However, we have the following:

$$\int_{x_{k-1}}^{x_k} |p(x) - \frac{F_k - F_{k-1}}{\eta}|dx = \frac{1}{\eta}\int_{x_{k-1}}^{x_k} \Big|\int_{x_{k-1}}^{x_k} (p(x)-p(y))dy\Big|dx.$$

Since $p(x)$ is differentiable in (x_{k-1}, x_k), using the mean value theorem again, for any $x \neq y$ in (x_{k-1}, x_k), there exists $\xi_k(x, y) \in (x, y)$ such that

$$p(y) = p(x) + p'(\xi_k(x, y))(y - x).$$

Therefore,

$$\int_{x_{k-1}}^{x_k} |p(x) - \hat{p}(x)|dx \leq 2E_{\mathcal{X}} + ||p'||_{\mathcal{X}} \cdot \eta^2.$$

Consequently,

$$\Big|\sum_{k=1}^{K} \int_{x_{k-1}}^{x_k} f(x)(p(x) - \hat{p}(x))dx\Big| \leq 2E_{\mathcal{X}}\Big((K + n^f)||f||_{\mathcal{X}} + ||f'||_{\mathcal{X}}(x_K - x_0)\Big) + ||f'||_{\mathcal{X}}||p'||_{\mathcal{X}}(x_K - x_0)\eta^2.$$

As for the last two terms in (3.16), we have

$$|f(x_0)\hat{F}_0| \leq |f(x_0)|(E_{\mathcal{X}} + F_0), \quad |f(x_K)(1 - \hat{F}_K)| \leq |f(x_K)|(E_{\mathcal{X}} + 1 - F_K).$$

Combining the above, we obtain (3.15). □

Remark 3.2.2. *The total error bound is separated into three parts by three different lines in* (3.15)*. To control the total bias less than any given tolerance level ϵ, we can do the following: first, we choose x_0 and x_K properly so that the first line is less than $\epsilon/3$; then fix the chosen x_0 and x_K, we can choose the corresponding K such that the second line in* (3.15) *is less than $\epsilon/3$; finally, for the given set of x_0, x_K and K, we can decide the value of $E_{\mathcal{X}}$ to make the third error line less than $\epsilon/3$.*

3.2.2 Error bound when using Hilbert representation

In this part, we consider using the Hilbert representation method to approximate the CDF value. More specifically, (3.6) is used. And hence, $\hat{F}_k = F_{h,M}(x_k)$, and $E_{\mathcal{X}}$ is now replaced by:

$$E_{h,M,\mathcal{X}} = \sup_{x \in \mathcal{X}} |F(x) - F_{h,M}(x)|.$$

And of course, we assume the characteristic function ϕ falls in certain analytic class $H(\mathcal{D}_{(d_-,d_+)})$, and one of the tail behavior requirement (3.7) or (3.8) is satisfied as well. The following theorem summarizes the result we have, and was originally shown in [20].

Theorem 3.2.3. *Consider a continuous random variable X with characteristic function ϕ. Let*

$\{x_0, x_1, \cdots, x_K\}$ be a uniform grid with step size $\eta = (x_K - x_0)/K$ for some positive integer K. Suppose $\phi \in H(\mathcal{D}_{(d_-,d_+)})$ for some $-\infty < d_- < 0 < d_+ < \infty$, and satisfies (3.7) for some $\nu > 0$ or (3.8) for some $\nu > 2$. Let $\hat{X}$ be a random variable with distribution function $\hat{F}$ defined in (3.3), where $\hat{F}_k$ is given by $F_{h,M}(x_k)$ defined in (3.6). Suppose f is differentiable in $\bigcup_{i=1}^{K}(x_i, x_{i-1})$ except at n^f points, and $||f||_{\mathcal{X}} < \infty$, $||f'||_{\mathcal{X}} < \infty$. Then

$$\begin{aligned}
|\mathbb{E}[f(X)] - \mathbb{E}[f(\hat{X})]| \;\le\; & \frac{||\phi||^-}{2\pi}\Big(\int_{x_K}^{\infty} |f(x)|e^{xd_-}dx + \frac{1}{|d_-|}|f(x_K)|e^{x_K d_-}\Big) \\
& + \frac{||\phi||^+}{2\pi}\Big(\int_{-\infty}^{x_0} |f(x)|e^{xd_+}dx + \frac{1}{d_+}|f(x_0)|e^{x_0 d_+}\Big) \\
& + \frac{1}{2\pi K^2}||f'||_{\mathcal{X}}|\mathcal{X}|^3 \int_{\mathbb{R}} |\xi\phi(\xi)|d\xi \\
& + \big(|f(x_0)| + |f(x_K)| + 2(K + n^f)||f||_{\mathcal{X}} + 2||f'||_{\mathcal{X}}|\mathcal{X}|\big)\, E_{h,M,\mathcal{X}}.
\end{aligned} \tag{3.17}$$

Proof. Let F be the cdf of X. According to Theorem 3.2.1, the bias is bounded by (3.15). If $\phi \in H(\mathcal{D}_{(d_-,d_+)})$, from (3.11), for any $a \in (0, d_+)$, we have

$$f(x_0)F(x_0) = f(x_0)\int_{\mathbb{R}} \frac{e^{-ix_0(\xi+ia)}\phi(\xi+ia)}{-2\pi i(\xi+ia)}d\xi = f(x_0)\int_{-\infty+ia}^{+\infty+ia} \frac{e^{-ix_0 z}\phi(z)}{-2\pi iz}dz.$$

Since the integrand above is analytic in $\{z \in \mathbb{C} : \Im(z) \in (0, d_+)\}$, using the condition (2.1) and Cauchy's integral theorem, for any $\epsilon > 0$ such that $d_+ - \epsilon > a$, we have

$$f(x_0)F(x_0) = f(x_0)\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)} \frac{e^{-ix_0 z}\phi(z)}{-2\pi iz}dz = f(x_0)e^{(d_+-\epsilon)x_0}\int_{\mathbb{R}} \frac{e^{-ix_0\xi}\phi(\xi+i(d_+-\epsilon))}{-2\pi i(\xi+i(d_+-\epsilon))}d\xi.$$

Let $\epsilon \downarrow 0$, we obtain

$$|f(x_0)F(x_0)| \le \frac{||\phi||^+}{2\pi d_+}|f(x_0)|e^{x_0 d_+}.$$

Note that the probability density $p(x)$ admits the following inverse Fourier transform representation:

$$p(x) = \frac{1}{2\pi}\int_{\mathbb{R}} e^{-izx}\phi(z)dz.$$

By Cauchy's integral theorem and the condition (2.1), for any $\epsilon > 0$ such that $d_+ - \epsilon > 0$, we have

$$p(x) = \frac{1}{2\pi}\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)} e^{-izx}\phi(z)dz = e^{(d_+-\epsilon)x}\int_{\mathbb{R}} \frac{1}{2\pi}e^{-ix\xi}\phi(\xi+i(d_+-\epsilon))d\xi.$$

Consequently,

$$\int_{-\infty}^{x_0} |f(x)|p(x)dx \leq \frac{||\phi||^+}{2\pi} \int_{-\infty}^{x_0} |f(x)|e^{xd_+}dx.$$

Similarly, using the representation (3.10), we have the following:

$$|f(x_K)|(1 - F(x_K)) \leq \frac{||\phi||^-}{2\pi|d_-|}|f(x_K)|e^{x_K d_-},$$

$$\int_{x_K}^{\infty} |f(x)|p(x)dx \leq \frac{||\phi||^-}{2\pi} \int_{x_K}^{\infty} |f(x)|e^{xd_-}dx.$$

Since $\xi\phi(\xi)$ is absolutely integrable on $\mathbb{R}$ by the assumptions on ϕ,

$$p'(x) = \frac{1}{2\pi}\frac{d}{dx}\int_{\mathbb{R}} e^{-i\xi x}\phi(\xi)d\xi = \frac{1}{2\pi i}\int_{\mathbb{R}} e^{-i\xi x}\xi\phi(\xi)d\xi,$$

where the interchange of the integration and differentiation is valid due to the dominated convergence theorem. Therefore,

$$||p'||_{\mathcal{X}} \leq \frac{1}{2\pi}\int_{\mathbb{R}} |\xi\phi(\xi)|d\xi.$$

Combining the above, we obtain the bound for the bias in (3.17). □

Remark 3.2.4.

- We can see that the first line in (3.15) has been separated into two lines in (3.17). And actually, the first line only depends on x_K and the second line only depends on x_0. In this case, we can choose x_0 and x_K separately such that both terms are less than a predetermined tolerance level, for example $\epsilon/4$.

- When certain condition of ϕ is satisfied, as listed in the theorem, the Hilbert representation method shown in theorem 3.1.2 is very accurate. Hence, $E_{h,M,\mathcal{X}}$ could be smaller than 1E-10. In this case, we can set the third term in the error bound to be less than 0.01ϵ, so as to give more freedom to the first two types of error, say, instead of $\epsilon/3$, we can set them to be less than $\epsilon/2$.

3.2.3 The multidimensional case

For the multidimensional case, we can still work out an error bound for the total bias, which has similar form as the single dimensional case. However, constants in the error bound is hard to calculate in practice when the dimension d is very large. To solve this problem, we can try different

values of ϵ until we observe the desired convergence. We will see later in numerical examples that this procedure works very well. As before, we first quote the result from [20] directly:

Theorem 3.2.5. *Let $X_i, 1 \le i \le d$, be d independent random variables with distributions $F^{(i)}$ and densities p_i. For $1 \le i \le d$, let $\{x_0^{(i)}, \cdots, x_{K_i}^{(i)}\}$ be a uniform grid with step size $\eta_i = (x_{K_i}^{(i)} - x_0^{(i)})/K_i$ for some integer $K_i > 0$. Let $\hat{X}_i, 1 \le i \le d$, be d independent random variables with distributions $\hat{F}^{(i)}$ defined in (3.3) on $\mathcal{X}_i = [x_0^{(i)}, x_{K_i}^{(i)}]$. For any $1 \le i \le d$, $f(x_1, \cdots, x_d)$ as a function of x_i is differentiable in $\mathcal{X}_i$ except at up to n^f points, and both $|f(x_1, \cdots, x_d)|$ and $|df(x_1, \cdots, x_d)/dx_i|$ are bounded by $g_1(x_1)g_2(x_2)\cdots g_d(x_d)$ for some $g_i \ge 0$. For $1 \le i \le d$, g_i is differentiable in $\mathcal{X}_i$ except at up to n^f points and*

$$||g_i||_{\mathcal{X}_i}^* = \max\Big(\sup_{x \in \mathcal{X}_i} |g_i(x)|, \operatorname{ess\,sup}_{x \in \mathcal{X}_i} |g_i'(x)| \Big) < \infty.$$

Denote $||p_i g_i||_1 = \int_{\mathbb{R}} p_i(x) g_i(x) dx, 1 \le i \le d$. Define

$$\begin{aligned} B_i \quad = \quad & g_i(x_0^{(i)}) F^{(i)}(x_0^{(i)}) + g_i(x_{K_i}^{(i)})(1 - F^{(i)}(x_{K_i}^{(i)})) + \Big(\int_{-\infty}^{x_0^{(i)}} + \int_{x_{K_i}^{(i)}}^{\infty} \Big) g_i(x) p_i(x) dx \\ & + \frac{1}{K_i^2} ||g_i||_{\mathcal{X}_i}^* ||p_i'||_{\mathcal{X}_i} |\mathcal{X}_i|^3 \qquad (3.18) \\ & + \Big(g_i(x_0^{(i)}) + g_i(x_{K_i}^{(i)}) + 2(K_i + n^f) ||g_i||_{\mathcal{X}_i}^* + 2||g_i||_{\mathcal{X}_i}^* |\mathcal{X}_i| \Big) E_{\mathcal{X}_i}, \end{aligned}$$

where $||p_i'||_{\mathcal{X}_i}, |\mathcal{X}_i|, E_{\mathcal{X}_i}$ are defined similarly as in Theorem 3.2.1. Let $B = \max(B_1, \cdots, B_d)$. Then there exist constants $a_i > 0, 0 \le i \le d-1$, independent of B such that

$$|\mathbb{E}[f(X_1, \cdots X_d)] - \mathbb{E}[f(\hat{X}_1, \cdots, \hat{X}_d)]| \le B(a_0 + a_1 B + \cdots + a_{d-1} B^{d-1}). \qquad (3.19)$$

Proof. Denote the density of $\hat{F}^{(i)}$ by $\hat{p}_i$. From Theorem 3.2.1, we have the following for any $1 \le i \le d$:

$$\int_{\mathbb{R}} g_i(x) \hat{p}_i(x) dx \le ||p_i g_i||_1 + B_i,$$

$$\Big| \int_{\mathbb{R}} f(x_1, \cdots, x_d)(p_i(x_i) - \hat{p}_i(x_i)) dx_i \Big| \le B_i \prod_{j=1, j \ne i}^{d} g_j(x_j).$$

Following [45], $|\mathbb{E}[f(X_1,\cdots X_d)]-\mathbb{E}[f(\hat{X}_1,\cdots,\hat{X}_d)]|$ is bounded by the following:

$$\begin{aligned}
&\Big|\mathbb{E}[f(X_1,X_2,\cdots,X_{d-1},X_d)]-\mathbb{E}[f(X_1,X_2,\cdots,X_{d-1},\hat{X}_d)]\Big|\\
+&\Big|\mathbb{E}[f(X_1,X_2,\cdots,X_{d-1},\hat{X}_d)]-\mathbb{E}[f(X_1,X_2,\cdots,\hat{X}_{d-1},\hat{X}_d)]\Big|+\\
&\vdots\\
+&\Big|\mathbb{E}[f(X_1,\hat{X}_2,\cdots,\hat{X}_{d-1},\hat{X}_d)]-\mathbb{E}[f(\hat{X}_1,\hat{X}_2,\cdots,\hat{X}_{d-1},\hat{X}_d)]\Big|
\end{aligned}$$

$$\begin{aligned}
\leq\ & B_d\prod_{j=1}^{d-1}||p_jg_j||_1+B_{d-1}\prod_{j=1}^{d-2}||p_jg_j||_1||\hat{p}_dg_d||_1+\cdots+B_1\prod_{j=2}^{d}||\hat{p}_jg_j||_1\\
\leq\ & B\Big(\prod_{j=1}^{d-1}||p_jg_j||_1+\prod_{j=1}^{d-2}||p_jg_j||_1(||p_dg_d||_1+B)+\cdots+\prod_{j=2}^{d}(||p_jg_j||_1+B)\Big),
\end{aligned}$$

where $||\hat{p}_ig_i||_1=\int_{\mathbb{R}}\hat{p}_i(x)g_i(x)dx, 1\leq i\leq d$. The conclusion then follows immediately. □

Remark 3.2.6.

- As we stated before, it is usually very hard to determine the positive constants a_i for $i=0,...,d-1$ when d is large. However, at least we know the total error bound is a polynomial function of B, which we are able to bound. Hence, by trying different error bound for B, we should be able to observe the convergence.

- When ϕ satisfies certain conditions, we can also apply the Hilbert method in high dimensional case, and we will have new representation for B_i, where :

$$\begin{aligned}
B_i\ =\ & \frac{||\phi^{(i)}||^-}{2\pi}\Big(\int_{x^{(i)}_{K_i}}^{\infty}g_i(x)e^{xd^{(i)}_-}dx+\frac{1}{|d^{(i)}_-|}g_i(x^{(i)}_{K_i})e^{x^{(i)}_{K_i}d^{(i)}_-}\Big)\\
&+\frac{||\phi^{(i)}||^+}{2\pi}\Big(\int_{-\infty}^{x^{(i)}_0}g_i(x)e^{xd^{(i)}_+}dx+\frac{1}{d^{(i)}_+}g_i(x^{(i)}_0)e^{x^{(i)}_0d^{(i)}_+}\Big)\\
&+\frac{1}{2\pi K_i^2}||g_i||^*_{\mathcal{X}_i}|\mathcal{X}_i|^3\int_{\mathbb{R}}|\xi\phi^{(i)}(\xi)|d\xi\\
&+\Big(g_i(x^{(i)}_0)+g_i(x^{(i)}_{K_i})+2(K_i+n^f)||g_i||^*_{\mathcal{X}_i}+2||g_i||^*_{\mathcal{X}_i}|\mathcal{X}_i|\Big)E_{h_i,M_i,\mathcal{X}_i}.
\end{aligned}\tag{3.20}$$

- Although in general, distribution function $F^{(i)}$, $1\leq i\leq d$ could be different, but in financial applications, they are usually the same. For example, when pricing options, discretely monitored path-dependent options are often evenly monitored and hence has the same Lévy

increments. Then, all g_is are the same. This result could save us great time and computational effort since (3.18) and (3.20) are all the same for different $1 \leq i \leq d$.

3.3 Numerical example of option pricing

In this part, we will verify previous theory and demonstrate the efficiency of our method by pricing various option contacts in Lévy process models. Here we will consider three Lévy process models. The first one is Kou's jump diffusion model [49], which is a finite activity jump diffusion process. The process can be simulated in a very natural way, which we will show later. The second one is NIG (Normal inverse Gaussian) process [7], which is an infinite activity pure jump process. There are two methods for simulating NIG process: by its subordination structure [66] or through bridge sampling [62]. We will implement both and compare with our method. And for this part, we will price both European vanilla option and path-dependent lookback option. The third one is CGMY model [14] which is also an infinite activity pure jump process. And we did not find a natural way to simulate CGMY process except simulating from its characteristic function. In this part, we will price discretely monitored Asian option and verify the power of various variance reduction techniques like RQMC and control variate. The computer we use to implement is Lenovo laptop T61p with Intel Core 2 Duo 2.5GHz CPU and 3GB RAM.

3.3.1 Kou's jump diffusion model

Kou's jump diffusion model is very much like Merton's jump diffusion model [57]. The process can be written as:

$$X_t = \mu t + \sigma B_t + \sum_{i=1}^{N_t} Z_i,$$

where $\mu = r - q - \frac{1}{2}\sigma^2 - \lambda(p/(\eta_1 - 1) - (1-p)/(\eta_2 + 1))$ is determined by the martingale condition. Also, random processes B_t, N_t and Z_i are independent, where B_t is a standard Brownian motion, N_t is a Poisson process with positive jump intensity λ. And Z_i are i.i.d double exponential process with intensity:

$$p\eta_1 e^{-\eta_1 x}\mathbf{1}_{\{x>0\}} + (1-p)\eta_2 e^{\eta_2 x}\mathbf{1}_{\{x<0\}},$$

where $0 \leq p \leq 1$. Moreover, the characteristic function of Kou's model is given by:

$$\phi_t(\xi) = \exp\Big(-\frac{1}{2}\sigma^2 t\xi^2 + i\mu t\xi + i\lambda t\xi\Big(\frac{p}{\eta_1 - i\xi} - \frac{1-p}{\eta_2 + i\xi}\Big)\Big),$$

which is in $H(\mathcal{D}_{(d_-,d_+)})$ with $d_+ = \eta_2$ and $d_- = -\eta_1$. Also, it has the exponential tail behavior we assumed previously, for $\kappa = 1$, $c = \frac{1}{2}\sigma^2 t$, $\nu = 2$ as in (3.7). The algorithm for simulating Kou's double exponential jump diffusion process is very intuitive, as the one provided in [20]:

Algorithm 3.3.1 (Simulating Kou's jump diffusion process)**.**

For $t > 0$, simulate $X_t = \mu t + \sigma B_t + \sum_{i=1}^{N_t} Z_i$

1. Use inverse transform method to generate Poisson process N_t, as shown in p.128 of [43].
2. For $1 \leq i \leq N_t$, generate Z_i by:
   - Generate standard uniform random variable U_i,
   - If $U_i \leq p$, $Z_i \sim exp(\eta_1)$; Otherwise, $Z_i \sim -exp(\eta_2)$.
3. Generate $G \sim N(0, t)$.
4. Set $X_t = \mu t + \sigma G + \sum_{i=1}^{N_t} Z_i$.

Remark 3.3.2. *We can see that on average $2 + 2\lambda t$ random variables are generated, which leads to much heavier computational cost than the method we carry out, since we only need to generate one single random variable for each iteration.*

We implement algorithm 3.3.1 and our method to price European put option, with the following parameters:

$$\sigma = 0.1, \lambda = 3, p = 0.3, \eta_1 = 40, \eta_2 = 12, r = 0.05, q = 0.02, S_0 = K = 100, T = 1.$$

In this case, $z_0 = \ln(K/S_0) = 0$, hence, we can set $x_K = 0$ directly. And for x_0, we can obtain it by considering the following in (3.17):

$$\frac{||\phi_T||^+}{2\pi}\Big(\int_{-\infty}^{x_0} |f(x)|e^{xd_+}dx + \frac{1}{d_+}|f(x_0)|e^{x_0 d_+}\Big) = \frac{||\phi_T||^+}{2\pi d_+}e^{x_0 d_+}\Big(2 - \frac{2d_+ + 1}{d_+ + 1}e^{x_0}\Big),$$

where $f(x) = (0, K/S_0 - e^x)^+$ for European put. For ϵ_b = 1E-2, 1E-3, 1E-4, we choose a x_0 such that the above quantity is less than $\epsilon_b/2$. Then, from the third line in (3.17), we are able to choose an proper K such that the error term is also less than $\epsilon_b/2$. And the corresponding K can be easy calculated as:

$$K = \Big(\frac{1}{\pi\epsilon_b}||f'||_{\mathcal{X}}|\mathcal{X}|^3 \int_{\mathbb{R}} |\xi\phi_T(\xi)|d\xi\Big)^{1/2}.$$

ϵ_b	x_0	x_K	K	$E_{h,M,\mathcal{X}}$	h	M
10^{-2}	-0.715	0	31	3.0×10^{-6}	3.651	13
10^{-3}	-1.029	0	167	4.6×10^{-8}	2.803	19
10^{-4}	-1.332	0	774	8.8×10^{-10}	2.295	27

Table 3.1: Grid parameters used for European put in Kou's model

It is easy to see that $||f'||_{\mathcal{X}} = 1$, then we can calculate K accordingly. And after that, we can obtain the desired value of $E_{h,M,\mathcal{X}}$, such that the fourth line in (3.17):

$$(|f(x_0)| + |f(x_K)| + 2K||f||_{\mathcal{X}} + 2||f'||_{\mathcal{X}}(x_K - x_0))E_{h,M,\mathcal{X}} = ((2K+1)(1-e^{x_0}) + 2(x_K - x_0))E_{h,M,\mathcal{X}}.$$

should be less than $0.01\epsilon_b$. When we obtain the desired value of $E_{h,M,\mathcal{X}}$, we can choose the corresponding h and M according to (3.12). We record all those values in table 3.1.

Also, with these parameters, we are able to compare our method with algorithm 3.3.1. The result is provided in table 3.2. Here N is number of samples (in thousands), SE is the standard error, CPU is the computational time in seconds, $Error$ stands for the absolutely pricing error, where we obtain the benchmark price 5.98007999 by the Hilbert transform method in [38]. For the $Gain$, it is defined by

$$\frac{\sigma_A^2 \cdot t_A}{\sigma_B^2 \cdot t_B},$$

where σ_A and σ_B are the standard errors. A is for algorithm 3.3.1 and B is for our method. And t_A, t_B are corresponding computational times.

From the numerical result, we can see that our inverse transform method is about 3 to 4 times better than algorithm 3.3.1 in terms of the gain. Another observation is about the absolute pricing error, which has two parts, the bias and standard error. When N is getting larger, the standard error is getting smaller and converging to zero. Hence, when N is large enough, the absolute pricing error basically comes from bias. For the case ϵ_b = 1E-2, the absolute pricing error stops to decrease at around 1.1E-2 and begin to oscillate around this number. For the case ϵ_b = 1E-3, the stopping point is 1.4E-3. It is worth noting that the targeted total bias for option price is actually $S_0 e^{-rT}\epsilon_b$ rather than ϵ_b. Since the latter is the tolerance level for bias of the expectation $\mathbb{E}[f(X)]$.

	Algorithm 3.3.1				$\epsilon_b = 10^{-2}$			
N ($\times 10^3$)	SE	Error	CPU	Gain	SE	Error	CPU	Gain
64	4.0E-2	3.5E-2	0.021		4.0E-2	7.1E-3	0.008	2.6
256	2.0E-2	3.8E-2	0.082		2.0E-2	2.8E-2	0.022	3.7
1024	1.0E-2	5.1E-3	0.335		1.0E-2	1.6E-2	0.075	4.5
4096	5.0E-3	4.7E-3	1.332		5.0E-3	1.1E-2	0.297	4.5
16384	2.5E-3	2.2E-3	5.380		2.5E-3	1.4E-2	1.201	4.5
65536	1.3E-3	2.0E-3	21.270		1.2E-3	1.3E-2	4.846	5.2
	$\epsilon_b = 10^{-3}$				$\epsilon_b = 10^{-4}$			
64	4.0E-2	5.0E-2	0.008	2.6	4.0E-2	2.6E-2	0.015	1.4
256	2.0E-2	2.8E-2	0.024	3.4	2.0E-2	3.2E-2	0.033	2.5
1024	1.0E-2	3.3E-2	0.089	3.8	1.0E-2	1.5E-3	0.105	3.2
4096	5.0E-3	9.9E-3	0.341	3.9	5.0E-3	8.1E-3	0.394	3.4
16384	2.5E-3	1.4E-3	1.349	4.0	2.5E-3	2.8E-3	1.512	3.6
65536	1.3E-3	3.4E-3	5.377	4.0	1.3E-3	4.0E-4	6.029	3.5

Table 3.2: European vanilla put in Kou's model: inverse transform method vs Algorithm 3.3.1.

3.3.2 The Normal Inverse Gaussian model

The normal inverse Gaussian (NIG) process is given by

$$X_t = \mu t + \beta z_t + B_{z_t}$$

where z_t is an inverse Gaussian processes, B_t is a standard Brownian motion. And they are independent with each other. It worths noting that z_t has the same distribution as the first passage time of a Brownian motion $BM(\gamma, 1)$ to the level δt [4]. The characteristic function of of NIG process is:

$$\phi_t(\xi) = \mathbb{E}[e^{i\xi X_t}] = \exp\Big(i\mu t\xi - \delta t(\sqrt{\alpha^2 - (\beta + i\xi)^2} - \sqrt{\alpha^2 - \beta^2})\Big).$$

where $\alpha = \sqrt{\beta^2 + \gamma^2}$, and $\mu = r - q + \delta(\sqrt{\alpha^2 - (\beta + 1)^2} - \sqrt{\alpha^2 - \beta^2})$ is determined by martingale condition. As Kou's model, the characteristic function of NIG process is in $H(\mathcal{D}_{(d_-, d_+)})$, where $d_- = \beta - \alpha$ and $d_+ = \beta + \alpha$. Also, it has the exponential tail behavior we assumed previously, for $\kappa = exp(\delta t\sqrt{\alpha^2 - \beta^2})$, $c = \delta t$, $\nu = 1$ as in (3.7).

Hence, it is very natural to simulate NIG process by its subordination structure: generate z_t first and then generate a standard normal random variable with variance z_t. The following algorithm is provided in [4] and [66], then used by Chen, Feng and Lin in [20] :

Algorithm 3.3.3 (Simulating NIG process by its subordination structure)**.**

For $t > 0$, simulate $X_t = \mu t + \beta z_t + B_{z_t}$

1. Generate $G_1 \sim N(0,1)$ by Beasley-Springer-Moro algorithm in [43].

2. Set $Z = G_1^2/\gamma$, and $\zeta = \frac{1}{\gamma}(\delta t + \frac{1}{2}Z - \sqrt{\delta t Z + Z^2/4})$.

3. Generate $U \sim U(0,1)$. If $U < \delta t/(\delta t + \gamma\zeta)$, $z_t = \zeta$. Otherwise, $z_t = \delta^2 t^2/(\gamma^2\zeta)$.

4. Generate $G_2 \sim N(0,1)$. Let $X_t = \mu t + \beta z_t + \sqrt{z_t}G_2$.

For single dimensional problem, like pricing European options, the above algorithm could be used. For multi-dimensional case, one can employ inverse Gaussian bridge together with quasi Monte Carlo as shown in [62]. This method could possibly decrease the effective dimension and could be used for pricing path-dependent options like lookback option. Still, we quote the following algorithm from [20]:

Algorithm 3.3.4 (Simulating a NIG process using inverse Gaussian bridge)**.**

For $0 = t_0 < t_1 < \cdots < t_d$, simulate $X_{t_j} = \mu t_j + \beta z_{t_j} + B_{z_{t_j}}, 1 \le j \le d$

1. Generate z_{t_d} by following steps 1-3 in Algorithm 3.3.3. Generate a standard normal random variable G_d and obtain $W_{z_{t_d}} = \beta z_{t_d} + \sqrt{z_{t_d}}G_d$.

2. Construct the bridge as below: for $0 \le t_i < t_j < t_k \le t_d$,

   (a) Inverse Gaussian bridge: conditional on z_{t_i} and z_{t_k}, generate a standard normal random variable G_{1j}, and compute

   $$\lambda = \frac{\delta^2(t_k - t_j)^2}{z_{t_k} - z_{t_i}},\ \theta = \frac{t_k - t_j}{t_j - t_i},\ Q = G_{1j}^2,\ s_1 = \theta + \frac{\theta^2 Q}{2\lambda} - \frac{\theta}{2\lambda}\sqrt{4\theta\lambda Q + \theta^2 Q^2}.$$

   Generate a uniform random variable U_j on $(0,1)$. If $U_j < \frac{\theta(1+s_1)}{(1+\theta)(\theta+s_1)}$, $s = s_1$; otherwise, $s = \theta^2/s_1$. Then

   $$z_{t_j} = z_{t_i} + \frac{z_{t_k} - z_{t_i}}{1+s}.$$

   (b) Brownian bridge (see [43] Chapter 3): conditional on $W_{z_{t_i}}$ and $W_{z_{t_k}}$, generate a standard normal random variable G_{2j}, and compute

   $$m = \frac{(z_{t_k} - z_{t_j})W_{z_{t_i}} + (z_{t_j} - z_{t_i})W_{z_{t_k}}}{z_{t_k} - z_{t_i}}, \sigma^2 = \frac{(z_{t_j} - z_{t_i})(z_{t_k} - z_{t_j})}{z_{t_k} - z_{t_i}},\ W_{z_{t_j}} = m + \sigma G_{2j}.$$

3. Let $X_{t_j} = \mu t_j + W_{z_{t_j}}, 1 \le j \le d$.

ϵ_b	x_0	x_K	K	$E_{h,M,\mathcal{X}}$	h	M
10^{-2}	-0.477	0	22	5.5×10^{-6}	4.926	11
10^{-3}	-0.736	0	133	7.1×10^{-8}	3.630	20
10^{-4}	-0.983	0	645	1.2×10^{-9}	2.903	30

Table 3.3: Grid parameters used for European put options in the NIG model

	Algorithm 3.3.3				$\epsilon_b = 10^{-2}$			
N ($\times 10^3$)	SE	Error	CPU	Gain	SE	Error	CPU	Gain
64	3.0E-2	3.0E-2	0.036		2.9E-2	7.1E-3	0.007	5.5
256	1.5E-2	3.6E-2	0.139		1.5E-2	7.2E-3	0.026	5.3
1024	7.4E-3	1.4E-2	0.545		7.3E-3	7.8E-3	0.096	5.8
4096	3.7E-3	1.1E-2	2.191		3.7E-3	7.6E-3	0.387	5.7
16384	1.9E-3	1.8E-3	8.640		1.8E-3	7.1E-3	1.576	6.1
65536	9.3E-4	5.9E-4	34.382		9.2E-4	7.4E-3	6.162	5.7
	$\epsilon_b = 10^{-3}$				$\epsilon_b = 10^{-4}$			
64	3.0E-2	4.8E-2	0.009	4.0	3.0E-2	3.2E-2	0.016	2.3
256	1.5E-2	3.2E-4	0.029	4.8	1.5E-2	6.3E-3	0.040	3.5
1024	7.4E-3	6.8E-3	0.111	4.9	7.4E-3	1.3E-3	0.132	4.1
4096	3.7E-3	5.9E-3	0.444	4.9	3.7E-3	5.1E-4	0.494	4.4
16384	1.9E-3	1.7E-3	1.777	4.9	1.9E-3	1.1E-4	1.965	4.4
65536	9.3E-4	1.5E-3	7.108	4.8	9.3E-4	2.8E-4	7.745	4.4

Table 3.4: European vanilla put in the NIG model: inverse transform method vs Algorithm 3.3.3.

Now, we price European option using NIG model. The parameters are the same as in [40]:

$$\alpha = 15, \beta = -5, \delta = 0.5, r = 0.05, q = 0.02, S_0 = K = 100, T = 0.5.$$

Follow the same process as previous in Kou's model, we are able to obtain those grid parameters used, as in table 3.3. For more detail, please refer to [20]:

And we are also able to compare algorithm 3.3.3 with our method. The result is shown in table 3.4. In this case, the gains, which are around 5, are even better than Kou's model.

Now we consider pricing floating strike lookback put options in NIG model. Here we assume the discrete monitoring dates are equally spaced, each with length Δ. Hence, if there are d monitoring dates and denote the maturity by T, then $\Delta = T/d$. The floating strike lookback option price is given by:

$$V = e^{-rT}\mathbb{E}[\max(S_0, S_\Delta, \cdots, S_{d\Delta}) - S_{d\Delta}] = S_0 e^{-rT}\mathbb{E}[f(Y_1, \cdots, Y_d)],$$

where $Y_i = X_{i\Delta} - X_{(i-1)\Delta}$, $1 \le i \le d$ are the Lévy increments. Also,

$$f(y_1, \cdots, y_d) = \max(1, e^{y_1}, e^{y_1+y_2}, \cdots, e^{y_1+\cdots+y_d}) - e^{y_1+\cdots+y_d}.$$

ϵ_b	x_0	x_K	K	$E_{h,M,\mathcal{X}}$	h	M
10^{-2}	-0.721	0.340	273	8.8×10^{-8}	3.127	76
10^{-3}	-0.977	0.461	1546	1.2×10^{-9}	2.456	123
10^{-4}	-1.233	0.582	7875	1.8×10^{-11}	2.028	180

Table 3.5: Grid parameters used for Lookback put options in the NIG model

	Algorithm 3.3.4 with RQMC					$\epsilon_b = 10^{-2}$				
N ($\times 10^3$)	SE	Error	CPU	Price	Gain	SE	Error	CPU	Price	Gain
16	1.5E-2	1.2E-2	0.049	10.1985		1.8E-2	2.4E-3	0.024	10.1837	1.4
64	1.2E-2	9.0E-3	0.186	10.1951		6.4E-3	7.6E-3	0.085	10.1937	7.7
256	3.5E-3	3.3E-3	0.737	10.1828		2.6E-3	1.9E-3	0.332	10.1880	4.0
1024	1.9E-3	1.4E-3	2.847	10.1847		6.3E-4	1.2E-3	1.320	10.1873	19.6
4096	4.9E-4	7.0E-4	11.446	10.1854		1.8E-4	2.8E-3	5.290	10.1889	16.0
16384	2.0E-4	8.2E-5	45.787	10.1862		7.4E-5	2.6E-3	21.221	10.1887	15.8
65536	9.5E-5	7.0E-5	183.005	10.1862		4.3E-5	2.5E-3	84.185	10.1886	10.6
	$\epsilon_b = 10^{-3}$					$\epsilon_b = 10^{-4}$				
16	1.3E-2	3.6E-2	0.028	10.1500	2.3	1.6E-2	1.4E-2	0.040	10.1717	1.1
64	5.8E-3	2.0E-3	0.100	10.1842	8.0	4.9E-3	1.8E-3	0.113	10.1879	9.9
256	2.0E-3	1.1E-3	0.373	10.1873	6.1	1.3E-3	1.7E-3	0.423	10.1844	12.6
1024	9.6E-4	4.1E-4	1.464	10.1857	7.6	8.0E-4	7.5E-4	1.650	10.1869	9.7
4096	3.2E-4	3.9E-4	5.818	10.1865	4.6	1.6E-4	2.4E-4	6.574	10.1864	16.3
16384	7.5E-5	3.9E-5	23.307	10.1862	14.0	5.0E-5	4.6E-5	25.973	10.1862	28.2
65536	4.1E-5	1.2E-4	94.097	10.1862	10.4	3.6E-5	1.0E-5	103.713	10.1861	12.3

Table 3.6: Discrete floating strike lookback put in the NIG model: inverse transform method with RQMC vs Algorithm 3.3.4 with RQMC.

One can verify that taking $g_i(x) = e^{|x|}$ for all $1 \le i \le d$ satisfies the assumption of theorem 3.2.5, and in this case $n^f = 1$.

Now we take $T = 1$ year, $d = 8$. Since the monitoring dates are equally spaced, for all $\phi^{(i)}$ in (3.20), we can use the same ϕ_Δ. Following the same idea as for the European option case, we can obtain the grid parameters as in table 3.5.

Notice that in this case, x_0 and x_K are chosen so that the first two terms in (3.20) are less than $\epsilon_b/4$. We show the numerical result in table 3.6. Here both methods are implemented with digital shift RQMC as shown in section 3.1.3. We use $L = 10$, following the suggestion in [51]. Here the benchmark price is 10.18611401, calculated by the method proposed in [41]. We would like to mention that, in this case, $K + 1 = 7876$ is very large. To reduce the computational cost in calculating the CDF, we use FFT with Teoplitz matrix vector multiplication algorithm. Without FFT, the computational time for ϵ_b = 1E-2, 1E-3 and 1E-4 are 0.008, 0.071 and 0.528 seconds respectively. When FFT is implemented, the computational time reduce to 0.002, 0.004, 0.012 seconds, which is a significant improvement. Also, it can be easily observed that our method is around 10 times better than the RQMC method (with bridge) in terms of Gain.

3.3.3 CGMY model

For CGMY process X_t, there is no explicit formula for its density. Also, there is no specific structure like subordination to simulate from. Hence, simulating from characteristic function seems inevitable. The c.f. of CGMY process is given by:

$$\phi_t(\xi) = \exp(i\mu t\xi - tC\Gamma(-Y)(\mathbb{M}^Y - (\mathbb{M} - i\xi)^Y + G^Y - (G + i\xi)^Y)),$$

where $C > 0, G > 0, \mathbb{M} > 0, 0 < Y < 2$, and $\mu = r - q - C\Gamma(-Y)((\mathbb{M} - 1)^Y - \mathbb{M}^Y + (G + 1)^Y - G^Y)$ is determined by martingale condition. Similar as Kou's and NIG model, the characteristic function of CGMY process is in $H(\mathcal{D}_{(d_-,d_+)})$, where $d_- = -\mathbb{M}$ and $d_+ = G$. Also, it has the exponential tail behavior we assumed previously. For example, when $0 < Y < 1$, $\kappa = \exp(-tC\Gamma(-Y)(\mathbb{M}^Y + G^Y))$, $c = 2tC|\Gamma(-Y)\cos(\pi Y/2)|$, $\nu = Y$ as in (3.7). For more detail, one can refer to [14] and [37].

We now price (arithmetic average) discretely monitored Asian options. Similar as the previous case for lookback option, we assume the monitoring dates are equally spaced with length Δ, and we will have $\Delta = T/d$, where T is the option maturity and d is number of monitoring dates. The Asian option price is:

$$V = e^{-rT}\mathbb{E}[\max(0, \frac{1}{d}\sum_{i=1}^{d} S_{i\Delta} - K)] = S_0 e^{-rT}\mathbb{E}[f(Y_1, \cdots, Y_d)],$$

where $Y_i = X_{i\Delta} - X_{(i-1)\Delta}$, $1 \le i \le d$ are the Lévy increments. Also,

$$f(y_1, \cdots, y_d) = \max\Big(0, \frac{1}{d}(e^{y_1} + e^{y_1+y_2} \cdots + e^{y_1+\cdots+y_d}) - \frac{K}{S_0}\Big).$$

One can verify that similar as in the lookback option case, we can choose $g_i(x) = e^{|x|}$ for all $1 \le i \le d$, and in this case $n^f = 1$.

Moreover, it is well known that geometric average Asian option price is a very good control variate for pricing arithmetic average Asian option. Also, due to its special structure, geometric average Asian option is usually much easier to price. In fact, the pricing structure of the geometric average Asian option is very much like an European option when a variable change is conducted. And hence can be priced by Hilbert transform method very fast and accurate. Please refer to [20] section 5.6.1 and [38] for more detail.

Now let us denote the (benchmark) geometric average Asian option price by V^g. Notice that the geometric average Asian option could also be priced by Monte Carlo simulation: after we generate

a sample path of the underlying asset price, we can use it to get an arithmetic average Asian option price V_i^a as well as a geometric average Asian option price V_i^g. It is easy to observe that, due to the similar structure, two options tend to be over-priced or under-priced together. Hence, we could adjust the arithmetic average Asian option price accordingly if we assume their pricing error has certain linear relationship:

$$V_i^b := V_i^a + b(V^g - V_i^g).$$

Here b is a positive constant estimating the ratio of pricing errors between arithmetic and geometric average Asian price. If N sample paths are generated, the adjusted arithmetic average Asian price is given by:

$$\bar{V}_N^b = \frac{1}{N}\sum_{i=1}^{N} V_i^b.$$

It is easy to prove that $\bar{V}_N^b$ is an unbiased estimator of arithmetic average Asian option. And also, if we let $b = \rho\sigma_a/\sigma_g$, where σ_a, σ_g and ρ are standard deviation of arithmetic average Asian option price, geometric average Asian option price and the correlation coefficient between them respectively. And if we further define

$$\bar{V}_N = \frac{1}{N}\sum_{i=1}^{N} V_i^a$$

Then, it is not hard to show that:

$$\mathrm{var}(\bar{V}_N^b) = (1-\rho^2)\mathrm{var}(\bar{V}_N),$$

which shows that the control variate has variance reduction effect, especially when $\rho \to 1$, there should be a deep reduction in variance.

Here we take the same set of parameters as in [40]:

$$C = 4, G = 50, \mathbb{M} = 60, Y = 0.7, r = 0.05, q = 0.02, S_0 = K = 100, T = 0.5.$$

And we price two sets of option with different monitoring dates, one with $d = 6$, which is monthly monitoring; and the other with $d = 26$, which is weekly monitoring. The benchmark prices are 4.00703627 and 3.65349339, which we compute by the method in Chapter 5. Similar as what we did before, given tolerance level ϵ_b for B in theorem 3.2.3, we are able to compute the parameters for CDF grids. The result is listed below in table 3.7.

Then we can use the grids to price corresponding Asian options. In table 3.8, we priced discrete

Asian call in CGMY model by using RQMC as well as control variate. We can see that for both $d = 6$ and $d = 26$, the pricing error is very small. Moreover, the convergence result can be observed for different ϵ_b value. In table 3.9, we set $\epsilon_b =$ 1E-3, and compare different methods, which include: pure Monte Carlo simulation, MC with control variate, RQMC as well as RQMC with control variate. As we can see, using quasi Monte Carlo method itself could already improve the simulation result. And when control variate is incorporated, this new combined method is very powerful, which contributes gain values of several thousands or even tens of thousands.

Asian call in the CGMY model ($d = 6$)

ϵ_b	x_0	x_K	K	$E_{h,M,\mathcal{X}}$	h	M
10^{-2}	-0.213	0.205	44	8.7×10^{-7}	10.899	21
10^{-3}	-0.260	0.244	188	2.0×10^{-8}	9.107	28
10^{-4}	-0.307	0.283	771	4.8×10^{-10}	7.822	35

Asian call in the CGMY model ($d = 26$)

ϵ_b	x_0	x_K	K	$E_{h,M,\mathcal{X}}$	h	M
10^{-2}	-0.129	0.111	45	9.3×10^{-7}	15.748	31
10^{-3}	-0.176	0.150	226	1.8×10^{-8}	12.187	49
10^{-4}	-0.223	0.189	1039	3.8×10^{-10}	9.954	73

Table 3.7: Grids and parameters for CGMY model.

$d = 6$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N ($\times 10^3$)	SE	Error	Price	SE	Error	Price	SE	Error	Price
1	1.5E-3	2.4E-4	4.0068	1.8E-3	2.0E-3	4.0090	2.4E-3	3.7E-4	4.0067
4	4.9E-4	5.9E-4	4.0076	4.7E-4	1.7E-3	4.0088	7.8E-4	1.6E-4	4.0072
16	2.5E-4	7.3E-5	4.0071	3.4E-4	3.6E-4	4.0067	3.8E-4	3.0E-4	4.0073
64	1.2E-4	1.4E-4	4.0072	7.8E-5	1.6E-4	4.0072	1.4E-4	1.9E-4	4.0072
256	3.5E-5	2.6E-4	4.0068	5.9E-5	5.9E-6	4.0070	4.0E-5	7.9E-5	4.0070
1024	3.0E-5	3.0E-4	4.0073	2.5E-5	1.5E-5	4.0070	2.8E-5	3.9E-5	4.0071
4096	1.6E-5	9.7E-4	4.0080	8.6E-6	2.4E-5	4.0071	8.2E-6	1.7E-5	4.0070
16384	4.6E-6	8.7E-4	4.0079	3.9E-6	2.2E-5	4.0071	2.9E-6	3.6E-6	4.0070
65536	1.6E-6	9.4E-4	4.0080	1.9E-6	4.9E-6	4.0070	2.3E-6	1.0E-6	4.0070

$d = 26$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N ($\times 10^3$)	SE	Error	Price	SE	Error	Price	SE	Error	Price
1	2.1E-3	1.8E-3	3.6553	2.5E-3	2.6E-4	3.6537	2.6E-3	5.2E-3	3.6587
4	8.4E-4	1.2E-3	3.6547	8.8E-4	8.3E-4	3.6527	1.1E-3	2.1E-4	3.6537
16	6.3E-4	2.6E-4	3.6537	5.2E-4	1.6E-4	3.6537	4.8E-4	4.1E-4	3.6539
64	1.9E-4	5.5E-5	3.6534	1.9E-4	1.4E-4	3.6534	1.6E-4	8.1E-5	3.6534
256	5.2E-5	9.0E-5	3.6536	8.5E-5	3.4E-4	3.6532	7.0E-5	3.6E-5	3.6535
1024	3.4E-5	6.3E-5	3.6536	2.2E-5	4.5E-5	3.6535	2.2E-5	5.5E-6	3.6535
4096	1.6E-5	1.2E-4	3.6536	1.9E-5	2.4E-5	3.6535	1.7E-5	9.8E-6	3.6535
16384	4.1E-6	1.8E-4	3.6537	9.7E-6	1.2E-5	3.6535	8.7E-6	7.7E-6	3.6535
65536	3.9E-6	5.8E-5	3.6536	4.3E-6	5.5E-6	3.6535	2.6E-6	4.5E-6	3.6535

Table 3.8: Discrete Asian call in the CGMY model: randomized quasi-Monte Carlo ($L = 10$) with control variates.

Monthly monitoring ($d = 6$)

	MC				MC-CV			
N ($\times 10^3$)	SE	Error	CPU	Gain	SE	Error	CPU	Gain
1	1.8E-1	5.8E-2	0.004		3.2E-3	2.4E-3	0.004	3164
4	9.2E-2	8.1E-2	0.006		1.7E-3	2.3E-3	0.006	2929
16	4.6E-2	3.1E-2	0.015		8.7E-4	8.9E-4	0.016	2621
64	2.3E-2	3.8E-2	0.046		4.5E-4	8.4E-5	0.054	2225
256	1.2E-2	2.6E-2	0.173		2.3E-4	1.7E-4	0.204	2308
1024	5.8E-3	1.0E-2	0.687		1.1E-4	9.2E-5	0.805	2373
4096	2.9E-3	4.1E-3	2.757		5.6E-5	4.9E-5	3.215	2300
16384	1.4E-3	4.7E-4	10.906		2.8E-5	1.9E-5	12.764	2136
65536	7.2E-4	8.1E-4	43.234		1.4E-5	2.6E-5	50.907	2246
	RQMC				RQMC-CV			
N ($\times 10^3$)	SE	Error	CPU	Gain	SE	Error	CPU	Gain
1	5.8E-2	7.5E-2	0.004	10	1.8E-3	2.0E-3	0.005	8000
4	2.1E-2	2.9E-2	0.007	16	4.7E-4	1.7E-3	0.009	25544
16	6.7E-3	4.4E-5	0.017	42	3.4E-4	3.6E-4	0.019	14451
64	2.1E-3	2.8E-3	0.054	102	7.8E-5	1.6E-4	0.064	62495
256	5.1E-4	7.3E-5	0.208	460	5.9E-5	5.9E-6	0.241	29695
1024	1.4E-4	4.7E-4	0.864	1365	2.5E-5	1.5E-5	0.959	38558
4096	7.1E-5	3.3E-4	3.288	1399	8.6E-6	2.4E-5	3.835	81747
16384	3.1E-5	4.1E-4	13.017	1709	3.9E-6	2.2E-5	15.302	91843
65536	1.2E-5	3.9E-4	51.935	2997	1.9E-6	4.9E-6	61.222	101409

Weekly monitoring ($d = 26$)

	MC				MC-CV			
N ($\times 10^3$)	SE	Error	CPU	Gain	SE	Error	CPU	Gain
1	1.7E-1	6.1E-2	0.009		3.0E-3	6.3E-4	0.016	1806
4	8.6E-2	8.3E-2	0.021		1.6E-3	1.6E-3	0.022	2758
16	4.2E-2	1.4E-2	0.053		7.8E-4	1.5E-4	0.061	2519
64	2.1E-2	1.3E-3	0.187		4.0E-4	1.6E-4	0.209	2466
256	1.1E-2	3.4E-3	0.736		2.0E-4	2.2E-4	0.825	2699
1024	5.2E-3	2.0E-3	2.993		1.0E-4	4.2E-5	3.279	2468
4096	2.6E-3	4.0E-4	11.776		5.0E-5	1.4E-5	13.032	2443
16384	1.3E-3	1.2E-3	46.944		2.5E-5	8.5E-6	51.760	2452
65536	6.6E-4	1.2E-3	186.393		1.2E-5	2.0E-5	207.167	2722
	RQMC				RQMC-CV			
N ($\times 10^3$)	SE	Error	CPU	Gain	SE	Error	CPU	Gain
1	9.0E-2	5.4E-2	0.011	3	2.5E-3	2.6E-4	0.010	4162
4	3.7E-2	4.5E-3	0.020	6	8.8E-4	8.3E-4	0.021	9551
16	1.2E-2	6.4E-3	0.062	10	5.2E-4	1.6E-4	0.066	5239
64	2.8E-3	2.6E-3	0.219	48	1.9E-4	1.4E-4	0.268	8524
256	1.0E-3	3.9E-4	0.862	103	8.5E-5	3.4E-4	0.974	12655
1024	6.5E-4	5.5E-4	3.464	55	2.2E-5	4.5E-5	3.866	43252
4096	2.5E-4	5.6E-4	13.882	92	1.9E-5	2.4E-5	15.446	14276
16384	1.4E-4	4.5E-5	54.602	74	9.7E-6	1.2E-5	61.938	13613
65536	4.0E-5	2.1E-4	218.003	233	4.3E-6	5.5E-6	244.542	17957

Table 3.9: Discrete Asian call in the CGMY model: standard and quasi-Monte Carlo methods with or without control variates.

Chapter 4

Sensitivity Estimation

Previously, we introduced how to simulate from characteristic function and its application in pricing financial derivatives. In this chapter, we further discuss this method and apply it in sensitivity estimation problem, which has equal importance in financial industry, if not more, as derivatives pricing. In order to apply simulation method, sensitivity is transformed into a probability expectation through likelihood ratio method (LRM). And similar as the case in derivatives pricing, Glasserman and Liu's Laplace transform method in [45] could also apply, yet so does the weakness of their method. Again, through the analysis of the error terms, our method is able to provide an explicit error bound for the total bias in single dimensional problem, by taking advantage of the powerful approximation theory we introduced before. The sensitivity estimation problem is actually more sophisticated than derivative pricing, and some good theoretical result we had before could not be extended to sensitivity estimation. Hence, for multidimensional problem, no simple explicit bound could be provided. However, certain procedure could be set up to guarantee the convergence of bias. Another difference between derivative pricing and sensitivity estimation is that, besides the CDF table used in derivative pricing, one more table need to be generated to store the values of the corresponding partial derivative function, which we will introduce later in detail. Finally, although we illustrate our method in Lévy process models, it could be applied to more general class where the characteristic functions have the analyticity property .

4.1 The likelihood ratio method

4.1.1 LRM estimator and score function

We start with the one-dimensional case. For an option pricing problem, we are interested with evaluating expectation $\mathbb{E}[f(X)]$, where f is a payoff function and X is a random variable following

geometric Lévy process. If we denote the probability density function of X by $p(x)$, then we have:

$$\mathbb{E}[f(X)] = \int f(x)p(x)dx.$$

For sensitivity analysis, we move one step further to deal with the partial derivative of the expectation above. Here we assume that the sensitivity parameter interested only appearing in $p(x)$, denote it as θ. We further assume the partial derivative of $p(x)$ with respect to θ exits and denote it as $\dot{p}_\theta(x)$. For LRM, we assume the following identity holds:

$$\frac{d}{d\theta}\mathbb{E}[f(X)] = \frac{d}{d\theta}\int f(x)p(x)dx = \int f(x)\frac{d}{d\theta}p(x)dx = \int f(x)\dot{p}_\theta(x)dx. \tag{4.1}$$

The key of the above identity is changing order of differentiation and integration, which is not always just to do. However, it can be validated under certain regularity conditions, as shown in Proposition 7.3.5 in [4]. Formula (4.1) can be further developed to:

$$\frac{d}{d\theta}\mathbb{E}[f(X)] = \int f(x)\dot{p}_\theta(x)dx = \int f(x)\frac{\dot{p}_\theta(x)}{p(x)}p(x)dx = \mathbb{E}[f(X)\frac{\dot{p}_\theta(X)}{p(X)}]. \tag{4.2}$$

The expression

$$S_\theta(x) = \dot{p}_\theta(x)/p(x) \tag{4.3}$$

here is called score function. Incorporating this, the LRM estimator is given by $f(X)S_\theta(X)$ and we have:

$$\frac{d}{d\theta}\mathbb{E}[f(X)] = \mathbb{E}[f(X)S_\theta(X)]. \tag{4.4}$$

It is very important to note here that random variable X has probability density function $p(x)$, however, in identity (4.2), we are free to choose any distribution and not restricted to $p(x)$. We will later see that, the choice of the distribution has nothing to do with the bias of simulation, and can only affect the variance.

For the multi-dimensional case, we are interested with the following expectation:

$$\mathbb{E}[h(Y_{t_1}, \cdots, Y_{t_d})], \tag{4.5}$$

where function h depends on the values of a Lévy process $Y = \{Y_t, t \geq 0\}$ on equally spaced time grids $0 = t_0 < t_1 < \cdots < t_d = T$. By the independent and stationary increment property of Lévy

process, $\{Y_{t_{k+1}} - Y_{t_k}, 0 \leq k \leq d-1\}$ are i.i.d random variables. If we define $X_k = Y_{t_{k+1}} - Y_{t_k}, 1 \leq k \leq d$, then the target expectation (4.5) can be rewritten as:

$$\mathbb{E}[f(X_1, \cdots, X_d)]. \tag{4.6}$$

Similarly, we can obtain the LRM identity as:

$$\frac{d}{d\theta}\mathbb{E}[f(X_1, \cdots, X_d)] = \mathbb{E}[f(X_1, \cdots, X_d)\mathbf{S}_\theta(X_1, \cdots, X_d)]. \tag{4.7}$$

The corresponding LRM estimator is hence $f(X_1, \cdots, X_d)\mathbf{S}_\theta(X_1, \cdots, X_d)$. By the independent and stationary property, the join density function is nothing but the product of marginal densities, and the score function is thus a summation with the following expression:

$$\mathbf{S}_\theta(x_1, \cdots, x_d) = \sum_{i=1}^{d} \frac{\dot{p}_\theta(x_i)}{p(x_i)}. \tag{4.8}$$

Remark 4.1.1. *While the independent property is necessary for the above analysis, we only use the stationary property so as to simplify the notation for illustration purpose. Similar result can be easily derived without the stationary property. In that case, $X_1, ..., X_d$ are independent but have different distributions. Denote the probability density function of X_i by $p^{(i)}(x)$. Then we only need to change the formula of $\mathbf{S}_\theta(x_1, \cdots, x_d)$ a little to get similar result, where*

$$\mathbf{S}_\theta(x_1, \cdots, x_d) = \sum_{i=1}^{d} \frac{\dot{p}^{(i)}_\theta(x_i)}{p^{(i)}(x_i)}. \tag{4.9}$$

4.1.2 The inverse transform method and approximation to LRM estimator

Now let us first discuss the one-dimensional case. We have already reformed the sensitivity analysis problem into the problem of calculating the LRM estimator. The transform we did in (4.2) is just like importance sampling technique. And this gives us a lot flexibilities in choosing the distribution of X. Although the choice of the distribution will not affect the bias, we tend to choose a distribution that is similar to X in some respects, in order to make the variance small.

Denote the cumulative distribution function of random variable X by $F(x)$. And our choice of the distribution is a piecewise linear CDF function $\hat{F}(x)$, which is a very good approximation of $F(x)$. And hence they are very "similar". We denote the corresponding random variable by $\hat{X}$.

Adopting the notation from previous chapter, we take a large enough interval $\mathcal{X} = [x_0, x_K]$ on real axis, then divide the interval into K equally spaced subintervals each with length $\eta = (x_K - x_0)/K$. Denote the nodes of $\mathcal{X}$ by x_k, then $x_k = x_0 + k \cdot \eta$ with $0 \leq k \leq K$. Denote $F_k = F(x_k)$ for $0 < k < K$. Assume $\hat{F}_k$ is an approximation of F_k, store the pairs $(x_k, \hat{F}_k)$ in the following table:

$$\begin{pmatrix} x_0 & x_1 & \cdots & x_K \\ \hat{F}_0 & \hat{F}_1 & \cdots & \hat{F}_K \end{pmatrix}. \tag{4.10}$$

Then we choose to simulate from the following function $\hat{F}(x)$, which is a piecewise linear function very similar to $F(x)$:

$$\hat{F}(x) = \begin{cases} 0, & x \leq x_0 \\ \hat{F}_{k-1} + \frac{\hat{F}_k - \hat{F}_{k-1}}{\eta}(x - x_{k-1}), & x_{k-1} < x < x_k, \quad 1 \leq k \leq K \\ 1, & x \geq x_K \end{cases}. \tag{4.11}$$

Here we made a sloppy notation at those joint points x_k. The reason is that how we define the function values at those points does not matter at all. When simulates from a continuous distribution, one gets those points with zero probability. Now $\hat{X}$ can be generated through inverse transform method:

$$\hat{X} = \begin{cases} x_0, & 0 \leq U < \hat{F}_0 \\ x_{k-1} + \frac{x_k - x_{k-1}}{\hat{F}_k - \hat{F}_{k-1}}(U - \hat{F}_{k-1}), & \hat{F}_{k-1} \leq U < \hat{F}_k, \quad 1 \leq k \leq K \\ x_K, & \hat{F}_K < U \leq 1 \end{cases}. \tag{4.12}$$

Once we have $\hat{X}$, we still need to estimate values of the score function (4.3). To do this, we define $\dot{F}_\theta = \partial F / \partial \theta$, and we have the following identity:

$$\dot{p}_\theta = \frac{\partial p}{\partial \theta} = \frac{\partial^2 F}{\partial x \partial \theta} = \frac{\partial \dot{F}_\theta}{\partial x}.$$

Similarly, we can find a piecewise linear approximation $\hat{\dot{F}}_\theta(x)$ for $\dot{F}_\theta(x)$. We choose an interval $\dot{\mathcal{X}} = [\dot{x}_0, \dot{x}_K]$, divide it into $\dot{K}$ equally spaced subintervals each with length $\dot{\eta}$. Denote the nodes on the interval as $\dot{x}_k$ and denote $\dot{F}_k = \dot{F}_\theta(\dot{x}_k)$ for $0 \leq k \leq \dot{K}$. Assuming $\hat{\dot{F}}_k$ is an approximation of $\dot{F}_k$, we can also generate a similar table as (4.10) :

$$\begin{pmatrix} \dot{x}_0 & \dot{x}_1 & \cdots & \dot{x}_{\dot{K}} \\ \hat{\dot{F}}_0 & \hat{\dot{F}}_1 & \cdots & \hat{\dot{F}}_{\dot{K}} \end{pmatrix}. \tag{4.13}$$

Slightly different from the definition of $\hat{F}(x)$, we define $\hat{\dot{F}}_\theta(x)$ by:

$$\hat{\dot{F}}_\theta(x) = \begin{cases} \hat{\dot{F}}_{k-1} + \frac{\hat{\dot{F}}_k - \hat{\dot{F}}_{k-1}}{\dot{\eta}}(x - \dot{x}_{k-1}), & \dot{x}_{k-1} \le x < \dot{x}_k, \quad 1 \le k \le \dot{K} \\ 0, & \text{otherwise.} \end{cases} \tag{4.14}$$

Further, if we denote the partial derivative of $\hat{F}(x)$ and $\hat{\dot{F}}_\theta(x)$ w.r.t variable x by $\hat{p}(x)$ and $\hat{\dot{p}}_\theta(x)$ respectively, then we can obtain the following equations:

$$\hat{\dot{p}}_\theta(x) = \begin{cases} (\hat{\dot{F}}_\theta(\dot{x}_k) - \hat{\dot{F}}_\theta(\dot{x}_{k-1}))/\dot{\eta}, & \text{if } x \in (\dot{x}_{k-1}, \dot{x}_k) \\ 0, & \text{otherwise,} \end{cases} \tag{4.15}$$

$$\hat{p}(x) = \frac{1}{\eta}\sum_{k=1}^{K}(\hat{F}_k - \hat{F}_{k-1}) \cdot \mathbf{1}_{(x_{k-1}, x_k)}(x), \tag{4.16}$$

where $\mathbf{1}_A(x)$ is the indicator function takes value one if $x \in A$ and zero otherwise.

Then, we can approximate the score function $S_\theta(x) = \dot{p}(x)/p(x)$ by $\hat{S}_\theta(x) = \hat{\dot{p}}(x)/\hat{p}(x)$ and approximate our target expectation $\mathbb{E}_X[f(X)S_\theta(X)]$ by $\mathbb{E}_{\hat{X}}[f(\hat{X})\hat{S}_\theta(\hat{X})]$. Notice that we use the lower index for these two expectations to address the fact that they are for different random variables. Then the approximation error or the bias, is thus given by:

$$\begin{aligned} \left|\mathbb{E}_X[f(X)S_\theta(X)] - \mathbb{E}_{\hat{X}}[f(\hat{X})\hat{S}_\theta(\hat{X})]\right| &= \Big| \int f(x)S_\theta(x)p(x)dx - \int f(x)\hat{S}_\theta(x)\hat{p}(x)dx \Big| \\ &= \Big| \int f(x)\dot{p}(x)dx - \int f(x)\hat{\dot{p}}(x)dx \Big|. \end{aligned} \tag{4.17}$$

We leave the bias analysis in detail to Section 4.2. And this formula makes it clear that the choice of $\hat{X}$ has nothing to do with the estimation bias.

To compute $\mathbb{E}_{\hat{X}}[f(\hat{X})\hat{S}_\theta(\hat{X})]$, we only need to generate N samples of $\{\hat{X}^n : 1 \le n \le N\}$ and estimate the above expectation by the following:

$$\frac{1}{N}\sum_{n=1}^{N} f(\hat{X}^n)\hat{S}_\theta(\hat{X}^n).$$

Similarly, for multi-dimensional case, we approximate $\mathbb{E}_{\mathbf{X}}[f(X_1, ..., X_d)\mathbf{S}_\theta(X_1, ..., X_d)]$ in (4.7) by

$$\mathbb{E}_{\hat{\mathbf{X}}}[f(\hat{X}_1, ..., \hat{X}_d)\hat{\mathbf{S}}_\theta(\hat{X}_1, ..., \hat{X}_d)], \tag{4.18}$$

where $\mathbf{X} = (X_1, ..., X_d)$ and $\hat{\mathbf{X}} = (\hat{X}_1, ..., \hat{X}_d)$ are d-dimensional random variables. And we use the

following approximation for $\mathbf{S}_\theta(x_1, \cdots, x_d)$ in (4.9):

$$\hat{\mathbf{S}}_\theta(x_1, \cdots, x_d) = \sum_{i=1}^{d} \frac{\hat{p}_\theta^{(i)}(x_i)}{\hat{p}^{(i)}(x_i)}. \tag{4.19}$$

We first generate $\hat{\mathbf{X}}^n = (\hat{X}_1^n, ..., \hat{X}_d^n)$, $1 \le n \le N$, then compute $\hat{\mathbf{S}}_\theta(\hat{X}_1^n, ..., \hat{X}_d^n)$, and finally use the following to approximate the expectation.

$$\mathbb{E}_{\hat{\mathbf{X}}}[f(\hat{X}_1^n, ..., \hat{X}_d^n)\hat{\mathbf{S}}_\theta(\hat{X}_1^n, ..., \hat{X}_d^n)] \approx \frac{1}{N}\sum_{n=1}^{N} f(\hat{X}_1^n, ..., \hat{X}_d^n)\hat{\mathbf{S}}_\theta(\hat{X}_1^n, ..., \hat{X}_d^n).$$

For the multi-dimensional case, the grids we use might be different in each dimension. But in most applications, those are the same. In order to make the theory more general, at this moment, we assume we would obtain different grids and denote the corresponding parameters by x_0^i, $x_{K_i}^i$, η_i for $1 \le i \le d$.

Remark 4.1.2. *In this part, we set different grids for approximating $F(x)$ and $\dot{F}(x)$ to show they could be different theoretically. While in practice, we may set the grids the same and combine two tables into one to save computational cost, as shown in [19].*

4.1.3 Hilbert transform

In section 4.1.2, we use $\hat{F}_k$ and $\hat{\dot{F}}_k$ to denote approximation of F_k and $\dot{F}_k$ respectively. Function $\dot{F}(x)$ can be approximated by the following Hilbert transform representation:

$$\dot{F}_{\dot{h},\dot{M}}(x) = \frac{1}{2}\dot{\phi}_\theta(0) + \frac{i}{2}\sum_{m=-\dot{M}}^{\dot{M}} e^{-ix(m-1/2)\dot{h}} \frac{\dot{\phi}_\theta((m-1/2)\dot{h})}{(m-1/2)\pi}, \quad \dot{h} > 0, \dot{M} \ge 1. \tag{4.20}$$

If we assume the following property holds for some constants $\dot{\kappa}, \dot{c}, \dot{\nu} > 0$,

$$|\dot{\phi}(\xi)| \le \dot{\kappa}\exp(-\dot{c}|\xi|^{\dot{\nu}}), \quad \xi \in \mathbb{R}, \tag{4.21}$$

which is true in many applications, then we will obtain exponentially decaying errors. Now we summarize it in the following theorem without proof:

Theorem 4.1.3. *Let $\dot{F}_\theta$ and $\dot{p}_\theta$ be defined as previously, then suppose that $\dot{\phi}_\theta \in H(\mathcal{D}_{(\dot{d}_-,\dot{d}_+)})$. we have*

$$\dot{F}_\theta(x) = \frac{1}{2}\dot{\phi}_\theta(0) - \frac{i}{2}\mathcal{H}(e^{-i\xi x}\dot{\phi}_\theta(\xi))(0).$$

For any $a \in (\dot{d}_-, 0)$,

$$\dot{\phi}(0) - \dot{F}_\theta(x) = e^{ax} \int_{\mathbb{R}} \frac{e^{-ix\xi}\dot{\phi}(\xi + ia)}{2\pi i(\xi + ia)} d\xi.$$

For any $a \in (0, \dot{d}_+)$,

$$\dot{F}_\theta(x) = e^{ax} \int_{\mathbb{R}} \frac{e^{-ix\xi}\dot{\phi}(\xi + ia)}{-2\pi i(\xi + ia)} d\xi.$$

If $|\dot{\phi}| \le \dot{\kappa} \exp(-\dot{c}|\xi|^{\dot{\nu}})$ *for some* $\dot{\kappa}, \dot{c}, \dot{\nu} > 0$, *then for any* $\dot{h} > 0$ *and integer* $\dot{M} \ge 1$,

$$\begin{aligned} |\dot{F}_\theta(x) - \dot{F}_{\dot{h},\dot{M}}(x)| \le & \frac{e^{-2\pi|\dot{d}_-|/\dot{h} + x\dot{d}_-}}{2\pi|\dot{d}_-|(1 - e^{-2\pi|\dot{d}_-|/\dot{h}})} \|\dot{\phi}_\theta\|^- + \frac{e^{-2\pi\dot{d}_+/\dot{h} + x\dot{d}_+}}{2\pi\dot{d}_+(1 - e^{-2\pi\dot{d}_+/\dot{h}})} \|\dot{\phi}_\theta\|^+ \\ & + \frac{\dot{\kappa}}{2\pi}\left(\frac{1}{\dot{M}} + \frac{2}{\dot{\nu}\dot{c}(\dot{M}\dot{h})^{\dot{\nu}}}\right) \exp -\dot{c}(\dot{M}\dot{h})^{\dot{\nu}}. \end{aligned} \tag{4.22}$$

If $|\dot{\phi}| \le \dot{\kappa}|\xi|^{-\dot{\nu}}$ *for some* $\dot{\kappa}, \dot{\nu} > 0$, *then for any* $\dot{h} > 0$ *and integer* $\dot{M} \ge 1$,

$$|\dot{F}_\theta(x) - \dot{F}_{\dot{h},\dot{M}}(x)| \le \frac{e^{-2\pi|\dot{d}_-|/\dot{h} + x\dot{d}_-}}{2\pi|\dot{d}_-|(1 - e^{-2\pi|\dot{d}_-|/\dot{h}})} \|\dot{\phi}_\theta\|^- + \frac{e^{-2\pi\dot{d}_+/\dot{h} + x\dot{d}_+}}{2\pi\dot{d}_+(1 - e^{-2\pi\dot{d}_+/\dot{h}})} \|\dot{\phi}_\theta\|^+ + \frac{\dot{\kappa}}{\pi}\left(\frac{1}{\dot{M}} + \frac{2}{\dot{\nu}}\right)\frac{1}{(\dot{M}\dot{h})^{\dot{\nu}}}.$$

Remark 4.1.4. *The proof of the theorem is straightforward following the proof of Theorem 3.1 in [20]. One thing different from the original proof that worth mention is the following identity:*

$$\begin{aligned} \dot{F}_\theta(x) &= \int_{-\infty}^{x} \dot{p}_\theta(y)dy = \int_{\mathbb{R}} \dot{p}_\theta(y)\mathbf{1}_{(-\infty,x)}(y)dy \\ &= \mathcal{F}(\mathbf{1}_{(-\infty,x)} \cdot \dot{p}_\theta)(0) = \frac{1}{2}\dot{\phi}_\theta(0) - \frac{i}{2}\mathcal{H}(e^{-i\xi x}\dot{\phi}_\theta(\xi))(0). \end{aligned}$$

Also, in this proof, we need to justify

$$\dot{p}_\theta(x) = \frac{1}{2\pi} \int_{\mathbb{R}} e^{-izx}\dot{\phi}_\theta(z)dz = \mathcal{F}^{-1}(\dot{\phi}_\theta)$$

which is basically changing order of differentiation and integration. It is easy to validate given the following assumption: for all θ *in a small region* V, ϕ *is continuous differentiable w.r.t* θ, *and there is an integrable function* H *for which* $|\dot{\phi}_\theta(z)| < H(z)$ *for all real* z. *And this assumption is naturally satisfied in our case.*

4.2 Bias analysis

In this section, we analyze the bias incurred by using $\mathbb{E}_{\hat{X}}[f(\hat{X})\hat{S}_\theta(\hat{X})]$ to approximate our target expectation $\mathbb{E}_X[f(X)S_\theta(X)]$.

4.2.1 The general case

Now we discuss the estimation bias in (4.17) in detail. Denote

$$|\dot{\mathcal{X}}| = \dot{x}_K - \dot{x}_0,$$

$$||f||_{\dot{\mathcal{X}}} = \sup_{x\in\dot{\mathcal{X}}} |f(x)|, \quad ||f'||_{\dot{\mathcal{X}}} = \operatorname{ess\,sup}_{x\in\dot{\mathcal{X}}} |f'(x)|,$$

$$||\dot{p}'_\theta||_{\dot{\mathcal{X}}} = \operatorname{ess\,sup}_{x\in\dot{\mathcal{X}}} |\dot{p}'_\theta(x)|,$$

$$\dot{E}_{\dot{\mathcal{X}}} = \sup_{x\in\dot{\mathcal{X}}} |\dot{F}(x) - \hat{\dot{F}}(x)|.$$

Then we have the following upper bound for the estimation bias.

Theorem 4.2.1. *Consider a continuous r.v.X with cdf $F(x)$ and pdf $p(x)$. Let $\{\dot{x}_0, ..., \dot{x}_{\dot{K}}\}$ be a uniform grid with step size $\dot{\eta} = (\dot{x}_{\dot{K}} - \dot{x}_0)/\dot{K}$ for some positive $\dot{K}$. Let $\hat{X}$ be a random variable with probability density function $\hat{p}(x)$ defined as in (4.16). $\dot{p}_\theta$, the partial derivative of $p(x)$, is bounded on $\dot{\mathcal{X}}$, differentiable in $\bigcup_{i=1}^{\dot{K}}(\dot{x}_{i-1}, \dot{x}_i)$, and $\|\dot{p}'_\theta\|_{\dot{\mathcal{X}}} < \infty$. $\hat{\dot{p}}_\theta(x)$, $S_\theta(x)$, $\hat{S}_\theta(x)$ are as defined previously . Suppose f is differentiable in $\bigcup_{i=1}^{\dot{K}}(\dot{x}_{i-1}, \dot{x}_i)$ except at n^f points, and $\|f'\|_{\dot{\mathcal{X}}} < \infty$, $\|f\|_{\dot{\mathcal{X}}} < \infty$. Then*

$$\begin{aligned}\left|\mathbb{E}[f(X)\cdot S_\theta(X)] - \mathbb{E}[f(\hat{X})\cdot \hat{S}_\theta(\hat{X})]\right| \le (&\int_{-\infty}^{\dot{x}_0} + \int_{\dot{x}_K}^{\infty})|f(x)|\cdot|\dot{p}_\theta(x)|dx \\ &+ \frac{1}{\dot{K}^2}\|f'\|_{\dot{\mathcal{X}}}\cdot\|\dot{p}'_\theta\|_{\dot{\mathcal{X}}}|\dot{\mathcal{X}}|^3 \\ &+ 2((\dot{K}+n^f)\|f\|_{\dot{\mathcal{X}}} + \|f'\|_{\dot{\mathcal{X}}}\cdot|\dot{\mathcal{X}}|)\dot{E}_{\dot{\mathcal{X}}}.\end{aligned} \tag{4.23}$$

Proof. The bias is given by

$$\begin{aligned}\mathbb{E}[f(X)\cdot S_\theta(X)] - \mathbb{E}[f(\hat{X})\cdot \hat{S}_\theta(\hat{X})] &= \int_{\mathbb{R}} f(x)\frac{\dot{p}_\theta(x)}{p(x)}\cdot p(x)dx - \int_{\mathbb{R}} f(x)\frac{\hat{\dot{p}}_\theta(x)}{\hat{p}(x)}\cdot\hat{p}(x)dx \\ &= \int_{\mathbb{R}} f(x)\cdot\dot{p}_\theta(x)dx - \int_{\mathbb{R}} f(x)\cdot\hat{\dot{p}}_\theta(x)dx \\ &= \int_{\mathbb{R}} f(x)\cdot\dot{p}_\theta(x)dx - \int_{\dot{x}_0}^{\dot{x}_K} f(x)\cdot\hat{\dot{p}}_\theta(x)dx \\ &= \sum_{k=1}^{\dot{K}}\int_{\dot{x}_{k-1}}^{\dot{x}_k} f(x)\cdot(\dot{p}_\theta(x) - \hat{\dot{p}}_\theta(x))dx \\ &+ (\int_{-\infty}^{\dot{x}_0} + \int_{\dot{x}_K}^{\infty})f(x)\cdot\dot{p}_\theta(x)dx.\end{aligned} \tag{4.24}$$

Assume $f(x)$ is differentiable in $(\dot{x}_{k-1}, \dot{x}_k)$, then

$$
\begin{aligned}
&|\int_{\dot{x}_{k-1}}^{\dot{x}_k} f(x)\cdot(\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x))dx| \\
&= \big|\int_{\dot{x}_{k-1}}^{\dot{x}_k} \Big(f(\dot{x}_{k-1}+)+f'\big(\xi_k(x)\big)(x-\dot{x}_{k-1})\Big)\cdot(\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x))dx\big| \\
&\le \|f\|_{\dot{\mathcal{X}}}\cdot|\dot{F}_k-\dot{F}_{k-1}-(\hat{\dot{F}}_k-\hat{\dot{F}}_{k-1})|+\|f'\|_{\dot{\mathcal{X}}}\cdot\dot{\eta}\cdot\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x)|dx \\
&\le 2\|f\|_{\dot{\mathcal{X}}}\cdot\dot{E}_{\dot{\mathcal{X}}}+\|f'\|_{\dot{\mathcal{X}}}\cdot\dot{\eta}\cdot\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x)|dx.
\end{aligned}
$$

In general, if $f(x)$ is not differentiable at n_k^f points in $(\dot{x}_{k-1}, \dot{x}_k)$, where $\sum_{k=1}^K n_k^f = n^f$, it can be shown in the same way that

$$
|\int_{\dot{x}_{k-1}}^{\dot{x}_k} f(x)\cdot(\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x))dx| \le 2(n_k^f+1)\|f\|_{\dot{\mathcal{X}}}\cdot\dot{E}_{\dot{\mathcal{X}}}+\|f'\|_{\dot{\mathcal{X}}}\cdot\dot{\eta}\cdot\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x)|dx.
$$

And since

$$
\begin{aligned}
&\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x)|dx \\
&=\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\frac{\dot{F}_k-\dot{F}_{k-1}}{\dot{\eta}}+\frac{\dot{F}_k-\dot{F}_{k-1}}{\dot{\eta}}-\frac{\hat{\dot{F}}_k-\hat{\dot{F}}_{k-1}}{\dot{\eta}}|dx \\
&=|\dot{F}_k-\dot{F}_{k-1}-(\hat{\dot{F}}_k-\hat{\dot{F}}_{k-1})|+\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\frac{\dot{F}_k-\dot{F}_{k-1}}{\dot{\eta}}|dx \\
&\le 2\dot{E}_{\dot{\mathcal{X}}}+\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\frac{\dot{F}_k-\dot{F}_{k-1}}{\dot{\eta}}|dx.
\end{aligned}
$$

Then,

$$
\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\frac{\dot{F}_k-\dot{F}_{k-1}}{\dot{\eta}}|dx=\frac{1}{\dot{\eta}}\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\int_{\dot{x}_{k-1}}^{\dot{x}_k}(\dot{p}_\theta(x)-\dot{p}_\theta(y))dy|dx.
$$

Assume $\dot{p}_\theta(x)$ is differentiable in $(\dot{x}_{k-1}, \dot{x}_k)$, then since $\dot{p}_\theta(y)=\dot{p}_\theta(x)+\dot{p}'(\xi_k(x,y))(y-x)$, we have

$$
\int_{\dot{x}_{k-1}}^{\dot{x}_k}|\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x)|dx \le 2\dot{E}_{\dot{\mathcal{X}}}+\|\dot{p}'_\theta\|_{\dot{\mathcal{X}}}\cdot\dot{\eta}^2.
$$

Then

$$
\big|\sum_{k=1}^{\dot{K}}\int_{\dot{x}_{k-1}}^{\dot{x}_k} f(x)\cdot(\dot{p}_\theta(x)-\hat{\dot{p}}_\theta(x))dx\big| \le 2\dot{E}_{\dot{\mathcal{X}}}((\dot{K}+n^f)\|f\|_{\dot{\mathcal{X}}}+\|f'\|_{\dot{\mathcal{X}}}(\dot{x}_{\dot{K}}-\dot{x}_0))+\|f'\|_{\dot{\mathcal{X}}}\|\dot{p}'_\theta\|_{\dot{\mathcal{X}}}(\dot{x}_K-\dot{x}_0)\cdot\dot{\eta}^2.
$$

And

$$\begin{aligned}\left|\mathbb{E}[f(X)\cdot S_\theta(X)]-\mathbb{E}[f(\hat{X})\cdot\hat{S}_\theta(\hat{X})]\right| \le (&\int_{-\infty}^{\dot{x}_0}+\int_{\dot{x}_{\dot{K}}}^{\infty})|f(x)|\cdot|\dot{p}_\theta(x)|dx\\ &+\frac{1}{\dot{K}^2}\|f'\|_{\dot{\mathcal{X}}}\cdot\|\dot{p}'_\theta\|_{\dot{\mathcal{X}}}|\dot{\mathcal{X}}|^3\\ &+2((\dot{K}+n^f)\|f\|_{\dot{\mathcal{X}}}+\|f'\|_{\dot{\mathcal{X}}}\cdot|\dot{\mathcal{X}}|)\dot{E}_{\dot{\mathcal{X}}}.\end{aligned}$$

□

4.2.2 Class with analyticity property

We use the same notations as we did previously by denoting the c.f. of X as ϕ and denoting the partial derivative of ϕ w.r.t θ as $\dot{\phi}$. Then $\dot{\phi}$ is actually the Fourier transform function of $\dot{F}(x)$ under some mild condition. In the following, we assume that $\dot{\phi}\in H(\mathcal{D}_{(\dot{d}_-,\dot{d}_+)})$. And values of the partial derivative functions are computed through the Hilbert transform representation by (4.20), that is, $\hat{\dot{F}}_k=\dot{F}_{\dot{h},\dot{M}}(\dot{x}_k)$. We further define

$$E_{\dot{h},\dot{M},\dot{\mathcal{X}}}=\sup_{x\in\dot{\mathcal{X}}}|\dot{F}(x)-\dot{F}_{\dot{h},\dot{M}}(x)|.$$

Then we can update our results of estimation bias from previous chapter:

Theorem 4.2.2. *Besides the assumptions given in Theorem 4.2.1, we further assume that $\dot{\phi}_\theta\in H(\mathcal{D}_{\dot{d}_-,\dot{d}_+})$ for some $-\infty<\dot{d}_-<0<\dot{d}_+<\infty$, and $\dot{\phi}_\theta$ satisfies (4.21). Moreover, for all θ in a small region V, ϕ is continuous differentiable w.r.t θ, and there is an integrable function H for which $|\dot{\phi}_\theta(z)|<H(z)$ for all real z, then*

$$\begin{aligned}\left|\mathbb{E}[f(X)\cdot S_\theta(X)]-\mathbb{E}[f(\hat{X})\cdot\hat{S}_\theta(\hat{X})]\right| \le{}& \frac{\|\dot{\phi}_\theta\|^+}{2\pi}\int_{-\infty}^{\dot{x}_0}|f(x)|e^{x\dot{d}_+}dx+\frac{\|\dot{\phi}_\theta\|^-}{2\pi}\int_{\dot{x}_{\dot{K}}}^{\infty}|f(x)|e^{x\dot{d}_-}dx\\ &+\frac{1}{2\pi\dot{K}^2}\|f'\|_{\dot{\mathcal{X}}}\cdot|\dot{\mathcal{X}}|^3\int_{\mathbb{R}}|\xi\dot{\phi}_\theta(\xi)|d\xi\\ &+2((\dot{K}+n^f)\|f\|_{\dot{\mathcal{X}}}+\|f'\|_{\dot{\mathcal{X}}}\cdot|\dot{\mathcal{X}}|)\dot{E}_{h,M,\dot{\mathcal{X}}}.\end{aligned}\tag{4.25}$$

for all $\theta\in V$.

Proof. The probability density function $p(x)$ of X admits the following inverse Fourier transform representation:

$$p(x)=\frac{1}{2\pi}\int_{\mathbb{R}}e^{-izx}\phi(z)dz.$$

Then, give $\phi(z)$ is continuous differentiable w.r.t θ in V. By dominated convergence theorem, we

have:

$$\dot{p}_\theta(x) = \frac{\partial p(x)}{\partial \theta} = \frac{1}{2\pi}\int_{\mathbb{R}} e^{-izx}\frac{\partial \phi(z)}{\partial \theta}dz = \frac{1}{2\pi}\int_{\mathbb{R}} e^{-izx}\dot{\phi}_\theta(z)dz.$$

By Cauchy's integral theorem, for any $\epsilon > 0$ such that $\dot{d}_+ - \epsilon > 0$, we have

$$\dot{p}_\theta(x) = \frac{1}{2\pi}\int_{-\infty+i(\dot{d}_+-\epsilon)}^{+\infty+i(\dot{d}_+-\epsilon)} e^{-izx}\dot{\phi}_\theta(z)dz = e^{(\dot{d}_+-\epsilon)x}\int_{\mathbb{R}}\frac{1}{2\pi}e^{-ix\xi}\dot{\phi}_\theta(\xi + i(\dot{d}_+ - \epsilon))d\xi.$$

Then

$$\int_{-\infty}^{\dot{x}_0}|f(x)||\dot{p}_\theta(x)|dx \leq \frac{\|\dot{\phi}_\theta\|^+}{2\pi}\int_{-\infty}^{\dot{x}_0}|f(x)|e^{x\dot{d}_+}dx.$$

Similarly,

$$\int_{\dot{x}_{\dot{K}}}^{\infty}|f(x)||\dot{p}_\theta(x)|dx \leq \frac{\|\dot{\phi}_\theta\|^-}{2\pi}\int_{\dot{x}_{\dot{K}}}^{\infty}|f(x)|e^{x\dot{d}_-}dx.$$

Since $\xi\dot{\phi}_\theta(\xi)$ is absolutely integrable on $\mathbb{R}$ given the assumptions on $\dot{\phi}_\theta$,

$$\dot{p}'_\theta = \frac{1}{2\pi}\frac{d}{dx}\int_{\mathbb{R}} e^{-i\xi x}\dot{\phi}_\theta(\xi)d\xi = \frac{1}{2\pi i}\int_{\mathbb{R}} e^{-i\xi x}\xi\dot{\phi}_\theta(\xi)d\xi,$$

where the interchange of the integration and differentiation is valid again by the dominated convergence theorem. Therefore,

$$\|\dot{p}'_\theta\|_{\dot{\mathcal{X}}} \leq \frac{1}{2\pi}\int_{\mathbb{R}}|\xi\dot{\phi}_\theta(\xi)|d\xi.$$

Combine the previous results, we are done with the proof. □

4.2.3 The multidimensional case

Multidimensional case in sensitivity analysis is much harder. We approximate

$$\mathbb{E}_{\mathbf{X}}[f(X_1, ..., X_d)\mathbf{S}_\theta(X_1, ..., X_d)]$$

in (4.7) by

$$\mathbb{E}_{\hat{\mathbf{X}}}[f(\hat{X}_1, ..., \hat{X}_d)\hat{\mathbf{S}}_\theta(\hat{X}_1, ..., \hat{X}_d)].$$

We are trying to find a bound for the difference between the two terms. Although we can follow the idea as in the proof of theorem 3.2.5, the computational process is very complex. In fact, the

estimation bounds can be decomposed into d^2 terms, each of them has the following format:

$$\int_{\mathbb{R}} f(x_1,...,x_d)p^{(i)}(x_i)dx_i - \int_{x_0^i}^{x_{K_i}^i} f(x_1,...,x_d)\hat{p}^{(i)}(x_i)dx_i \tag{4.26}$$

$$\int_{\mathbb{R}} f(x_1,...,x_d)\dot{p}_\theta^{(i)}(x_i)dx_i - \int_{x_0^i}^{x_{K_i}^i} f(x_1,...,x_d)\hat{\dot{p}}_\theta^{(i)}(x_i)dx_i \tag{4.27}$$

We can then take advantage of theorem 4.2.2 and corresponding theorem in previous chapter. If we further assume the following bounds

$$|f(x_1,...,x_d)| \le g_1(x_1)\cdot g_d(x_d), \quad |\frac{d}{dx_i}f(x_1,...,x_d)| \le g_1(x_1)\cdot g_d(x_d),$$

are satisfied for all $1 \le i \le d$, then we only need to bound the following two terms as shown in [19]:

$$\begin{aligned} B_i^p &= \frac{||\phi^{(i)}||^-}{2\pi}\int_{x_{K_i}^{(i)}}^{\infty} g_i(x)e^{xd_-^{(i)}}dx + \frac{||\phi^{(i)}||^+}{2\pi}\int_{-\infty}^{x_0^{(i)}} g_i(x)e^{xd_+^{(i)}}dx \\ &\quad + \frac{1}{2\pi K_i^2}||g_i||_{\mathcal{X}_i}|\mathcal{X}_i|^3\int_{\mathbb{R}}|\xi\phi^{(i)}(\xi)|d\xi \\ &\quad + 2\left(K_i + n^f + |\mathcal{X}_i|\right)||g_i||_{\mathcal{X}_i}\cdot E_{h_i,M_i,\mathcal{X}_i}. \end{aligned}$$

$$\begin{aligned} B_i^{\dot{p}} &= \frac{\|\dot{\phi}_\theta\|^+}{2\pi}\int_{-\infty}^{\dot{x}_0} g_i(x)e^{x\dot{d}_+}dx + \frac{\|\dot{\phi}_\theta\|^-}{2\pi}\int_{\dot{x}_{\dot{K}}}^{\infty} g_i(x)e^{x\dot{d}_-}dx \\ &\quad + \frac{1}{2\pi\dot{K}^2}\|g_i\|_{\dot{\mathcal{X}}}\cdot|\dot{\mathcal{X}}|^3\int_{\mathbb{R}}|\xi\dot{\phi}_\theta(\xi)|d\xi \\ &\quad + 2((\dot{K} + n^f + |\dot{\mathcal{X}}|)\|g_i\|_{\dot{\mathcal{X}}}\dot{E}_{h,M,\dot{\mathcal{X}}}. \end{aligned}$$

Then we can control a tolerance level ϵ_b for $B = \max\{B_i^p, B_i^{\dot{p}}\}$ for $1 \le i \le d$. In practice, we adjust the grids $\{\dot{x}_0^{(i)},\cdots,\dot{x}_{\dot{K}_i}^{(i)}\}$, $\{x_0^{(i)},\cdots,x_{K_i}^{(i)}\}$ and other parameters as well to decrease B sequentially. And by doing this several times, we are able to control the bias to our desired level.

4.3 Numerical results

In this section, we verify the theoretical results in Section 4.2, and illustrate how to determine the grid (4.13) and the numerical parameters for Hilbert transform method. We then provide sensitivity analysis for both European and Asian option in CGMY model [14] and illustrate the efficiency of our method. Computations are done on a Lenovo laptop T61p with Intel Core 2 Duo 2.5GHz CPU

and 3GB RAM.

4.3.1 The CGMY process

A CGMY process X_t is a pure jump Lévy process with drift μ and the following Lévy density

$$\frac{Ce^{Gx}}{|x|^{1+Y}}\mathbf{1}_{\{x<0\}} + \frac{Ce^{-\mathbb{M}x}}{|x|^{1+Y}}\mathbf{1}_{\{x>0\}}$$

for some $C > 0, G > 0, \mathbb{M} > 0, 0 < Y < 2$. The martingale condition requires $\mu = r - q - C\Gamma(-Y)((\mathbb{M}-1)^Y - \mathbb{M}^Y + (G+1)^Y - G^Y)$. Explicit expressions for the density and the cdf of X_t are not available. However, the characteristic function of X_t is known explicitly:

$$\phi_t(\xi) = \exp(i\mu t\xi - tC\Gamma(-Y)(\mathbb{M}^Y - (\mathbb{M}-i\xi)^Y + G^Y - (G+i\xi)^Y)),$$

where $\Gamma(\cdot)$ is gamma function. ϕ is in $H(\mathcal{D}_{(d_-,d_+)})$ with $d_- = -\mathbb{M}, d_+ = G$. When $0 < Y < 1$, ϕ satisfies (3.7) with $\kappa = \exp(-tC\Gamma(-Y)(\mathbb{M}^Y + G^Y))$, $c = 2tC|\Gamma(-Y)\cos(\pi Y/2)|$, and $\nu = Y$.

When taking partial derivatives of $\phi_t(\xi)$, the derivative function $\dot{\phi}_\theta(\xi) = \phi_t(\xi)P(\xi)$, where $P(\xi)$ is a power function of ξ with order no more than $Y + 2$. Hence, the exponential tail of $\phi_t(\xi)$ dominates. Please refer to [19] for more detail. Also, $\phi(\xi)$ and $\dot{\phi}_\theta(\xi)$ have the same analytic strip. Hence, the same analysis can be adopted without much change as in [20].

4.3.2 Sensitivity Analysis of CGMY model

In this section, we consider the sensitivity analysis for CGMY model. The parameters are the same as those in [40]:

$$C = 4, G = 50, \mathbb{M} = 60, Y = 0.7, r = 0.05, q = 0.02, S_0 = K_s = 100, T = 0.5.$$

For the sensitivity of European call option, we take $\dot{x}_0 = \ln(K_s/S_0) = 0$ and determine $\dot{x}_{\dot{K}} > 0$ according to the second term in (4.25):

$$\frac{||\dot{\phi}_\theta||^-}{2\pi}\int_{\dot{x}_{\dot{K}}}^{\infty}|f(x)|e^{x\dot{d}_-}dx = \frac{||\dot{\phi}_\theta||^-}{2\pi}e^{\dot{x}_{\dot{K}}\dot{d}_-}\Big(\frac{1}{\dot{d}_-} - \frac{1}{\dot{d}_- + 1}e^{\dot{x}_{\dot{K}}}\Big).$$

For three tolerance levels of total bias $\epsilon_b = 10^{-2}, 10^{-3}, 10^{-4}$, we find the smallest $\dot{x}_K$ so that the above is bounded by $\epsilon_b/2$. With the root finding approach, we obtain $\dot{x}_K$ as reported in Table 4.1.

Sensitivity of European call in the CGMY model w.r.t C

ϵ_b	$\dot{x}_0$	$\dot{x}_{\dot{K}}$	$\dot{K}$	$\dot{E}_{\dot{h},\dot{M},\dot{\mathcal{X}}}$	$\dot{h}$	$\dot{M}$
10^{-2}	0	1.011	49	5.6×10^{-7}	2.882	78
10^{-3}	0	1.050	166	1.6×10^{-8}	2.744	83
10^{-4}	0	1.089	566	4.5×10^{-10}	2.617	88

Sensitivity of European call in the CGMY model w.r.t G

ϵ_b	$\dot{x}_0$	$\dot{x}_{\dot{K}}$	$\dot{K}$	$\dot{E}_{\dot{h},\dot{M},\dot{\mathcal{X}}}$	$\dot{h}$	$\dot{M}$
10^{-2}	0	0.943	11	2.5×10^{-6}	3.022	73
10^{-3}	0	0.982	35	8.1×10^{-8}	2.874	78
10^{-4}	0	1.022	119	2.3×10^{-9}	2.735	83

Sensitivity of European call in the CGMY model w.r.t $\mathbb{M}$

ϵ_b	$\dot{x}_0$	$\dot{x}_{\dot{K}}$	$\dot{K}$	$\dot{E}_{\dot{h},\dot{M},\dot{\mathcal{X}}}$	$\dot{h}$	$\dot{M}$
10^{-2}	0	1.005	10	2.4×10^{-6}	3.053	72
10^{-3}	0	1.044	33	7.7×10^{-8}	2.902	77
10^{-4}	0	1.084	111	2.3×10^{-9}	2.761	82

Sensitivity of European call in the CGMY model w.r.t Y

ϵ_b	$\dot{x}_0$	$\dot{x}_{\dot{K}}$	$\dot{K}$	$\dot{E}_{\dot{h},\dot{M},\dot{\mathcal{X}}}$	$\dot{h}$	$\dot{M}$
10^{-2}	0	1.055	217	2.4×10^{-6}	2.726	85
10^{-3}	0	1.094	738	7.7×10^{-8}	2.601	91
10^{-4}	0	1.134	2513	2.3×10^{-9}	2.486	96

Asian call in the CGMY model ($d = 6$) w.r.t C

ϵ_b	x_0	x_K	K	$E_{h,M,\mathcal{X}}$	h	M
10^{-2}	-0.213	0.205	44	8.7×10^{-7}	10.899	21
10^{-3}	-0.260	0.244	188	2.0×10^{-8}	9.107	28
10^{-4}	-0.307	0.283	771	4.8×10^{-10}	7.822	35
ϵ_b	$\dot{x}_0$	$\dot{x}_{\dot{K}}$	$\dot{K}$	$E_{\dot{h},\dot{M},\dot{\mathcal{X}}}$	$\dot{h}$	$\dot{M}$
10^{-2}	-0.211	0.207	23	1.7×10^{-6}	10.890	32
10^{-3}	-0.258	0.246	99	3.8×10^{-8}	9.101	41
10^{-4}	-0.305	0.285	404	9.1×10^{-10}	7.818	51

Asian call in the CGMY model ($d = 26$) w.r.t C

ϵ_b	x_0	x_K	K	$E_{h,M,\mathcal{X}}$	h	M
10^{-2}	-0.129	0.111	45	9.3×10^{-7}	15.748	31
10^{-3}	-0.176	0.150	226	1.8×10^{-8}	12.187	49
10^{-4}	-0.223	0.189	1039	3.8×10^{-10}	9.954	73
ϵ_b	$\dot{x}_0$	$\dot{x}_{\dot{K}}$	$\dot{K}$	$E_{\dot{h},\dot{M},\dot{\mathcal{X}}}$	$\dot{h}$	$\dot{M}$
10^{-2}	-0.190	0.141	17	2.5×10^{-6}	18.636	164
10^{-3}	-0.144	0.126	98	4.4×10^{-8}	13.777	269
10^{-4}	-0.191	0.165	478	8.6×10^{-10}	10.969	398

Table 4.1: Grids and parameters for Fourier transform inversion.

We then select the step size $(\dot{x}_K - \dot{x}_0)/\dot{K}$ of the grid according to the third term in (4.25):

$$\frac{1}{2\pi\dot{K}^2}||f'||_{\dot{\mathcal{X}}}|\dot{\mathcal{X}}|^3\int_{\mathbb{R}}|\xi\dot{\phi}_\theta(\xi)|d\xi.$$

The smallest positive integer $\dot{K}$ so that the above is bounded by $\epsilon_b/2$ is given by:

$$\dot{K} = \Big(\frac{1}{\pi\epsilon_b}||f'||_{\dot{\mathcal{X}}}|\dot{\mathcal{X}}|^3\int_{\mathbb{R}}|\xi\dot{\phi}_\theta(\xi)|d\xi\Big)^{1/2}.$$

It is easy to find that $||f'||_{\mathcal{X}} = e^{\dot{x}_K}$. The corresponding values for $\dot{K}$ are reported in Table 4.1. We then control the last term in (4.25):

$$(2\dot{K}||f||_{\dot{\mathcal{X}}} + 2||f'||_{\dot{\mathcal{X}}}(\dot{x}_{\dot{K}} - \dot{x}_0))\dot{E}_{h,M,\dot{\mathcal{X}}} = (2\dot{K}(e^{\dot{x}_K} - 1) + 2(\dot{x}_{\dot{K}} - \dot{x}_0)e^{\dot{x}_K})\dot{E}_{h,M,\dot{\mathcal{X}}}.$$

We make the above term equal to $0.01\epsilon_b$, which is negligible when compared to ϵ_b. The corresponding values for $E_{\dot{h},\dot{M},\dot{\mathcal{X}}}$ can be found in Table 4.1. According to (4.22), since $\dot{x}_0 = 0$, we have

$$\begin{aligned} E_{\dot{h},\dot{M},\dot{\mathcal{X}}} \quad \leq \quad & \frac{e^{-2\pi|\dot{d}_-|/\dot{h}}}{2\pi|\dot{d}_-|(1-e^{-2\pi|\dot{d}_-|/\dot{h}})}||\dot{\phi}_\theta||^- + \frac{e^{-2\pi\dot{d}_+/\dot{h}+\dot{x}_{\dot{K}}\dot{d}_+}}{2\pi\dot{d}_+(1-e^{-2\pi\dot{d}_+/\dot{h}})}||\dot{\phi}_\theta||^+ \\ & + \frac{\kappa}{2\pi}\Big(\frac{1}{\dot{M}} + \frac{2}{\nu c(\dot{M}\dot{h})^\nu}\Big)e^{-c(\dot{M}\dot{h})^\nu}. \end{aligned} \tag{4.28}$$

We then solve for $\dot{h}$ and $\dot{M}$ so that the summation of the first two terms above is equal to one half of the value in the fifth column of Table 4.1. And so is the third term above. The resulting $\dot{h}$ and $\dot{M}$ are reported in Table 4.1. For the choice of grids used for simulating $\hat{X}$, we take the same values used in [20].

With the settings in Table 4.1, we estimate the sensitivity of European vanilla call option. The benchmark value is computed using the Hilbert transform method of [38] combined with finite difference. The results are shown in Table 4.3. "N" refers to the sample size in thousands. "SE" represents the standard error. "Error" represents the actual estimation error. "CPU" represents the computational time in seconds. The pre-simulation time for computing the matrix is around 0.01 second for $\epsilon_b = 10^{-2}$, and around 0.02 second for $\epsilon_b = 10^{-4}$, which is very fast. It can be observed that when the sample size N is large enough, the bias dominates. And this phenomenon is rather obvious for $\mathbb{M}$ and G's case, since their SE is smaller than the case of C. Take $\theta = G$ as an example, when $\epsilon_b = 10^{-3}$, while N is decreasing, the absolute error decreases at first but keeps stable around 2E-4 and does not decrease anymore. On the other hand, SE is decreasing to the level of 1E-5,

N ($\times 10^3$)	RMSE	Error	CPU(s)	Delta
1	4.1E-2	2.9E-2	0.017	0.5373
4	2.1E-2	6.2E-3	0.018	0.5606
16	1.0E-2	1.6E-3	0.020	0.5652
64	5.1E-3	6.7E-3	0.025	0.5601
256	2.6E-3	1.7E-3	0.043	0.5651
1024	1.3E-3	1.9E-3	0.123	0.5649
4096	6.4E-4	1.4E-3	0.455	0.5654
16384	3.2E-4	8.8E-4	1.770	0.5659
65536	1.6E-4	6.5E-4	6.926	0.5661

Table 4.2: European call option delta in the CGMY model. $\epsilon_b = 0.001$. Exact value: 0.566793

which implies the absolute error is dominated by the bias, which should be about 2E-4. Notice that the bias is smaller than the target level we set for, which is $S_0 e^{-rT}\epsilon_b$. The same reasoning remains for $\epsilon_b = 10^{-2}$, the bias is around 1.5E-3 in this case. But for $\epsilon_b = 10^{-4}$, it is hard to tell whether the absolute error dominates by bias or SE or even both, since SE is around 1E-5, about the same level as its absolute error.

Now we consider the sensitivity of fixed strike Asian option with respect to parameter C. We use the same parameter as used for European call option. The average price is based on either monthly monitoring ($d = 6$) or weekly monitoring ($d = 26$). Benchmark prices are computed using the Fourier transform method [8; 17] combined with finite difference. For $d = 6, 26$ and $\epsilon_b = 10^{-2}, 10^{-3}, 10^{-4}$ (the tolerance level for B in section 4.2.3), we determine the grids $\mathcal{X}$, $\dot{\mathcal{X}}$ and the numerical parameters for Fourier transform inversion . The results are also reported in Table 4.1. With this settings, we compute the sensitivity w.r.t C and report the result in Table 4.4. "N" refers to the sample size in thousands. "SE" represents the standard error. "Error" represents the actual estimation error. "SV" represents the sensitivity values. We can see that, for both $d = 6$ and $d = 26$, as ϵ_b decrease, the absolute error decrease accordingly. It can be observed that, with the same bound ϵ_b, the absolute error for $d = 6$ is actually smaller than for $d = 26$. This agrees with the rule of dimension curse: as the dimension increase, the problem becomes harder to solve.

In table 4.5, we report the sensitivity values of different options w.r.t parameter C . We can observe that, European option is more sensitive than Asian option, and Asian option with less monitoring dates is more sensitive than Asian option with more monitoring dates. This result agrees with the fact that Asian options price would suffer less from financial market changes when comparing with European options, which is more volatile.

$\theta = C$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N ($\times 10^3$)	SE	Error	CPU	SE	Error	CPU	SE	Error	CPU
1	1.4E-1	1.0E-1	0.011	1.2E-1	1.0E-1	0.013	1.3E-1	2.4E-2	0.027
4	5.8E-2	1.0E-1	0.012	6.5E-2	5.1E-2	0.014	6.3E-2	4.1E-2	0.027
16	3.2E-2	1.6E-2	0.013	3.6E-2	2.7E-2	0.016	3.3E-2	3.7E-2	0.029
64	1.7E-2	1.3E-2	0.020	1.7E-2	5.6E-3	0.024	1.7E-2	1.6E-2	0.039
256	8.7E-3	1.3E-3	0.045	8.4E-3	1.2E-2	0.052	8.7E-3	8.7E-3	0.078
1024	4.4E-3	1.1E-3	0.160	4.3E-3	4.9E-3	0.229	4.3E-3	6.8E-4	0.232
4096	2.2E-3	1.6E-3	0.599	2.1E-3	2.5E-3	0.650	2.1E-3	2.1E-4	0.831
16384	1.1E-3	3.1E-3	2.384	1.1E-3	1.2E-4	2.589	1.1E-3	6.5E-5	3.289
65536	5.4E-4	1.3E-3	9.420	5.4E-4	1.4E-4	10.237	5.4E-4	3.1E-4	12.903
262144	2.7E-4	1.1E-3	37.720	2.7E-4	2.0E-4	40.741	2.7E-4	2.3E-5	51.635

$\theta = G$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N ($\times 10^3$)	SE	Error	CPU	SE	Error	CPU	SE	Error	CPU
1	6.6E-3	1.1E-3	0.009	8.9E-3	1.6E-2	0.009	6.4E-3	2.5E-3	0.012
4	3.6E-3	1.3E-3	0.010	3.5E-3	7.9E-4	0.010	3.2E-3	4.4E-3	0.013
16	1.9E-3	7.1E-5	0.011	1.7E-3	1.0E-3	0.012	1.6E-3	9.2E-4	0.015
64	9.8E-4	1.8E-3	0.018	8.5E-4	4.8E-4	0.020	8.3E-4	6.8E-4	0.022
256	5.0E-4	1.5E-3	0.043	4.2E-4	2.0E-4	0.048	4.1E-4	1.3E-4	0.050
1024	2.5E-4	9.6E-4	0.143	2.1E-4	4.6E-4	0.157	2.1E-4	4.8E-5	0.182
4096	1.3E-4	1.4E-3	0.613	1.1E-4	6.0E-5	0.584	1.0E-4	2.0E-5	0.701
16384	6.2E-5	1.6E-3	2.192	5.3E-5	1.7E-4	2.334	5.2E-5	3.2E-5	2.474
65536	3.1E-5	1.5E-3	8.726	2.6E-5	1.2E-4	9.208	2.6E-5	9.7E-6	9.877
262144	1.6E-5	1.5E-3	34.970	1.3E-5	1.3E-4	37.049	1.3E-5	1.3E-5	39.515

$\theta = M$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N ($\times 10^3$)	SE	Error	CPU	SE	Error	CPU	SE	Error	CPU
1	6.0E-3	4.1E-3	0.009	7.7E-3	6.1E-2	0.009	5.1E-3	5.7E-3	0.015
4	3.7E-3	8.9E-4	0.010	3.7E-3	7.4E-4	0.011	2.7E-3	5.2E-3	0.016
16	1.8E-3	1.5E-3	0.011	1.5E-3	2.0E-3	0.012	1.5E-3	1.1E-3	0.017
64	9.6E-4	1.4E-3	0.018	7.8E-4	5.5E-4	0.019	8.0E-4	3.2E-5	0.027
256	4.9E-4	1.8E-3	0.043	4.1E-4	3.7E-4	0.046	4.2E-4	1.1E-4	0.063
1024	2.7E-4	1.4E-3	0.140	2.1E-4	1.9E-4	0.175	2.0E-4	2.9E-4	0.189
4096	1.4E-4	1.4E-3	0.558	1.0E-4	8.5E-5	0.625	1.0E-4	7.5E-5	0.633
16384	6.7E-5	1.4E-3	2.159	5.2E-5	3.4E-5	2.301	5.1E-5	5.4E-5	2.432
65536	3.3E-5	1.4E-3	8.608	2.6E-5	1.3E-4	9.331	2.5E-5	2.7E-6	9.779
262144	1.7E-5	1.4E-3	34.570	1.3E-5	1.3E-4	36.737	1.3E-5	9.5E-6	38.95

Table 4.3: European Vanilla call in the CGMY model, derivative of option price w.r.t θ

$d = 6$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N ($\times 10^3$)	SE	Error	SV	SE	Error	SV	SE	Error	SV
1	1.3E-1	2.3E-1	0.672	1.1E-1	2.9E-2	0.474	1.1E-1	8.5E-2	0.360
4	6.1E-2	4.8E-2	0.493	6.1E-2	9.9E-2	0.544	6.4E-2	6.4E-2	0.509
16	3.0E-2	4.3E-3	0.449	2.9E-2	6.0E-4	0.444	3.0E-2	1.1E-2	0.456
64	1.5E-2	3.5E-2	0.410	1.5E-2	7.2E-3	0.452	1.5E-2	2.6E-3	0.442
256	7.2E-3	4.4E-2	0.401	7.3E-3	1.4E-2	0.431	7.3E-3	2.5E-4	0.445
1024	3.6E-3	4.9E-2	0.396	3.6E-3	5.6E-3	0.439	3.7E-3	8.8E-4	0.444
4096	1.8E-3	4.5E-2	0.400	1.8E-3	7.5E-3	0.438	1.8E-3	7.2E-4	0.446
16384	9.0E-4	4.4E-2	0.402	9.1E-4	6.9E-3	0.438	9.2E-4	9.1E-4	0.444
65536	4.5E-4	4.4E-2	0.401	4.6E-4	7.0E-3	0.438	4.6E-4	8.1E-4	0.444

$d = 26$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N ($\times 10^3$)	SE	Error	SV	SE	Error	SV	SE	Error	SV
1	1.7E-1	2.1E-1	0.194	1.8E-1	2.0E-2	0.427	1.7E-1	7.4E-2	0.481
4	8.4E-2	4.0E-1	0.010	9.2E-2	2.5E-2	0.382	9.0E-2	9.8E-2	0.506
16	4.1E-2	3.7E-1	0.034	4.6E-2	1.2E-2	0.419	4.4E-2	1.6E-2	0.391
64	2.1E-2	3.7E-1	0.036	2.2E-2	3.0E-2	0.377	2.2E-2	2.7E-2	0.380
256	1.0E-2	3.6E-1	0.048	1.1E-2	3.6E-2	0.372	1.1E-2	3.0E-3	0.404
1024	5.3E-3	3.6E-1	0.048	5.5E-3	2.8E-2	0.379	5.6E-3	2.1E-3	0.405
4096	2.6E-3	3.6E-1	0.048	2.8E-3	3.1E-2	0.376	2.8E-3	2.5E-4	0.407
16384	1.3E-3	3.6E-1	0.050	1.4E-3	3.2E-2	0.376	1.4E-3	1.1E-3	0.406
65536	6.6E-4	3.6E-1	0.051	6.9E-4	3.3E-2	0.375	7.0E-4	2.1E-3	0.405

Table 4.4: Sensitivity of Discrete Asian call w.r.t C

Option	European Call	Asian Call $d = 6$	Asian Call $d = 26$
Sensitivity w.r.t C	0.685	0.444	0.400

Table 4.5: Sensitivity of Discrete Asian call w.r.t C

$\theta = Y$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N $(\times 10^3)$	SE	Error	CPU	SE	Error	CPU	SE	Error	CPU
1	2.1E-0	5.7E-2	0.015	2.4E-0	3.3E-1	0.033	2.8E-0	4.0E-0	0.096
4	1.4E-0	1.7E-0	0.016	1.1E-0	9.0E-1	0.033	1.2E-0	5.3E-1	0.098
16	6.1E-1	1.9E-3	0.018	5.6E-1	2.6E-1	0.036	5.6E-1	2.7E-1	0.102
64	2.9E-1	1.5E-1	0.026	2.8E-1	3.9E-1	0.046	2.9E-1	4.7E-1	0.122
256	1.5E-1	1.6E-1	0.147	1.4E-1	2.9E-1	0.086	1.4E-1	1.5E-1	0.199
1024	7.2E-2	3.7E-2	0.178	7.2E-2	6.1E-2	0.259	7.2E-2	7.6E-2	0.499
4096	3.6E-2	4.6E-3	0.671	3.6E-2	3.3E-2	0.919	3.6E-2	3.8E-3	1.719
16384	1.8E-2	4.5E-3	2.638	1.8E-2	5.6E-3	3.631	1.8E-2	1.4E-2	6.656
65536	9.0E-3	4.0E-3	10.581	9.0E-3	7.1E-3	14.277	9.0E-3	6.9E-3	26.257
262144	4.5E-3	4.6E-3	44.608	4.5E-3	2.2E-3	56.997	4.5E-3	1.4E-4	105.141

Table 4.6: European Vanilla call in the CGMY model, derivative of option price w.r.t $\theta = Y$

$d = 6$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N $(\times 10^3)$	SE	Error	SV	SE	Error	SV	SE	Error	SV
1	4.3E-3	1.1E-2	-0.0138	5.1E-3	2.3E-3	-0.0230	5.5E-3	1.5E-3	-0.0268
4	2.1E-3	1.3E-2	-0.0119	3.1E-3	6.1E-4	-0.0259	2.7E-3	2.7E-4	-0.0256
16	1.1E-3	1.2E-2	-0.0133	1.4E-3	2.7E-3	-0.0226	1.3E-3	8.1E-4	-0.0245
64	5.6E-4	1.2E-2	-0.0129	6.7E-4	3.4E-3	-0.0219	6.5E-4	1.6E-3	-0.0237
256	2.8E-4	1.2E-2	-0.0129	3.3E-4	4.1E-3	-0.0212	3.4E-4	3.1E-4	-0.0250
1024	1.4E-4	1.3E-2	-0.0127	1.6E-4	3.2E-3	-0.0221	1.7E-4	3.6E-4	-0.0249
4096	6.8E-5	1.2E-2	-0.0128	8.1E-5	3.7E-3	-0.0216	8.3E-5	7.2E-4	-0.0246
16384	3.4E-5	1.2E-2	-0.0128	4.1E-5	3.6E-3	-0.0217	4.1E-5	6.7E-4	-0.0246
65536	1.7E-5	1.3E-2	-0.0128	2.0E-5	3.6E-3	-0.0217	2.1E-5	6.1E-4	-0.0247

$d = 26$

	$\epsilon_b = 10^{-2}$			$\epsilon_b = 10^{-3}$			$\epsilon_b = 10^{-4}$		
N $(\times 10^3)$	SE	Error	SV	SE	Error	SV	SE	Error	SV
1	8.8E-3	2.2E-1	0.1955	8.5E-3	3.2E-2	0.0084	8.6E-3	1.3E-3	-0.0244
4	4.3E-3	2.2E-1	0.1932	4.1E-3	3.3E-2	0.0098	4.3E-3	1.8E-3	-0.0213
16	2.1E-3	2.2E-1	0.1953	2.0E-3	3.1E-2	0.0075	2.1E-3	4.7E-3	-0.0185
64	1.1E-3	2.2E-1	0.1947	9.9E-4	3.0E-2	0.0068	1.0E-3	3.0E-3	-0.0201
256	5.4E-4	2.2E-1	0.1936	4.9E-4	3.1E-2	0.0075	5.3E-4	2.6E-3	-0.0206
1024	2.7E-4	2.2E-1	0.1932	2.4E-4	3.1E-2	0.0074	2.6E-4	2.5E-3	-0.0207
4096	1.3E-4	2.2E-1	0.1932	1.2E-4	3.0E-2	0.0071	1.3E-4	2.5E-3	-0.0206
16384	6.7E-5	2.2E-1	0.1931	6.1E-5	3.1E-2	0.0074	6.6E-5	2.6E-3	-0.0205
65536	3.4E-5	2.2E-1	0.1932	3.1E-5	3.0E-2	0.0073	3.3E-5	2.6E-3	-0.0206

Table 4.7: Sensitivity of Discrete Asian call w.r.t G

4.3.3 European and Asian delta

Delta is one of the most Greeks used in financial industry for hedging and pricing purpose. It is define as the sensitivity of the option price with respect to the underlying asset price. To simulate the delta, we need to rewrite the characteristic function of CGMY model into the following form:

$$\phi_t(\xi) = \exp(i\xi(\ln(S_0) + \mu t) - tC\Gamma(-Y)(\mathbb{M}^Y - (\mathbb{M} - i\xi)^Y + G^Y - (G + i\xi)^Y)),$$

Then, the partial derivative is given by

$$\dot{\phi}_t(\xi) = \frac{i\xi}{S_0}\phi_t(\xi).$$

For European delta, we only need to design one set of grid since it is single dimensional. For Asian option, although it has d dimension, S_0 only appears in X_1. Hence, we only need to compute two set of grids instead of d sets since X_2, ..., X_d are i.i.d..

We can calculate the parameters for the grids then and also obtain the delta, which is listed in table 4.2.

Chapter 5

Discrete Asian Options in Jump Diffusion models

In this chapter, a new scheme is proposed to price fixed strike Asian options. Although not as heavily traded as European and American options, Asian option still has a large share in natural resource derivative market, like crude oil and heating oil derivative market. However, due to the difficulty inherited from its structure, very few efficient methods have been developed so far to price Asian options in Lévy process models, especially for discretely monitored arithmetic average Asian options. On the one hand, the simulation method we mentioned in the first part of the thesis could be adopted. On the other hand, in this chapter, we propose a new numerical scheme based on transform method. For the new scheme, exponential convergence of its numerical error could be achieved both theoretically and practically.

5.1 Discrete Asian Options

5.1.1 Basic Algorithm

We first introduce the method proposed by Carverhill and Clewlow in [17]. We assume the underlying asset price S_t follows a geometric Lévy process:

$$S_t = S_0 e^{X_t}.$$

In this case, the Asian call option price with strike price K is given by:

$$\mathbb{E}[e^{-rT}(A_T - K)^+],$$

where A_T is the average asset price on $d+1$ monitoring dates $\{t_0, t_1, ..., t_d\}$ ($t_0 = 0$ and $t_d = T$) and has the following mathematical expression:

$$A_T = \frac{1}{d+1} \sum_{n=0}^{d} S_{t_n}.$$

Note that in some reference, the average is on d asset prices excluding S_{t_0} or S_0. Although the option price varies for different definition, all pricing methods introduced here should be able to adapt to both situations.

If we further define the Lévy increment by $\Delta X_n = X_{t_n} - X_{t_{n-1}}$, for $n = 1, 2..., d$, and assume $X_0 = 0$. Then, we have the following expression for A_T through the Steward-Hodges factorization:

$$\begin{aligned}
A_T &= \frac{1}{d+1} \sum_{n=0}^{d} S_{t_n} = \frac{1}{d+1}(S_0 + S_0 e^{X_1} + S_0 e^{X_2} + ... + S_0 e^{X_d}) \\
&= \frac{S_0}{d+1}(1 + e^{\Delta X_1} + e^{\Delta X_1 + \Delta X_2} + ... + e^{\Delta X_1 + ... + \Delta X_{d-1}} + e^{\Delta X_1 + ... + \Delta X_d}) \\
&= \frac{S_0}{d+1}(1 + e^{\Delta X_1}(1 + e^{\Delta X_2}(1 + ... + e^{\Delta X_{d-1}}(1 + e^{\Delta X_d})))) \\
&= \frac{S_0}{d+1}(1 + \exp(\Delta X_1 + \ln(1 + \exp(\Delta X_2 + \ln(1 + ... + \exp(\Delta X_{d-1} + \ln(1 + \exp(\Delta X_d))))))))).
\end{aligned}$$

Moreover, in order to simplify notation, we adopt a new set of random variable:

$$\begin{aligned}
B_d &= \Delta X_d, \\
B_n &= \Delta X_n + \ln(1 + e^{B_{n+1}}), \quad n = d-1, ..., 1.
\end{aligned} \tag{5.1}$$

Then, A_T will have a very simple expression:

$$A_T = \frac{S_0}{d+1}(1 + e^{B_1}).$$

Beginning with B_d, a Lévy increment, B_n could be calculated recursively for $n = d-1, ..., 1$. After B_1 is obtained, we are able to calculate A_T and then the option price. To be more specific, if we denote the probability density functions of ΔX_n, B_n and $\ln(1 + e^{B_n})$ by p_n, f_n and g_n respectively, then we will have the following relationship for any n in $\{1, ..., d\}$:

$$g_n(x) = f_n(\ln(e^x - 1)) \frac{e^x}{e^x - 1}, \quad x > 0, \tag{5.2}$$

as well as

$$f_n(x) = (p_n \star g_{n+1})(x).$$

where $\star$ is the convolution operator. Notice that since ΔX_n, $n = d, ..., 1$ are Lévy increments, which are independent with each other, we then have ΔX_n and $\ln(1 + e^{B_{n+1}})$ are independent. And hence as a summation of the two random variables, density function of B_n could be calculated as the convolution of these two individual probability density functions.

To summarize, we obtain the following algorithm, which was first studied by Carverhill and Clewlow in [17].

Algorithm 5.1.1. *1. $f_d(x) = p_d(x)$ is defined as the density function of the Lévy increment;*

2. For $n = d - 1, ..., 1$, calculate g_{n+1} from (5.2) and then compute $f_n = p_n \star g_{n+1}$,

3. Finally, the discretely monitored Asian call option price is given by

$$c = e^{-rT} \int_{\mathbb{R}} \big(\frac{S_0}{d+1}(1 + e^x) - K\big)^+ f_1(x)dx.$$

5.1.2 Recentered Algorithm

Benhamou in [8] claims that f_n will shift right as n increasing, since for every step in the recursion, a Lévy increment ΔX_n is added, which usually comes with positive mean. And this will lead to a much larger integration interval if numerical computation is applied. Also, he proposes an adjustment to the original algorithm (5.1.1) to alleviate this problem by a simple re-centering. Assume the mean estimation of B_n is denoted by b_n. For the first one, $b_d = \mathbb{E}[\Delta X_d]$ can be easily calculated. For the rest b_ns, an intuitive way is to calculate recursively through:

$$b_n = \mathbb{E}[\Delta X_n] + \ln(1 + e^{b_{n+1}}), \quad n = d - 1, ..., 1.$$

Then, they consider the re-centered process $A_n = B_n - b_n$, for $n = d, ..., 1$. And now, an updated version of (5.1) becomes

$$\begin{aligned} A_d &= \Delta X_d - b_d, \\ A_n &= \Delta X_n + \ln(1 + e^{A_{n+1}+b_{n+1}}) - b_n, \quad n = d - 1, ..., 1, \end{aligned}$$

and the average asset price now have the following representation:

$$A_T = \frac{S_0}{d+1}(1 + e^{A_1+b_1}).$$

Follow the same argument used in the original algorithm, if we denote the density of ΔX_n, A_n and $\ln(1 + e^{A_n+b_n}) - b_{n-1}$ by p_n, f_n and g_n respectively, then we will have the following identity:

$$g_n(x) = f_n(\ln(e^{x+b_{n-1}} - 1) - b_n)\frac{e^{x+b_{n-1}}}{e^{x+b_{n-1}} - 1}, \quad x > -b_{n-1}. \tag{5.3}$$

And also, the following convolution equation still holds for $n = d, ..., 1$,

$$f_n(x) = (p_n \star g_{n+1})(x).$$

To summarize, Benhamou's algorithm as in [8] is shown as below:

Algorithm 5.1.2. *1. $f_d(x) = p_d(x + b_d)$ is defined as a shifted density function of Lévy increment;*

2. For $n = d - 1, ..., 1$, calculate g_{n+1} from (5.3) and then compute $f_n = p_n \star g_{n+1}$;

3. Finally, the discretely monitored Asian call option price is given by

$$c = e^{-rT} \int_{\mathbb{R}} \big(\frac{S_0}{d+1}(1 + e^{x+b_1}) - K\big)^+ f_1(x)dx.$$

5.2 A New Scheme

The main drawback of the previous two schemes is that they both suffer from slow convergence rate. For example, the rate is only quadratic if Trapezoidal's rule is applied. While different numerical quadratures could be applied to obtain better convergence rate, it is still restricted to polynomial order, as the case in [42] by Fusai and Meucci. Even worse, the numerical result they obtained can hardly achieve an accuracy of 1E-5, which is discovered by Cerny and Kyriakou and shown in [68]. Meanwhile, Cerny and Kyriakou provided a better scheme in [68]. However, their method is still restricted by polynomial convergence. Here, we propose a new numerical scheme. By taking advantage of the analytic property and exponential tail behavior of characteristic functions of Lévy process, exponential convergence could be achieved in numerical integration. Also, the numerical

results shows that an accuracy of up to 10 digits after the decimal point could be achieved for most popular Lévy process models.

5.2.1 Modified Algorithm

We denote the characteristic function of the Lévy increment by $\phi(\xi)$. Then, $\phi(\xi)$ is the Fourier transform of the density $p_n(x)$:

$$\mathcal{F}(p_n) = \phi(\xi).$$

We further denote the Fourier transform of g_n and f_n by $\hat{g}_n$ and $\hat{f}_n$ respectively, where $g_n(x)$ and $f_n(x)$ are defined as in previous section. Then, take Fourier transform on both side of

$$f_n(x) = (p_n \star g_{n+1})(x).$$

According to the convolution theorem, we obtain:

$$\mathcal{F}(f_n) = \mathcal{F}(p_n \star g_{n+1}) = \mathcal{F}(p_n) \cdot \mathcal{F}(g_{n+1}),$$

or

$$\hat{f}_n(\xi) = \phi(\xi) \cdot \hat{g}_{n+1}(\xi).$$

Instead of managing all the computations within the state space as the previous two algorithms, we move around between the state space and Fourier space in the new algorithm to take advantage of properties of characteristic function so as to obtain exponential convergence result. To the same consideration, we make some modification in the last step of the previous algorithm. It is noticeable that the integrand $(\frac{S_0}{d+1}(1 + e^{x+b_1}) - K)^+$ has a discontinuous first order derivative. In this case, Trapezoidal's rule usually lead to a quadratic convergence. However, by taking advantage of the Hilbert transform method proposed in [38], we are able to achieve exponential convergence result. We state the scheme here and leave the theoretical detail to the next section. First, the integrand can be divided into two parts:

$$c = e^{-rT} \int_{\mathbb{R}} (\frac{S_0}{d+1}(1+e^{x+b_1})-K)^+ f_1(x)dx = e^{-rT} \frac{S_0}{d+1} \int_{z_0}^{\infty} (1+e^{x+b_1}) f_1(x)dx - e^{-rT} K \int_{z_0}^{\infty} f_1(x)dx.$$

where $z_0 = \ln((d+1)K/S_0 - 1) - b_1$. Further, we have:

$$\begin{aligned}\int_{z_0}^{\infty} f_1(x)dx &= \int_{-\infty}^{\infty} f_1(x)\mathbf{1}_{\{z_0,\infty\}}dx = \int_{-\infty}^{\infty} e^{i\cdot 0\cdot x} f_1(x)\mathbf{1}_{\{z_0,\infty\}}dx \\ &= \mathcal{F}\{f_1(x)\cdot \mathbf{1}_{\{z_0,\infty\}}\}(0) = \frac{1}{2}\hat{f}_1(0) + \frac{i}{2}e^{i\cdot 0\cdot z_0}\mathcal{H}(e^{-i\cdot\xi\cdot z_0}\hat{f}_1(\xi))(0).\end{aligned}$$

Here $\mathcal{H}(\cdot)$ represents the Hilbert transform. Similarly, if we define $h_1(x) = e^x f_1(x)$, then

$$\int_{z_0}^{\infty} e^x f_1(x)dx = \frac{1}{2}\hat{h}_1(0) + \frac{i}{2}e^{i\cdot 0\cdot z_0}\mathcal{H}(e^{-i\cdot\xi\cdot z_0}\hat{h}_1(\xi))(0),$$

where

$$\begin{aligned}\hat{h}_1(\xi) &= \int_{-\infty}^{\infty} e^{i\xi x}e^x f_1(x)dx = \int_{-\infty}^{\infty} e^{i\cdot(\xi-i)\cdot x} f_1(x)dx \\ &= \mathcal{F}\{f_1(x)\}(\xi - i) = \hat{f}_1(\xi - i) = \phi(\xi - i)\cdot\hat{g}_2(\xi - i).\end{aligned}$$

Hence, when $n = 2$ in the recursion, besides values of $\hat{g}_2(\xi)$, we need to calculate the function values $\hat{g}_2$ at $\xi - i$. This will be more clear when later we discuss the discrete approximation. Now, we summarize the new algorithm as below:

Algorithm 5.2.1. *(An alternative scheme)*

1. $\hat{f}_d = \phi(\xi)$.

2. *For* $n = d, ..., 2$,

$$f_n(x) = \mathcal{F}^{-1}(\hat{f}_n(\xi)). \tag{5.4a}$$

$$\hat{g}_n(\xi) = \int_{-b_{n-1}}^{\infty} e^{i\xi y}g_n(y)dy = \int_{-\infty}^{\infty} e^{i\xi y(x)}f_n(x)dx. \tag{5.4b}$$

$$\hat{f}_{n-1}(\xi) = \phi(\xi)\cdot\hat{g}_n(\xi). \tag{5.4c}$$

   where $y(x) = \ln(e^{x+b_n} + 1) - b_{n-1}$ *for each recursive step.*

3. *Compute* $\hat{g}_2(\xi - i)$ *when* $n = 2$*, and use Hilbert transform to compute the Asian call option price*

$$c = e^{-rT}\int_{\mathbb{R}}(\frac{S_0}{d+1}(1 + e^{x+b_1}) - K)^+ f_1(x)dx.$$

We would like to mention that under certain conditions, the computation in (5.4a) and (5.4b) by Trapezoidal's rule have exponentially decaying error. Also, the computations in (5.4c) are exact. Hence, algorithm 5.2.1 should obtain overall exponentially decaying error.

5.2.2 Discrete Approximation

In order to price the option numerically, we need to consider the densities in their discretization forms. Thanks to the re-centering scheme, $f_n(x)$ is centered around y-axis. Hence, we can truncate the domain symmetrically with respect to y-axis, say $[-\bar{x}, \bar{x}]$ for a large enough $\bar{x} > 0$. And then divide $[-\bar{x}, \bar{x}]$ into $2M$ equal intervals, each with length $\Delta x = \bar{x}/M$. Denote

$$x_m = -\bar{x} + m\Delta x, \quad m = 0, 1, ..., 2M.$$

Also, we need to compute some characteristic functions (Fourier transform of density functions) in the algorithm. Again, we select a symmetric and large enough domain $[-\bar{\xi}, \bar{\xi}]$ in the Fourier space. Divide $[-\bar{\xi}, \bar{\xi}]$ into $2K$ equal intervals, each with length $\Delta\xi = \bar{\xi}/K$. Denote

$$\xi_k = -\bar{\xi} + k\Delta\xi, \quad k = 0, 1, ..., 2K.$$

We therefore approximate the inverse Fourier transform (5.4a) and Fourier transform alike integral (5.4b) by the following scheme for $n = d, ..., 2$:

$$f_n(x_m) = \mathcal{F}^{-1}(\hat{f}_n(\xi))(x_m) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-i\xi x_m}\hat{f}_n(\xi)d\xi \approx \frac{1}{2\pi}\sum_{k=0}^{2K} e^{-i\xi_k x_m}\hat{f}_n(\xi_k)\Delta\xi, \quad m = 0, ..., 2M, \tag{5.5a}$$

$$\hat{g}_n(\xi_k) = \int_{-b_{n-1}}^{\infty} e^{i\xi_k y}g_n(y)dy = \int_{-\infty}^{\infty} e^{i\xi_k y}f_n(x)dx \approx \sum_{m=0}^{2M} e^{i\xi_k y_m}f_n(x_m)\Delta x, \quad k = 0, ..., 2K, \tag{5.5b}$$

where $y_m = \ln(e^{x_m + b_n} + 1) - b_{n-1}$ in each step. Similarly, we can obtain the approximation of $\hat{g}_2(\xi_k - i)$ by the following:

$$\begin{aligned}\hat{g}_2(\xi_k - i) &= \int_{-b_1}^{\infty} e^{i(\xi_k - i)y}g_n(y)dy = \int_{-\infty}^{\infty} e^{i(\xi_k - i)y}f_n(x)dx \\ &\approx \sum_{m=0}^{2M} e^{i(\xi_k - i)y_m}f_n(x_m)\Delta x, \quad k = 0, ..., 2K.\end{aligned}$$

Moreover, we still need to discretize the Hilbert transforms, which could be done by using the scheme in [38] section 2.4.2. Here we have:

$$\mathcal{H}(e^{-i\cdot\xi\cdot z_0}\hat{f}_1(\xi))(0) \approx \sum_{k=0, k\neq K}^{2K} e^{-iz_0\xi_k}\hat{f}_1(\xi_k)\frac{1 - (-1)^{-(k-K)}}{-\pi(k-K)}. \tag{5.6}$$

Then $\mathcal{H}(e^{-i\cdot\xi\cdot z_0}\hat{h}_1(\xi)(0)$ can be computed correspondingly. Now, we combine all pieces in the discretization steps and summarize the new scheme as following:

Scheme 1. *(Proposed scheme)*

1. $\hat{f}_d(\xi_k) = \phi(\xi_k)$, $k = 0, ..., 2K$.

2. *For* $n = d, ..., 2$,

$$f_n(x_m) \approx \frac{1}{2\pi}\sum_{k=0}^{2K} e^{-i\xi_k x_m}\hat{f}_n(\xi_k)\Delta\xi, \quad m = 0, ..., 2M, \tag{5.7a}$$

$$\hat{g}_n(\xi_k) \approx \sum_{m=0}^{2M} e^{i\xi_k y_m} f_n(x_m)\Delta x, \quad k = 0, ..., 2K, \tag{5.7b}$$

$$\hat{f}_{n-1}(\xi_k) = \phi(\xi_k)\cdot\hat{g}_n(\xi_k), \quad k = 0, ..., 2K, \tag{5.7c}$$

   where $y_m = \ln(e^{x_m+b_n}+1) - b_{n-1}$ *in each step.*

3. *When* $n = 2$, *compute*

$$\hat{g}_2(\xi_k - i) \approx \sum_{m=0}^{2M} e^{i(\xi_k - i)y_m} f_n(x_m)\Delta x, \quad k = 0, ..., 2K, \tag{5.7d}$$

$$\hat{h}_1(\xi_k) = \phi(\xi_k - i)\cdot\hat{g}_2(\xi_k - i), \quad k = 0, ..., 2K. \tag{5.7e}$$

4. *The Asian call option price is given by:*

$$\begin{aligned} c = {} & \frac{e^{-rT}}{2}\big(\frac{S_0}{d+1}e^{b_1}\hat{h}_1(\xi_K) + (\frac{S_0}{d+1} - K)\hat{f}_1(\xi_K)\big) \\ & + \frac{ie^{-rT}}{2}\sum_{k=0,k\neq K}^{2K} e^{-iz_0\xi_k}\big(\frac{S_0}{d+1}e^{b_1}\hat{h}_1(\xi_k) - (\frac{S_0}{d+1} - K)\hat{f}_1(\xi_k)\big)\frac{1-(-1)^{-(k-K)}}{-\pi(k-K)}. \end{aligned} \tag{5.7f}$$

5.3 Error Estimation

We will show that under certain conditions, the numerical scheme provided above, which uses Trapezoidal's rule, will obtain exponential convergence for both discretization error and truncation error in the whole recursive process.

We begin with a single step calculation. As we showed in Chapter 2, functions in the analytic class have very good numerical properties. The most important ones that we use include: the discretization errors for integration of ψ on $\mathbb{R}$, the Fourier and inverse Fourier transforms of ψ, as

well as the Hilbert transform of ψ, are of the order $O(\exp(-\pi d_a/h))$ for Trapezoidal's rule, where h is the step size for the discretization and d_a is some positive constant. It is then easy to observe that the discretization error decays exponentially in terms of $1/h$.

On the other hand, we may further assume $\psi(u+iv)$ has certain tail behavior in the analytic strip, say $v \in (d_-, d_+)$. For example, for many popular models in financial application, ψ has exponentially decaying tail:

$$|\psi(u+iv)| \leq \kappa e^{-c|u|^\nu}, \quad u \in \mathbb{R}. \tag{5.8}$$

here κ, c, ν are positive constants that may depend on v. But to simplify notation, we ignore the sub index when it will not cause confusion. Then, the truncation error is in the order of $O(\exp(-c(Lh)^\nu))$ when we truncate the integral region by $[-Lh, Lh]$, where L is a positive integer, and h, the step size, can also be denoted as Δu.

In this case, we can follow the approach proposed by Feng and Lin in [38] to balance the discretization error and truncation error. They set $\exp(-\pi d_a/h) = \exp(-c(Lh)^\nu)$, and then they are able to express step size h as a function of truncation level L as:

$$h(L) = (\pi d_a/c)^{\frac{1}{1+\nu}} L^{-\frac{\nu}{1+\nu}}, \quad L \geq 1. \tag{5.9}$$

Therefore, the total numerical error is in the order of $O(\exp(-c^{\frac{1}{1+\nu}}(\pi d_a L)^{\frac{\nu}{1+\nu}}))$. We suggest that readers may refer to [38] for more detail.

5.3.1 Numerical Error Analysis

Now we apply those lemmas in section (2.1) in our reasoning. When demonstrating our results, we first make some assumptions on $\hat{f}_n$ and f_n. Later in our applications, we will verify those assumptions. Also, we should mention that x and ξ in general are real variables, but when we discuss the analyticity in complex plane, we may use the same notation to express complex variables as well.

Theorem 5.3.1. *(Inverse Fourier transform) Suppose $\hat{f}_n \in H(\mathcal{D}_{(d_-,d_+)})$, then,*

$$\begin{aligned} |E^{IF}_{\Delta\xi}(\hat{f}_n, 0)(x)| &= \frac{1}{2\pi}\Big| \int_{\mathbb{R}} e^{-i\xi x}\hat{f}_n(\xi)d\xi - \sum_{k=-\infty}^{\infty} e^{-ixk\Delta\xi}\hat{f}_n(k\Delta\xi)\Delta\xi \Big| \\ &\leq \frac{e^{-2\pi(-d_-)/\Delta\xi}}{2\pi(1-e^{-2\pi(-d_-)/\Delta\xi})} e^{xd_-}\|\hat{f}_n\|^- + \frac{e^{-2\pi d_+/\Delta\xi}}{2\pi(1-e^{-2\pi d_+/\Delta\xi})} e^{xd_+}\|\hat{f}_n\|^+, \end{aligned} \tag{5.10}$$

where $d_a = 2\min(d_+, -d_-)$. *Also, suppose* $\hat{f}_n$ *satisfies (5.8) for* $v = 0$ *and some* $\kappa, c, \nu > 0$, *and* $\Delta\xi(K)$ *is selected according to (5.9). Then there exists a constant* $C > 0$ *independent of* K *such that*

$$|E^{IF}_{\Delta\xi(K),K}(\hat{f}_n, 0)(x)| = \frac{1}{2\pi}\Big| \int_{-\infty}^{\infty} e^{-i\xi x}\hat{f}_n(\xi)d\xi - \sum_{k=-K}^{K} e^{-ixk\Delta\xi}\hat{f}_n(k\Delta\xi)\Delta\xi\Big|$$
$$\leq C(M^{\frac{1-\nu}{1+\nu}} + e^{xd_-} + e^{xd_+})\exp(-c^{\frac{1}{1+\nu}}(\pi d_a K)^{\frac{\nu}{1+\nu}}).$$

Proof. First, as mentioned in Corollary 2.4 of [38]:

$$\mathcal{F}^{-1}\psi(x) = \frac{1}{2\pi}\int_{\mathbb{R}} e^{-ixy}\psi(y)dy = \frac{1}{2\pi}\mathcal{F}\psi(-x).$$

Then based on Lemma 2.1.3, the error involved in discretizing inverse Fourier transform is:

$$|E^{IF}_h(\psi, 0)(x)| = \frac{1}{2\pi}\Big| \int_{\mathbb{R}} e^{-ixy}\psi(y)dy - \sum_{l=-\infty}^{\infty} e^{-ixlh}\psi(lh)h\Big|$$
$$\leq \frac{e^{-2\pi(-d_-)/h}}{2\pi(1 - e^{-2\pi(-d_-)/h})}e^{xd_-}\|\psi\|^- + \frac{e^{-2\pi d_+/h}}{2\pi(1 - e^{-2\pi d_+/h})}e^{xd_+}\|\psi\|^+.$$

And the truncation error is:

$$|E^{IF}_{h(L),L}(\psi, 0)(x)| = \frac{1}{2\pi}\Big| \int_{-\infty}^{\infty} e^{-ixy}\psi(y)dy - \sum_{l=-L}^{L} e^{-ixlh}\psi(lh)h\Big|$$
$$\leq C(M^{\frac{1-\nu}{1+\nu}} + e^{xd_-} + e^{xd_+})\exp(-c^{\frac{1}{1+\nu}}(\pi d_a L)^{\frac{\nu}{1+\nu}}).$$

Replace ψ, y, h, l, L by $\hat{f}_n, \xi, \Delta\xi, k, K$ respectively, then we obtain the result. □

Theorem 5.3.2. *(Hilbert transform) Suppose* $e^{-i\xi z}\hat{f}_1(\xi) \in H(\mathcal{D}_{(d_-,d_+)})$, *then,*

$$|E^H_h(e^{-i\xi z}\hat{f}_1, 0)(0)| = \Big|\frac{1}{\pi}p.v.\int_{\mathbb{R}} \frac{e^{-i\xi z}\hat{f}_1(\xi)}{0 - \xi}d\xi - \sum_{k=-\infty, k\neq 0}^{\infty} e^{-ik\Delta\xi z}\hat{f}_1(k\Delta\xi)\frac{1 - (-1)^k}{-k\pi}\Big|$$
$$\leq \frac{e^{-\pi(-d_-)/\Delta\xi}}{\pi(-d_-)(1 - e^{-\pi(-d_-)/\Delta\xi})}e^{d_- z}\|\hat{f}_1\|^- + \frac{e^{-\pi d_+/\Delta\xi}}{\pi d_+(1 - e^{-\pi d_+/\Delta\xi})}e^{d_+ z}\|\hat{f}_1\|^+.$$

Let $d_a = \min(d_+, -d_-)$. *Also, for* $v = 0$, *suppose* $e^{-i\xi z}\hat{f}_1(\xi)$ *satisfies (5.8) for some* $\kappa, c, \nu > 0$, *and* $\Delta\xi(K)$ *is selected according to (5.9). Then there exists a constant* $C > 0$ *independent of* K *such*

that

$$|E^H_{\Delta\xi(K),K}(e^{-i\xi z}\hat{f}_1,0)(0)| = |\frac{1}{\pi}p.v.\int_{\mathbb{R}}\frac{e^{-i\xi z}\hat{f}_1(\xi)}{0-\xi}d\xi - \sum_{k=-K,k\neq 0}^{K} e^{-ik\Delta\xi z}\hat{f}_1(k\Delta\xi)\frac{1-(-1)^k}{-k\pi}|$$
$$\leq C(K^{\frac{1}{1+\nu}} + e^{d_- z} + e^{d_+ z})\exp(-c^{\frac{1}{1+\nu}}(\pi d_a K)^{\frac{\nu}{1+\nu}}).$$

Proof. Just replace $\psi(x)$ in Lemma 2.1.4 by $e^{-i\xi z}\hat{f}_1$, and replace l, h, L by $k, \Delta\xi$, and K respectively. Also, notice that

$$\|e^{-i\xi z}\hat{f}_1\|^- = e^{zd_-}\|\hat{f}_1\|^-, \quad \|e^{-i\xi z}\hat{f}_1\|^+ = e^{zd_+}\|\hat{f}_1\|^+.$$

Then we have:

$$|E^H_h(e^{-i\xi z}\hat{f}_1,0)(0)| \leq \frac{e^{-\pi(-d_-)/\Delta\xi}}{\pi(-d_-)(1-e^{-\pi(-d_-)/\Delta\xi})}e^{d_- z}\|\hat{f}_1\|^- + \frac{e^{-\pi d_+/\Delta\xi}}{\pi d_+(1-e^{-\pi d_+/\Delta\xi})}e^{d_+ z}\|\hat{f}_1\|^+$$
$$\leq \frac{e^{-\pi d_a/\Delta\xi}}{\pi d_a(1-e^{-\pi d_a/\Delta\xi})}(e^{d_- z}+e^{d_+ z})(\|\hat{f}_1\|^- + \|\hat{f}_1\|^+).$$

Then plug in $\Delta\xi(K)$, follow the same argument in Theorem 2.7 and Corollary 2.8 in [38], we can obtain:

$$|E^H_{\Delta\xi(K),K}(e^{-i\xi z}\hat{f}_1,0)(0)| \leq C(K^{\frac{1}{1+\nu}} + e^{d_- z} + e^{d_+ z})\exp(-c^{\frac{1}{1+\nu}}(\pi d_a K)^{\frac{\nu}{1+\nu}}).$$

□

Remark 5.3.3. *For the Hilbert transform approximation, we are able to further improve the bound. See section 2.4.3 a special case in [38]. But here, since the main point is the exponential convergence, we do not push the bound to its best.*

Theorem 5.3.4. *(Modified based on Lemma 2.1.2) Suppose $e^{i\xi y(x)}f_n(x) \in H(\mathcal{D}_{(d_-,d_+)})$ for some $d_- < 0$ and $d_+ > 0$, where $y(x) = \ln(e^{x+b_n}+1) - b_{n-1}$. Then,*

$$|E^T_{\Delta x}(e^{i\xi y(x)}f_n(x),0)(\xi)| = |\int_{-\infty}^{\infty} e^{i\xi y(x)}f_n(x)dx - \sum_{m=-\infty}^{\infty} e^{i\xi y(m\Delta x)}f_n(m\Delta x)\Delta x|$$
$$\leq \frac{e^{-2\pi(-d_-)/\Delta x}}{1-e^{-2\pi(-d_-)/\Delta x}}e^{\pi|\xi|}\|f_n\|^- + \frac{e^{-2\pi d_+/\Delta x}}{1-e^{-2\pi d_+/\Delta x}}e^{\pi|\xi|}\|f_n\|^+. \tag{5.11}$$

Denote $d_a = 2\min(d_+, -d_-)$ and for $v = 0$, suppose ψ satisfies (5.8) for some $\kappa, c, \nu > 0$, and h is

selected according to (5.9), then there exists a constant $C > 0$ independent of M such that

$$|E^T_{\Delta x(M),M}(e^{i\xi y(x)} f_n(x), 0)(\xi)| = |\int_{-\infty}^{\infty} e^{i\xi y(x)} f_n(x) dx - \sum_{m=-M}^{M} e^{i\xi y(m\Delta x)} f_n(m\Delta x)\Delta x|$$
$$\leq C(M^{\frac{1-\nu}{1+\nu}} + e^{|\xi|\pi}) \exp(-c^{\frac{1}{1+\nu}} (\pi d_a M)^{\frac{\nu}{1+\nu}}).$$

Proof. First, we replace ψ and h in Lemma 2.1.2 with $e^{i\xi y(x)} f_n(x)$ and Δx. Then, we can easily obtain:

$$|E^T_{\Delta x}(e^{i\xi y} f_n(x), 0)| \leq \frac{e^{-2\pi(-d_-)/\Delta x}}{1 - e^{-2\pi(-d_-)/\Delta x}} \|e^{i\xi y(x)} f_n(x)\|^- + \frac{e^{-2\pi d_+/\Delta x}}{1 - e^{-2\pi d_+/\Delta x}} \|e^{i\xi y(x)} f_n(x)\|^+.$$

Now we consider $\|e^{i\xi y(x)} f_n(x)\|^+$. Here we are safe to set $x = \eta + i d_+$, both η and d_+ are in $\mathbb{R}$, then,

$$y(x) = \ln(e^{(\eta + i d_+) + b_n} + 1) - b_{n-1} = \ln(s) - b_{n-1},$$

where $s = e^{\eta + b_n}\big(\cos(d_+) + i \sin(d_+)\big) + 1$ is complex. If we define $\theta = \arg(s)$, which is the principle argument of s. Then

$$y(x) = \ln(|s|) - b_{n-1} + i\theta.$$

Notice that both $\ln(|s|) - b_{n-1}$ and θ is real. Also, here ξ is real. Hence we have:

$$|e^{i\xi y(x)}| = |e^{i\xi\left(\ln(|s|) - b_{n-1} + i\theta\right)}| = |e^{-\xi\theta}| \leq e^{\pi|\xi|}.$$

Therefore $\|e^{i\xi y(x)} f_n(x)\|^+ \leq e^{\pi|\xi|} \|f_n(x)\|^+$. Similarly, we can obtain $\|e^{i\xi y(x)} f_n(x)\|^- \leq e^{\pi|\xi|} \|f_n(x)\|^-$. Combine those two together, we have

$$|E^T_{\Delta x}(e^{i\xi y} f_n(x), 0)| \leq \frac{e^{-2\pi(-d_-)/\Delta x}}{1 - e^{-2\pi(-d_-)/\Delta x}} e^{\pi|\xi|} \|f_n\|^- + \frac{e^{-2\pi d_+/\Delta x}}{1 - e^{-2\pi d_+/\Delta x}} e^{\pi|\xi|} \|f_n\|^+.$$

Now we follow the same argument as Theorem 2.3 in [38] and obtain

$$|E^T_{\Delta x(M),M}(e^{i\xi y(x)} f_n(x), 0)| = |\int_{-\infty}^{\infty} e^{i\xi y(x)} f_n(x) dx - \sum_{m=-M}^{M} e^{i\xi y(m\Delta x)} f_n(m\Delta x)\Delta x|$$
$$\leq C(M^{\frac{1-\nu}{1+\nu}} + e^{\pi|\xi|}) \exp(-c^{\frac{1}{1+\nu}} (\pi d_a M)^{\frac{\nu}{1+\nu}}).$$

□

Remark 5.3.5. *Since we always have $\psi \in H(\mathcal{D}_{(d'_-,d'_+)})$ given $\psi \in H(\mathcal{D}_{(d_-,d_+)})$ and $d_- < d'_- < 0 < d'_+ < d_+$. Then, if needed, we may set $-\pi < d_- < 0 < d_+ < \pi$. This restriction may avoid multiple branches issue for complex logarithm in our computation.*

5.3.2 Main Results

In this part, we show that for certain characteristic function $\phi(\xi)$ of Lévy process, the assumptions in section 5.3.1 are satisfied. And hence, we are able to obtain overall exponential convergence for our numerical error. Basically, we want to prove when considering ξ as a complex variable, $\hat{f}_n(\xi)$, $e^{-i\xi z_0}\hat{f}_1(\xi)$ as well as $e^{i\xi y(x)}f_n(x)$, for $n = d, d-1, ...1$ are all in $H(\mathcal{D}_{(d_-,d_+)})$. Although d_- and d_+ may possibly be different for different n.

First, notice that z_0 in $e^{-i\xi z_0}\hat{f}_1(\xi)$ is a constant, which implies that $e^{-i\xi z_0}$ is entire if we consider ξ as a complex variable. Also, $|e^{-i\xi z_0}|$ is bounded in the analytic strip $\Im(\xi) \in (d_-, d_+)$. Hence, in order to show $e^{-i\xi z_0}\hat{f}_1(\xi)$ is in $H(\mathcal{D}_{(d_-,d_+)})$, we only need to show $\hat{f}_1(\xi)$ is in $H(\mathcal{D}_{(d_-,d_+)})$. Also, it is not hard to see that $|e^{i\xi y(x)}|$ is bounded by $e^{\pi|\xi|}$ and is analytic in the strip $-\pi < \Im(x) < \pi$. Hence, in order to show $e^{i\xi y(x)}f_n(x)$ are in some analytic class, it is suffice to show $f_n(x)$ are in some $H(\mathcal{D}_{(d_-,d_+)})$ for $n = d, ..., 1$. Then, $e^{i\xi y(x)}f_n(x)$ will be in $H(\mathcal{D}_{(d'_-,d'_+)})$ where $(d'_-, d'_+) = (d_-, d_+)\bigcap(-\pi, \pi)$. We now prove that $\hat{f}_n(\xi)$ and $f_n(x)$ are in some analytic class through the following two theorems.

Theorem 5.3.6. *If $\phi(z)$, as a complex continuation of characteristic function $\phi(\xi)$, is in $H(\mathcal{D}_{(d_-,d_+)})$, then $\hat{f}_n$, for $n = d, ..., 1$, in Algorithm 5.2.1 are all in the same analytic class.*

Proof. We prove this by induction. Since $\hat{f}_d(z) = \phi(z)$, the $n = d$ case is trivial. Then assume $\hat{f}_k(z)$ is in the analytic class, we want to verify $\hat{f}_{k-1}(z)$ is also in this analytic class.

First, we show that $\hat{f}_{k-1}(z)$ is analytic in $d_- < \Im(z) < d_+$ given $\hat{f}_k(z)$ is analytic in $d_- < \Im(z) < d_+$. Denote the corresponding distribution function of $f_k(x)$ and $g_k(x)$ by $F_k(x)$ and $G_k(x)$ respectively. Then, since $\hat{f}_k$ is analytic in $d_- < \Im(z) < d_+$, by thm 2.2.2, we know:

$$1 - F_k(x) = o(e^{-rx}) \quad as \quad x \to \infty$$

holds for all $0 < r < -d_-$. Then for any positive c, $1 - F_k(x) \le ce^{-rx}$ when x is large enough. Then

$$1 - G_k(x) = 1 - F_k(\ln(e^x - 1)) \le ce^{-r\ln(e^x-1)} = c(e^x - 1)^{-r} \le 2^r ce^{-rx} = c_1 e^{-rx}.$$

$$1 - G_k(x) = 1 - F_k(\ln(e^{x+b_{k-1}} - 1) - b_k) \le ce^{-r(\ln(e^{x+b_{k-1}}-1)-b_k)}$$
$$= ce^{rb_k}(e^{x+b_{k-1}} - 1)^{-r} \le 2^r ce^{r(b_n - b_{n-1})}e^{-rx} = c_1 e^{-rx}.$$

And we know

$$1 - G_k(x) = o(e^{-rx}) \quad as \quad x \to \infty$$

holds for all $0 < r < -d_-$. Similarly, we can show

$$G_k(-x) = o(e^{-rx}) \quad as \quad x \to \infty$$

holds for all $0 < r < d_+$. Notice that $\hat{g}_k$ is the corresponding characteristic function of G_k. Then, again by thm 2.2.2, we know $\hat{g}_k$ is analytic in $d_- < \Im(z) < d_+$. Hence, as a product of $\phi(z)$ and $\hat{g}_k$, we know $\hat{f}_{k-1}$ is also analytic in $d_- < \Im(z) < d_+$.

Now we verify that given $\hat{f}_k$ satisfies (2.1), condition (2.1) holds for $\hat{f}_{k-1}$. Notice that $\hat{f}_{k-1}(z) = \phi(z) \cdot \hat{g}_k(z)$, hence we only need to show $|\hat{g}_k(z)|$ is bounded in the analytic strip. To show this, we need to know the tail behavior of g_k. Since we already know $\hat{f}_k \in H(\mathcal{D}_{(d_-,d_+)})$, by thm 2.2.6, we know f_k has the exponential tail behavior on real axis. By similar argument used previously for G_k and F_k, we know g_k has the same tail behavior. Then

$$|\hat{g}_n(z)| = |\int_{-b_{n-1}}^{\infty} e^{izy} g_n(y)dy| \le \int_{-b_{n-1}}^{\infty} e^{-\Im(z)y} g_n(y)dy.$$

And the last integral is bounded for any $-d_- < \Im(z) < d_+$. Hence, we know $|\hat{g}_n(z)|$ is bounded. And thus $\hat{f}_{k-1}$ satisfies condition (2.1). And we know $\hat{f}_{k-1} \in H(\mathcal{D}_{(d_-,d_+)})$. By induction, we know $\hat{f}_n \in H(\mathcal{D}_{(d_-,d_+)})$ for any $n = d, ..., 1$. □

Now we show that, for all $n = d, ..., 1$, f_n are in analytic class:

Theorem 5.3.7. *If $\phi(\omega)$ is in $H(\mathcal{D}_{(d_-,d_+)})$, and has the following tail behavior in the analytic strip (on the real axis as well as the boundary of its analytic strip):*

$$\phi(u+iv) = \begin{cases} O(e^{-(a-\epsilon)u}) \quad as \quad u \to \infty; \\ O(e^{(b-\epsilon)u}) \quad as \quad u \to -\infty \end{cases}$$

where $\omega = u+iv$. Then f_n computed in Algorithm 5.2.1 are all in the same analytic class $H(\mathcal{D}_{(-b,a)})$, for $n = d, ..., 1$.

Proof. First, for $n = d, ..., 1$, we know $\hat{f}_n$ are in $H(\mathcal{D}_{(d_-,d_+)})$. Also, since

$$\hat{f}_n(\omega) = \phi(\omega) \cdot \hat{g}_{n+1}(\omega),$$

and we already showed that $\hat{g}_{n+1}(\omega)$ is bounded in the analytic strip, then we know $\hat{f}_n$ has the same tail behavior as ϕ does, for $n = d, ..., 1$. Hence, the assumptions posed on $\phi(\omega)$ of this theorem holds for all $\hat{f}_n$.

By the same argument used in theorem 2.2.4, we know $f_n(z)$, $z = x + iy$, are analytic in strip $-b < y < a$. And by proposition 2.2.9, we know f_n has the following tail behavior in the analytic strip:

$$f_n(z) = \begin{cases} O(e^{(d_- + \epsilon)x}) & as \quad x \to \infty; \\ O(e^{(d_+ - \epsilon)x}) & as \quad x \to -\infty. \end{cases}$$

Then

$$\int_{-a}^{b} |f_n(x + iy)| dy \to 0, \quad x \to \pm\infty,$$

$$||f_n||^+ := \lim_{\epsilon \to 0+} \int_{\mathbb{R}} |f_n(x + i(b - \epsilon))| dx < +\infty,$$

$$||f_n||^- := \lim_{\epsilon \to 0+} \int_{\mathbb{R}} |f_n(x + i(-a + \epsilon))| dx < +\infty.$$

Hence, condition (2.1) are satisfied. And f_n are in $H(\mathcal{D}_{(-b,a)})$, for $n = d, ..., 1$. □

5.4 Numerical Results

In this section, we first verify the exponential convergence result shown in section 5.3.1 by numerical experiments. Then we compare our method with some other methods, like Fusai and Meucci's in [42] and Cerny and Kyriakou's in [68]. Also, we will design a procedure to illustrate how to determine Δx, $\Delta\xi$, M and K in our algorithm 5.2.1 and corresponding numerical scheme.

5.4.1 Exponential Convergence Result

We want to observe how pricing error change if we change Δx and $\Delta\xi$ so as to find the convergence rate. We first fix $\Delta\xi$ small enough and K large enough. By doing this, we want to ensure that the numerical error caused by computation in Fourier space is negligible. In the meantime, we fix the truncation level in state space large enough, that is $M \cdot \Delta x$, and then change the value of Δx. By doing this, we make sure that all the numerical error is due to the value of Δx.

M	200	180	170	160	150	140	120
Pricing error	5.54E-11	1.49E-08	1.79E-07	1.79E-06	1.50E-05	1.05E-04	3.01E-03

Table 5.1: Pricing error of Discrete Asian call w.r.t Δx in Kou's model

K	450	425	400	380	360	350	340
Pricing error	7.30E-13	1.36E-10	1.54E-08	6.18E-07	2.21E-05	1.35E-04	8.59E-04

Table 5.2: Pricing error of Discrete Asian call w.r.t $\Delta\xi$ in Kou's model

We first consider Kou's model. We take the following parameters:

$$\sigma = 0.1, \lambda = 3, p = 0.3, \eta_1 = 40, \eta_2 = 12, r = 0.01, q = 0, S_0 = K = 100, T = 1, d = 12.$$

After several trials, we take $K = 6000$, $\Delta\xi = 0.058$, $M = 800$, $\Delta x = 0.00625$ to get the benchmark price 6.916117761635. Then, we fix K, $\Delta\xi$ at the same level, fix $M \cdot \Delta x = 5$, and decrease the value of M, and hence Δx will increase, and we can observe how the pricing error changes according to the value change of M and hence Δx. We show the result in table 5.1. The phenomena that error decays exponentially is fairly clear from the data itself. But to make it more observable, we draw a log-log plot with the pricing error in y-axis and $1/\Delta x$ in x-axis, as in the left graph in Figure 5.1. We can do the same thing for $\Delta\xi$. That is, we fix $M = 800$, $\Delta x = 0.00625$, and fix $K \cdot \Delta\xi = 1000$. Then, we decrease K, which has the same effect as increasing $\Delta\xi$, and observe how accordingly the pricing error changes. The result is shown in table 5.2. Just as the case for Δx, we can easily observe the exponential convergence. And the corresponding log-log plot is provided as in the right graph of Figure 5.1.

Similarly, we can verify the exponential convergence result in CGMY model. The parameters we take are:

$$C = 4, G = 50, \mathbb{M} = 60, Y = 1.5, r = 0.05, q = 0.02, S_0 = K = 100, T = 0.5, d = 12.$$

Here we take $K = 8000$, $\Delta\xi = 0.0125$, $M = 600$, $\Delta x = 0.01$ to get the benchmark price 21.54967237579. Then, we fix K, $\Delta\xi$ at the same level, fix $M \cdot \Delta x = 6$, and decrease the value of M. The result in table 5.3. The phenomena of exponential decay is also very obvious. And the log-log plot is shown as the left one in Figure 5.2. Then, we can do the same thing for $\Delta\xi$. Fix $M = 600$,

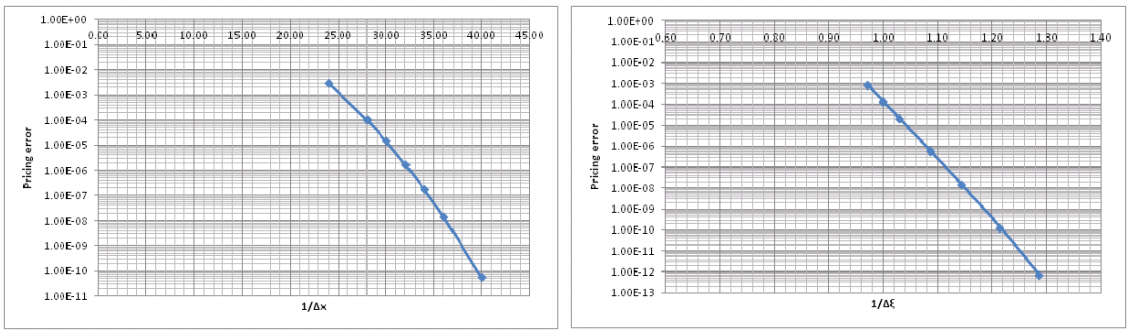

Figure 5.1: Pricing error vs Δx and $\Delta\xi$ in Kou's model .

M	28	24	22	20	19	18	16
Pricing error	1.72E-10	1.40E-09	4.53E-08	3.28E-07	1.98E-06	2.88E-05	1.31E-03

Table 5.3: Pricing error of Discrete Asian call w.r.t Δx in CGMY's model

$\Delta x = 0.01$, and fix $K \cdot \Delta\xi = 100$. Then, we decrease K and observe how the pricing error changes accordingly, as shown in table 5.4. And the corresponding log-log plot is the right one in Figure 5.2. Although the graph is not quite a straight line, we can still observe the trend of exponential decay.

5.4.2 Comparison with other methods

Now we do two simple comparisons between our result and the one obtained by Fusai and Meucci in [42] as well as the one by Cerny and Kyriakou in [68]. First, we compare the option prices obtained by our method and those two methods, as listed in table 5.5. We intend to verify if our result agrees with the other two. Computation is in BSM model with the following parameters: $r = 0.0367$, $\sigma = 0.17801$, $S_0 = 100$, $T = 1$ and $d = 50$. And we find that our price agrees with Cerny and Kyriakou's method to all the digits they provided. And we also confirmed the following fact as claimed by Cerny and Kyriakou in [68]: the method in [42] can barely reach the precision of 1E-5. And sometimes only 1E-4 could be obtained.

Then, we compare the computational time of our method with the one provided by Cerny and

K	170	160	150	138	136	134	130
Pricing error	3.26E-10	1.07E-08	1.12E-07	2.32E-06	1.63E-05	7.83E-05	1.23E-03

Table 5.4: Pricing error of Discrete Asian call w.r.t $\Delta\xi$ in CGMY's model

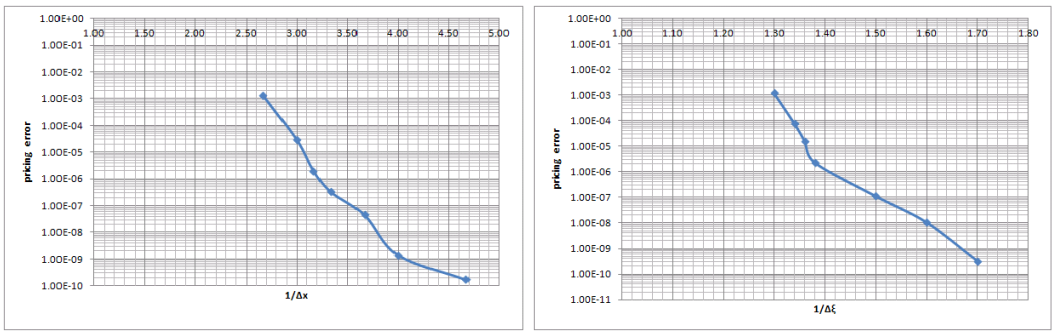

Figure 5.2: Pricing error vs Δx and $\Delta\xi$ in CGMY's model .

Strike K	Price of our method	Price of CK	Error	Price of FM	Error
90	11.93293820	11.9329382	±1E-7	11.93301	7E-05
100	4.93720281	4.9372028	±1E-7	4.93736	16E-05
110	1.40251551	1.4025155	±1E-7	1.40264	12E-05

Table 5.5: Comparison with the numerical scheme proposed in [42] and [68]

Kyriakou in [68] as shown in table 5.6. Here BSM model is used with $r = 0.04$, volatility $\sigma = 0.1$, 0.3 and 0.5, $S_0 = K = 100$. We can see that, their algorithm provides option prices with accuracy of around ±1E5. And ours is at least ±1E8. And for the CPU time, ours is only half of theirs. And for small volatility $\sigma = 0.1$, our CPU time is almost one tenth of their time. This indeed demonstrates the power of exponential convergence.

5.4.3 Automatic parameters determination

One disadvantage of our method is that we have four parameters M, Δx, K and $\Delta\xi$ to tune with. It would be annoying if we have to choose it manually. Hence, here we come up with an automatic procedure to choose those parameters.

First we determine a tolerance level ϵ. Then from (5.10), we can choose an appropriate $\Delta\xi$ such

σ	Price of our method	our CPU	Price of Cerny & Kyriakou	CK Error	CK CPU
0.1	3.33861711	0.12s	3.33861	±1E5	1.0s
0.3	7.69859895	0.16s	7.69859	±1E5	0.3s
0.5	12.09153558	0.14s	12.09153	±1E5	0.3s

Table 5.6: Comparison with the numerical scheme proposed in [68]

that both the first term and the second term in

$$\frac{e^{-2\pi(-d_-)/\Delta\xi}}{2\pi(1-e^{-2\pi(-d_-)/\Delta\xi})}e^{xd_-}\|\hat{f}_n\|^- + \frac{e^{-2\pi d_+/\Delta\xi}}{2\pi(1-e^{-2\pi d_+/\Delta\xi})}e^{xd_+}\|\hat{f}_n\|^+$$

are less than 0.5ϵ. When $n = d$, we have $\hat{f}_n = \phi$, then, $\|\hat{f}_n\|^\pm = \|\phi\|^\pm$. For $n < d$, $\hat{f}_n = \phi \cdot \hat{g}_{n+1}$, then

$$\|\hat{f}_n\|^\pm = \int_{\mathbb{R}} |\hat{f}_n(\xi + i(d_\pm \mp \epsilon))|d\xi = \int_{\mathbb{R}} |\phi(\xi + i(d_\pm \mp \epsilon)) \cdot \hat{g}_{n+1}(\xi + i(d_\pm \mp \epsilon))|d\xi.$$

Hence,

$$\|\hat{f}_n\|^\pm \le \|\phi\|^\pm \cdot \max_{\xi\in\mathbb{R}} |\hat{g}_{n+1}(\xi + i(d_\pm \mp \epsilon))| \quad \text{and} \quad \|\hat{f}_n\|^\pm \le \|\hat{g}_{n+1}\|^\pm \cdot \max_{\xi\in\mathbb{R}} |\phi(\xi + i(d_\pm \mp \epsilon))|.$$

We can estimate $\max_{\xi\in\mathbb{R}} |\hat{g}_{n+1}(\xi+i(d_\pm\mp\epsilon))|$ by numerical calculation, then it is not hard to estimate $\|\hat{f}_n\|^\pm$.

From [38], given $|e^{-i\xi x}\hat{f}_n(\xi)| = |\hat{f}_n(\xi)| = |\phi(\xi)\cdot\hat{g}_n(\xi)| \le |\phi(\xi)| \le \kappa e^{-c|\xi|^\nu}$ for real variable ξ, we have the following bound for truncation error in inverse Fourier transform:

$$\begin{aligned}|C_{\Delta\xi}(e^{-i\xi x}\hat{f}_n(\xi), 0) - C_{\Delta\xi,K}(e^{-i\xi x}\hat{f}_n(\xi), 0)| &\le 2\sum_{k\ge K+1} \kappa e^{-c(k\Delta\xi)^\nu} \le \frac{2\kappa\Delta\xi^{-1}}{\nu c^{1/\nu}}\Gamma(\frac{1}{\nu}, c(K\Delta\xi)^\nu) \\ &\le \frac{2\kappa\Delta\xi^{-1}}{\nu c^{1/\nu}}(K\Delta\xi)^{1-\nu}e^{-c(K\Delta\xi)^\nu}.\end{aligned}$$

By restricting the truncation error less than ϵ, we can get a minimum K. And also, by doing this, the total error for the Fourier inverse (5.4a) are less than 2ϵ.

Then for the computation in (5.4b), similar procedure can be applied. From (5.11), we can choose Δx by setting both terms in

$$\frac{e^{-2\pi(-d_-)/\Delta x}}{1-e^{-2\pi(-d_-)/\Delta x}}e^{\pi|\xi|}\|f_n\|^- + \frac{e^{-2\pi d_+/\Delta x}}{1-e^{-2\pi d_+/\Delta x}}e^{\pi|\xi|}\|f_n\|^+$$

to be less than 0.5ϵ. Here $\|f_n\|^\pm$ is approximated by $|f_n(x + id_\pm)|$ at $x = 0$. And also, once the characteristic function is given, we can give explicit bound for $|f_n(x + id_\pm)|$. Take Kou's model as

an example, we have the following result:

$$\begin{aligned}|f_n(x+id_\pm)| &\leq \frac{1}{2\pi}\int_{\mathbb{R}} |e^{-i\xi x+\xi d_\pm}\cdot \hat{f}_n(\xi)|d\xi \leq \frac{1}{2\pi}\int_{\mathbb{R}} e^{\xi d_\pm}|\hat{f}_n(\xi)|d\xi \\ &\leq \frac{1}{2\pi}\int_{\mathbb{R}} e^{\xi d_\pm}|\phi(\xi)|d\xi \leq \frac{1}{2\pi}\int_{\mathbb{R}} e^{\xi d_\pm}e^{-\Delta t\frac{\sigma^2}{2}|\xi|^2}d\xi \\ &= \frac{1}{\sqrt{2\pi\Delta t}\sigma}e^{\frac{d_\pm^2}{2\sigma^2\Delta t}}.\end{aligned}$$

Here, only the exponential part $\frac{d_\pm^2}{2\sigma^2\Delta t}$ is relevant to us since it dominates the bound. Δx is chosen so that the exponential part is under control.

And also, from Remark 2.2.7 we know that $|f(x)| \leq \frac{1}{2\pi}\|\hat{f}\|^- e^{-\alpha x}$ and also $|f(x)| \leq \frac{1}{2\pi}\|\hat{f}\|^+ e^{\beta x}$. Hence, we have the following inequality:

$$|f_n(x)| \leq \kappa_n e^{-c_n|x|^{\nu_n}},$$

where $\nu_n = 1$, $\kappa_n = \frac{1}{2\pi}\max(\|\hat{f}_n\|^+, \|\hat{f}_n\|^-)$, and $c_n = \min(-d_-, d_+)$. d_- and d_+ is the analytic strip boundary for $\hat{f}_n$. Then, M can be chosen by restricting the truncation error

$$|C_{\Delta x}(e^{i\xi y(x)}f_n, 0) - C_{\Delta x, M}(e^{i\xi y(x)}f_n, 0)| \leq \frac{2\kappa_n \Delta x^{-1}}{\nu_n c_n^{1/\nu_n}}(M\Delta x)^{1-\nu_n}e^{-c(M\Delta x)^{\nu_n}}$$

to be less than ϵ.

Chapter 6

Model Calibration with European and American option Data

In this part, we intend to calibrate Lévy process models from both European vanilla option and American option data. The calibration problem has two parts: the forward option pricing problem and the inverse optimization problem. Both problems are very challenging. For the forward problem, efficient pricer should be used and better with the property that multiple options with different strikes and maturities could be priced simultaneously. For the inverse problem, the objective function is ill-posed when the most popular least square formulation is used. And hence, regularization techniques should be considered. In our research, for the forward problem, a new scheme for pricing European vanilla options by sinc type method [65] is proposed, which is fast and accurate. We use Bermudan option price to approximate American option price. For single Bermudan option, it can be efficiently priced by the backward induction method proposed by Feng and Lin in [39]. For calibration purpose, we made modifications to the original algorithm. So that, one can obtain multiple option prices with different strike prices and maturities through one run of the backward induction. For the inverse problem, we propose a two steps procedure, combining the global search and the local search. For local search, Tikhonov regularization is used. Important issues like gradient estimation will also be discussed. Moreover, we will carry out empirical study to verify the effectiveness of the proposed calibration method.

6.1 Pricing European Options

In this part, we propose a new numerical scheme for pricing European vanilla options and we illustrate it by pricing European put option.

6.1.1 Transform method for European options

If we denote the maturity of an option by T and strike price by K. Then, the risk neutral pricing formula gives us the following put option price:

$$p = e^{-rT}\mathbb{E}[(K - S_T)^+]$$

Here, $(K - S_T)^+ = K - S_T$ if $K \geq S_T$, otherwise $(K - S_T)^+ = 0$.

An inverse Fourier transform representation is proposed in [16]. Notice that they used call option as example, but we can derive the following representation for put option in the same manner:

$$p = -\frac{K}{2\pi}e^{-rT}\int_{\mathbb{R}} e^{-i(\xi - i(\alpha+1))\ln(K/S_0)}\frac{\phi(\xi - i(\alpha+1))}{(\xi - i\alpha)(\xi - i(\alpha+1))}d\xi, \quad \alpha < -1. \tag{6.1}$$

Damping factor $\alpha < -1$ is required to ensure integrability. In order to reduce discretization error, Carr and Madan applied Simpson's rule in computing the above integral in [16]. However, simpler Trapezoidal's rule would provide much faster convergence rate in this case, as we will discuss later. In their later paper [15], they reported that large pricing error is found by using the above method, especially for extreme strike prices, like deep in-the-money or out-of-the-money options. This is very unideal for model calibration, since we need to price different options whose strike prices varies significantly. On the other hand, the authors also proposed another method in [15] using saddlepoint approximation. Although this method performs better than the previous one for options with extreme strike prices, it is not the case when the strike prices are close to the initial underlying asset price. Hence, any single method of these two could not succeed in model calibration alone. This motivates us to propose a robust numerical scheme. We plan to do so by first modify the original representation of (6.1) and then extend it to a more general form.

First, we set $z = \xi - i(\alpha + 1)$, change variable in (6.1) and obtain the following:

$$p = -\frac{K}{2\pi}e^{-rT}\int_{-\infty - i(\alpha+1)}^{+\infty - i(\alpha+1)} e^{-iz\ln(K/S_0)}\frac{\phi(z)}{z(z+i)}dz, \quad \alpha < -1.$$

Assume $\phi \in H(\mathcal{D}_{(d_-,d_+)})$. Then, as long as $d_- < -(a+1) < d_+$, by Cauchy integral theorem, one may shift the integration line to $-(a+1)i$ for any a by considering contour integration and picking up corresponding residues. The residue at the origin can be easily computed as:

$$-\frac{K}{2\pi}e^{-rT}\lim_{z\to 0} e^{-iz\ln(K/S_0)}\frac{\phi(z)}{z+i} = -\frac{1}{2\pi i}Ke^{-rT}.$$

Similarly, the residue at $-i$ is:

$$-\frac{K}{2\pi}e^{-rT}\lim_{z\to -i}e^{-iz\ln(K/S_0)}\frac{\phi(z)}{z+i}=-\frac{1}{2\pi i}S_0e^{-qT}.$$

Then, for arbitrary value a, we are able to obtain similar format of representations for European put option:

$$p=-\frac{1}{2\pi}Ke^{-rT}p_0+\begin{cases}Ke^{-rT}-S_0e^{-qT}, & a>0,\\ Ke^{-rT}-\frac{1}{2}S_0e^{-qT}, & a=0,\\ Ke^{-rT}, & -1<a<0,\\ \frac{1}{2}Ke^{-rT}, & a=-1,\\ 0, & a<-1.\end{cases} \tag{6.2}$$

$$p_0=(p.v.)\int_{\mathbb{R}}e^{-i(\xi-i(a+1))\ln(K/S_0)}\frac{\phi(\xi-i(a+1))}{(\xi-ia)(\xi-i(a+1))}d\xi.$$

When $a=0$ or -1, p_0 is defined by Cauchy principle value.

6.1.2 A new scheme

Now, the main issue is how to evaluate the following integral:

$$V(f,a)=(p.v.)\int_{\mathbb{R}}\frac{f(x-i(a+1))}{(x-ia)(x-i(a+1))}.$$

When function $f\in H(\mathcal{D}_{(d_-,d_+)})$, we would like to propose the following numerical scheme, which is able to approximate $V(f,a)$ accurately:

$$\begin{aligned}V_h(f,a)=h\sum_{m=-\infty}^{\infty}\frac{f(mh-i(a+1))}{(mh-ia)(mh-i(a+1))}\Big[1+i(mh-ia)\\ \mathcal{E}_{a+1}(-i\pi(mh-i(a+1))/h)-i(mh-i(a+1))\mathcal{E}_a(-i\pi(mh-ia)/h)\Big].\end{aligned} \tag{6.3}$$

Here $\mathcal{E}_a(z)$ is defined as $\mathcal{E}_a(z)=\cos(iz)-i\cdot\sin(iz)\cdot sgn(a)$, which equals $\exp(z\cdot sgn(a))$ if $a\neq 0$. And also, when $a=0$ or $a=-1$ and $m=0$, we define the $[\cdot]$ part as $1-e^{-\pi/h}$ by convention.

If we denote the discretization error by $E_h^V(f,a)=V(f,a)-V_h(f,a)$, then we are able to derive $E_h^V(f,a)$ explicitly as shown below:

Theorem 6.1.1. *Suppose $f \in H(\mathcal{D}_{(d_-,d_+)})$, then for any $-(a+1) \in (d_-, d_+)$,*

$$E_h^V(f,a) = \lim_{\epsilon \to 0+} \Big(\int_{-\infty+i(d_-+\epsilon)}^{\infty+i(d_-+\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))} (\frac{e^{-\frac{\pi}{h}|a|}}{z+i} - \frac{e^{-\frac{\pi}{h}|a+1|}}{z} + \frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}) dz$$
$$- \int_{-\infty+i(d_+-\epsilon)}^{\infty+i(d_+-\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))} (\frac{e^{-\frac{\pi}{h}|a|}}{z+i} - \frac{e^{-\frac{\pi}{h}|a+1|}}{z} + \frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}) dz \Big). \tag{6.4}$$

Proof. We now discuss different situations when $-d_+ - 1 < a < -1$, $a = -1$, $-1 < a < 0$, $a = 0$ and $a > 0$.

We first consider the case when $-d_+ - 1 < a < -1$, or $0 < -(a+1) < d_+$. Consider the following contour integral

$$\int_\gamma \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))} \Big(\frac{e^{\frac{\pi}{h}a}}{z+i} - \frac{e^{\frac{\pi}{h}(a+1)}}{z} + \frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)} \Big) dz,$$

where γ is counterclockwise boundary of a box lying in the analytic strip. And when we expand the box to fill the whole analytic strip region, the integration line would be contained in the box γ. More specifically, we define $\gamma = \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4$. Here $\gamma_1 = x + i(d_+ - \epsilon)$ is the horizontal line segment with imaginary part $d_+ - \epsilon$, and x goes from $N > 0$ (we will take the limit $N \to +\infty$ later) to $-N$, for some $\epsilon > 0$ small enough. Similarly, γ_3 is the horizontal line segment with imaginary part $d_- + \epsilon$, and x goes from $-N$ to N. Then γ_2 and γ_4 is defined as vertical line segment so that those four segment could together compose the directional boundary of box γ. The integrand has poles of order 1 at $\{mh - i(a+1), m \in \mathbb{Z}\}$ (note that $z = 0$ and $z = -i$ are removable singularities). The residue of the integrand at $mh - i(a+1)$ can be computed easily as

$$-\frac{hf(mh - i(a+1))}{2\pi i(mh - ia)(mh - i(a+1))} \big(1 + i(mh - ia)e^{i\frac{\pi}{h}(mh - i(a+1))} - i(mh - i(a+1))e^{i\frac{\pi}{h}(mh - ia)}\big).$$

By the residue theorem, letting $N \to +\infty$, we obtain that

$$-V_h(f,a) = \int_{-\infty+i(d_-+\epsilon)}^{\infty+i(d_-+\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))} (\frac{e^{\frac{\pi}{h}a}}{z+i} - \frac{e^{\frac{\pi}{h}(a+1)}}{z} + \frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}) dz$$
$$- \int_{-\infty+i(d_+-\epsilon)}^{\infty+i(d_+-\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))} (\frac{e^{\frac{\pi}{h}a}}{z+i} - \frac{e^{\frac{\pi}{h}(a+1)}}{z} + \frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}) dz.$$

Here since $f \in H(\mathcal{D}_{(d_-,d_+)})$, we know $\int_{\gamma_2}$ and $\int_{\gamma_4}$ converge to zero as $N \to +\infty$. In the second

integral above, since

$$ie^{-\frac{\pi}{h}i(z+i(a+1))} = ie^{\frac{\pi}{h}i(z+i(a+1))} + i\big(e^{-\frac{\pi}{h}i(z+i(a+1))} - e^{\frac{\pi}{h}i(z+i(a+1))}\big),$$

after moving the term corresponding to $i\big(e^{-\frac{\pi}{h}i(z+i(a+1))} - e^{\frac{\pi}{h}i(z+i(a+1))}\big)$ to the left, we obtain the following on the left hand side:

$$\begin{aligned}&\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)} \frac{f(z)i\big(e^{-\frac{\pi}{h}i(z+i(a+1))} - e^{\frac{\pi}{h}i(z+i(a+1))}\big)}{2z(z+i)\sin(\frac{\pi}{h}(z+i(a+1)))}dz - V_h(f,a)\\ &= \int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)} \frac{f(z)}{z(z+i)}dz - V_h(f,a),\end{aligned}$$

which equals $E_h^V(f,a)$ since from the Cauchy integral theorem, we have (recall that $0 < -(a+1) < d_+ - \epsilon$)

$$\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)} \frac{f(z)}{z(z+i)}dz = \int_{-\infty-i(a+1)}^{+\infty-i(a+1)} \frac{f(z)}{z(z+i)}dz = V(f,a).$$

Let $\epsilon \to 0+$, we obtain the expression for $E_h^V(f,a)$.

Secondly, when $a = -1$, we consider the same contour integral. The residues at $\{mh, m \in \mathbb{Z}, m \neq 0\}$ have the same expression. The residue at mh for $m = 0$ can be computed to be

$$-\frac{hf(0)(-e^{-\frac{\pi}{h}} - \frac{\pi}{h} + 1)}{2\pi i} = -\frac{hf(0)(1-e^{-\frac{\pi}{h}})}{2\pi i} + \frac{f(0)\pi}{2\pi i}.$$

It is straightforward to show that:

$$\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)} \frac{f(z)}{z(z+i)}dz + f(0)\pi = p.v.\int_{\mathbb{R}} \frac{f(x-i(a+1))}{(x-ia)(x-i(a+1))}dx.$$

Thirdly, when $-1 < a < 0$ (or $-1 < -(a+1) < 0$), consider the following contour integral:

$$\int_\gamma \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}\Big(\frac{e^{\frac{\pi}{h}a}}{z+i} - \frac{e^{-\frac{\pi}{h}(a+1)}}{z} + \frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}\Big)dz,$$

where γ is defined in the same way above. The integrand has poles of order 1 at $\{mh - i(a+1), m \in \mathbb{Z}\}$ and $z = 0$, (note that $z = -i$ is removable singularities). The residue of the

integrand at $mh - i(a+1)$ can be computed easily as:

$$-\frac{hf(mh-i(a+1))}{2\pi i(mh-ia)(mh-i(a+1))}\big(1+i(mh-ia)e^{i\frac{\pi}{h}(mh+i(a+1))}-i(mh-i(a+1))e^{i\frac{\pi}{h}(mh-ia)}\big).$$

The residual at $z=0$ is given by:

$$Res(0)=\big[\exp(\frac{\pi}{h}(a+1))-\exp(-\frac{\pi}{h}(a+1))\big]\frac{f(0)}{2\sin(\frac{\pi}{h}(i(a+1)))}=-if(0).$$

By the residue theorem, letting $N\to+\infty$, we obtain that

$$\begin{aligned}-V_h(f,a)=&\int_{-\infty+i(d_-+\epsilon)}^{\infty+i(d_-+\epsilon)}\frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}(\frac{e^{\frac{\pi}{h}a}}{z+i}-\frac{e^{-\frac{\pi}{h}(a+1)}}{z}+\frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)})dz\\&-\int_{-\infty+i(d_+-\epsilon)}^{\infty+i(d_+-\epsilon)}\frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}(\frac{e^{\frac{\pi}{h}a}}{z+i}-\frac{e^{-\frac{\pi}{h}(a+1)}}{z}+\frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)})dz.\end{aligned}$$

Here we have used the fact that, from (2.1), the $\int_{\gamma_2}$ and $\int_{\gamma_4}$ terms in the contour integral converge to zero as $N\to+\infty$. In the second integral above, we note that

$$ie^{-\frac{\pi}{h}i(z+i(a+1))}=ie^{\frac{\pi}{h}i(z+i(a+1))}+i\big(e^{-\frac{\pi}{h}i(z+i(a+1))}-e^{\frac{\pi}{h}i(z+i(a+1))}\big).$$

Moving the term corresponding to $i\big(e^{-\frac{\pi}{h}i(z+i(a+1))}-e^{\frac{\pi}{h}i(z+i(a+1))}\big)$ to the left, we obtain the following plus $2\pi i Res(0)=2\pi f(0)$ on the left hand side:

$$\begin{aligned}&\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)}\frac{f(z)i\big(e^{-\frac{\pi}{h}i(z+i(a+1))}-e^{\frac{\pi}{h}i(z+i(a+1))}\big)}{2z(z+i)\sin(\frac{\pi}{h}(z+i(a+1)))}dz-V_h(f,a)\\&=\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)}\frac{f(z)}{z(z+i)}dz-V_h(f,a),\end{aligned}$$

which equals to $E_h^V(f,a)$. Since from the Cauchy integral theorem, we have (recall that $-1<-(a+1)<0$, and the residual of $\frac{f(z)}{z(z+i)}$ at $z=0$ is $-if(0)$.) :

$$\int_{-\infty+i(d_+-\epsilon)}^{+\infty+i(d_+-\epsilon)}\frac{f(z)}{z(z+i)}dz+2\pi i Res(\frac{f(z)}{z(z+i)},0)=\int_{-\infty+i(a+1)}^{+\infty+i(a+1)}\frac{f(z)}{z(z+i)}dz=V(f,a).$$

Letting $\epsilon\to 0+$, we obtain the expression for $E_h^V(f,a)$.

Fourthly, we consider the case where $0<a<-d_--1$ (that is, $d_-<-(a+1)<-1$). Consider

the following contour integral:

$$\int_\gamma \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}\Big(\frac{e^{-\frac{\pi}{h}a}}{z+i}-\frac{e^{-\frac{\pi}{h}(a+1)}}{z}+\frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}\Big)dz,$$

where γ is defined in the same way as in the proof of Theorem 2.2 for some $\epsilon > 0$ such that $d_- + \epsilon < -(a+1) < d_+ - \epsilon$ and integer $N > 0$. The integrand has poles of order 1 at $\{mh - i(a+1), m \in \mathbb{Z}\}$ (note that $z = 0$ and $z = -i$ are removable singularities). The residue of the integrand at $mh - i(a+1)$ can be computed easily as:

$$-\frac{hf(mh-i(a+1))}{2\pi i(mh-ia)(mh-i(a+1))}\big(1+i(mh-ia)e^{-i\frac{\pi}{h}(mh-i(a+1))}-i(mh-i(a+1))e^{-i\frac{\pi}{h}(mh-ia)}\big).$$

By the residue theorem, letting $N \to +\infty$, we obtain that:

$$\begin{aligned}-V_h(f,a)(f,a) = &\int_{-\infty+i(d_-+\epsilon)}^{\infty+i(d_-+\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}(\frac{e^{-\frac{\pi}{h}a}}{z+i}-\frac{e^{-\frac{\pi}{h}(a+1)}}{z}+\frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)})dz\\ &-\int_{-\infty+i(d_+-\epsilon)}^{\infty+i(d_+-\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}(\frac{e^{-\frac{\pi}{h}a}}{z+i}-\frac{e^{-\frac{\pi}{h}(a+1)}}{z}+\frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)})dz.\end{aligned}$$

Here we have used the fact that, from (2.1), the $\int_{\gamma_2}$ and $\int_{\gamma_4}$ terms in the contour integral converge to zero as $N \to +\infty$. In the first integral above, we note that:

$$ie^{\frac{\pi}{h}i(z+i(a+1))} = ie^{-\frac{\pi}{h}i(z+i(a+1))} - i\big(e^{-\frac{\pi}{h}i(z+i(a+1))} - e^{\frac{\pi}{h}i(z+i(a+1))}\big).$$

Moving the term corresponding to $-i\big(e^{-\frac{\pi}{h}i(z+i(a+1))} - e^{\frac{\pi}{h}i(z+i(a+1))}\big)$ to the left, we obtain the following on the left hand side:

$$\begin{aligned}&\int_{-\infty+i(d_-+\epsilon)}^{+\infty+i(d_-+\epsilon)} \frac{f(z)i\big(e^{-\frac{\pi}{h}i(z+i(a+1))} - e^{\frac{\pi}{h}i(z+i(a+1))}\big)}{2z(z+i)\sin(\frac{\pi}{h}(z+i(a+1)))}dz - V_h(f,a)\\ &= \int_{-\infty+i(d_-+\epsilon)}^{+\infty+i(d_-+\epsilon)} \frac{f(z)}{z(z+i)}dz - V_h(f,a),\end{aligned}$$

which equals to $E_h^V(f,a)$. Since from the Cauchy integral theorem, we have (recall that $-d_- + \epsilon < -(a+1) < -1$):

$$\int_{-\infty+i(d_-+\epsilon)}^{+\infty+i(d_-+\epsilon)} \frac{f(z)}{z(z+i)}dz = \int_{-\infty+i(a+1)}^{+\infty+i(a+1)} \frac{f(z)}{z(z+i)}dz = V(f,a).$$

Letting $\epsilon \to 0+$, we obtain the expression for $E_h^V(f,a)$.

In the last, when $a = 0$, we consider the same contour integral. The residues at $\{mh, m \in \mathbb{Z}, m \neq 0\}$ have the same expression. The residue at mh for $m = 0$ is:

$$\begin{aligned}
&\lim_{z\to -i}(z+i)\frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}\Big(\frac{e^{-\frac{\pi}{h}a}}{z+i}-\frac{e^{-\frac{\pi}{h}(a+1)}}{z}+\frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}\Big)\\
=&\lim_{z\to -i}\frac{\frac{\pi}{h}(z+i)f(z)\frac{h}{\pi}}{2\sin(\frac{\pi}{h}(z+i))}\Big[\frac{1}{z+i}-\frac{e^{-\frac{\pi}{h}}}{z}+\frac{ie^{\frac{\pi}{h}i(z+i)}}{z(z+i)}\Big]=\lim_{z\to -i}\frac{f(z)h}{2\pi}\Big[\frac{z-(z+i)e^{-\frac{\pi}{h}}+ie^{\frac{\pi}{h}i(z+i)}}{z(z+i)}\Big]\\
=&\lim_{z\to -i}\frac{f(z)h}{2\pi}\Big[\frac{(z+i)(1-e^{-\frac{\pi}{h}})+i(e^{\frac{\pi}{h}i(z+i)}-1)}{z(z+i)}\Big]=\frac{f(-i)h}{2\pi}\Big[\frac{1-e^{-\frac{\pi}{h}}}{-i}+\lim_{z\to -i}\frac{i(e^{\frac{\pi}{h}i(z+i)}-1)}{z(z+i)}\Big]\\
=&\frac{f(-i)h}{2\pi}\Big[\frac{1-e^{-\frac{\pi}{h}}-\frac{\pi}{h}}{-i}\Big]=-\frac{hf(-i)(1-e^{-\frac{\pi}{h}})}{2\pi i}+\frac{f(-i)\pi}{2\pi i}.
\end{aligned}$$

Since the residual of $\frac{f(z)}{z(z+i)}$ at $-i$ is $\frac{f(-i)}{-i}$, it is straightforward to show that

$$\int_{-\infty+i(d_-+\epsilon)}^{+\infty+i(d_-+\epsilon)}\frac{f(z)}{z(z+i)}dz+f(-i)\pi=p.v.\int_{\mathbb{R}}\frac{f(x-i(a+1))}{(x-ia)(x-i(a+1))}dx.$$

Hence, after discussing all five situations above, we are able to draw the conclusion.

□

6.1.3 Error bound for discretization error

Once we obtain the explicit discretization error, we are able to estimate the upper bound of it for further numerical computation purpose. And the result is shown in the following theorem.

Theorem 6.1.2. *Suppose $f \in H(\mathcal{D}_{(d_-,d_+)})$. Then for any $-(a+1) \in (d_-, d_+)$, we have an exponentially decaying error bound in terms of $1/h$:*

$$|E_h^V(f,a)| \leq \frac{2e^{-\frac{\pi}{h}A(a)}}{(-1-d_-)(1-e^{\frac{2\pi}{h}(d_-+a+1)})}\|f\|^- + \frac{2e^{-\frac{\pi}{h}B(a)}}{d_+(1-e^{-\frac{2\pi}{h}(d_++a+1)})}\|f\|^+$$

$A(a), B(a)$ are positive (piecewise) linear functions of a. When $0 \leq -(a+1) < d_+$, $A(a) = |d_- + 2(a+1)|$, $B(a) = \min(d_+, 2(d_+ + a + 1))$; when $-1 < -(a+1) < 0$, $A(a) = \min(-d_- - 2a - 1, -d_-)$, $B(a) = \min(d_+ + 1, d_+ + 2a + 2)$; when $d_- < -(a+1) \leq -1$, $A(a) = \min(-d_- - 1, -2(d_- + a + 1))$, $B(a) = d_+ + 2a + 1$.

Proof. Here, we only prove for $-(a+1) < d_+$ or $-d_+ - 1 < a \leq -1$. For the rest situations, the error bound could be obtained in the same way.

To begin with, let us consider the first integral in (6.4). Assume $z = t + id_-$, then the following

hold:

$$|\frac{e^{-\frac{\pi}{h}|a|}}{z+i}| \leq \frac{e^{-\frac{\pi}{h}|a|}}{-1-d_-}.$$
$$|\frac{e^{-\frac{\pi}{h}|a+1|}}{z}| \leq \frac{e^{-\frac{\pi}{h}|a+1|}}{-d_-}.$$
$$|\frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}| = |\frac{ie^{-i\frac{\pi}{h}(t+id_-+i(a+1))}}{z(z+i)}| \leq \frac{e^{\frac{\pi}{h}(d_-+a+1)}}{-d_-(-1-d_-)}.$$

Then we combine these three terms together to obtain:

$$\begin{aligned}
|\frac{e^{-\frac{\pi}{h}|a|}}{z+i} - \frac{e^{-\frac{\pi}{h}|a+1|}}{z} + \frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}| &\leq \frac{e^{-\frac{\pi}{h}|a|}}{-1-d_-} + \frac{e^{-\frac{\pi}{h}|a+1|}}{-d_-} + \frac{e^{\frac{\pi}{h}(d_-+a+1)}}{-d_-(-1-d_-)} \\
&= \frac{-d_-e^{\frac{\pi}{h}a} + (-1-d_-)e^{\frac{\pi}{h}(a+1)} + e^{\frac{\pi}{h}(d_-+a+1)}}{-d_-(-1-d_-)} \\
&\leq \frac{-d_-e^{\frac{\pi}{h}(a+1)} + (-1-d_-)e^{\frac{\pi}{h}(a+1)} + e^{\frac{\pi}{h}(a+1)}}{-d_-(-1-d_-)} \\
&= \frac{-2d_-e^{\frac{\pi}{h}(a+1)}}{-d_-(-1-d_-)} = \frac{2e^{\frac{\pi}{h}(a+1)}}{-1-d_-} = \frac{2e^{-\frac{\pi}{h}|a+1|}}{-1-d_-}.
\end{aligned}$$

On the other hand,

$$\begin{aligned}
|2\sin(\frac{\pi}{h}(z+i(a+1)))| &= |e^{i(\frac{\pi}{h}(z+i(a+1)))} - e^{-i(\frac{\pi}{h}(z+i(a+1)))}| \\
&= |e^{i(\frac{\pi}{h}(t+id_-+i(a+1)))} - e^{-i(\frac{\pi}{h}(t+id_-+i(a+1)))}| \\
&= |e^{i\frac{\pi}{h}t-\frac{\pi}{h}(d_-+(a+1))} - e^{-i\frac{\pi}{h}t+\frac{\pi}{h}(d_-+(a+1))}| \\
&\geq e^{-\frac{\pi}{h}(d_-+a+1)} - e^{\frac{\pi}{h}(d_-+a+1)}.
\end{aligned}$$

Hence,

$$\begin{aligned}
&\Big| \lim_{\epsilon\to 0+} \int_{-\infty+i(d_-+\epsilon)}^{\infty+i(d_-+\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}(\frac{e^{-\frac{\pi}{h}|a|}}{z+i} - \frac{e^{-\frac{\pi}{h}|a+1|}}{z} + \frac{ie^{-i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)})dz\Big| \\
&\leq \frac{2e^{\frac{\pi}{h}(a+1)}}{(-1-d_-)(e^{-\frac{\pi}{h}(d_-+a+1)} - e^{\frac{\pi}{h}(d_-+a+1)})}\|f\|^- = \frac{2e^{\frac{\pi}{h}(d_-+2(a+1))}}{(-1-d_-)(1-e^{\frac{2\pi}{h}(d_-+a+1)})}\|f\|^- \\
&= \frac{2e^{-\frac{\pi}{h}|d_-+2(a+1)|}}{(-1-d_-)(1-e^{\frac{2\pi}{h}(d_-+a+1)})}\|f\|^-.
\end{aligned}$$

Then for the second integral, assume $z = t + id_+$, then we have:

$$
\begin{aligned}
&|\frac{e^{-\frac{\pi}{h}|a|}}{z+i}| \le \frac{e^{-\frac{\pi}{h}|a|}}{1+d_+}.\\
&|\frac{e^{-\frac{\pi}{h}|a+1|}}{z}| \le \frac{e^{-\frac{\pi}{h}|a+1|}}{d_+}.\\
&|\frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}| = |\frac{ie^{i\frac{\pi}{h}(t+id_++i(a+1))}}{z(z+i)}| \le \frac{e^{-\frac{\pi}{h}(d_++a+1)}}{d_+(1+d_+)}.
\end{aligned}
$$

And just like the previous case, we can obtain:

$$
\begin{aligned}
|\frac{e^{-\frac{\pi}{h}|a|}}{z+i} - \frac{e^{-\frac{\pi}{h}|a+1|}}{z} + \frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)}| &\le \frac{e^{-\frac{\pi}{h}|a|}}{1+d_+} + \frac{e^{-\frac{\pi}{h}|a+1|}}{d_+} + \frac{e^{-\frac{\pi}{h}(d_++a+1)}}{d_+(1+d_+)}\\
&= \frac{d_+e^{\frac{\pi}{h}a} + (1+d_+)e^{\frac{\pi}{h}(a+1)} + e^{-\frac{\pi}{h}(d_++a+1)}}{d_+(1+d_+)}\\
&\le \frac{d_+e^{\frac{\pi}{h}d_a} + (1+d_+)e^{\frac{\pi}{h}d_a} + e^{\frac{\pi}{h}d_a}}{d_+(1+d_+)}\\
&= \frac{2(1+d_+)e^{\frac{\pi}{h}d_a}}{d_+(1+d_+)} = \frac{2e^{\frac{\pi}{h}d_a}}{d_+} = \frac{2e^{-\frac{\pi}{h}|d_a|}}{d_+},
\end{aligned}
$$

where $d_a = \max(a+1, -d_+ - a - 1)$. On the other hand,

$$
\begin{aligned}
|2\sin(\frac{\pi}{h}(z+i(a+1)))| &= |e^{i(\frac{\pi}{h}(z+i(a+1)))} - e^{-i(\frac{\pi}{h}(z+i(a+1)))}|\\
&= |e^{i(\frac{\pi}{h}(t+id_++i(a+1)))} - e^{-i(\frac{\pi}{h}(t+id_++i(a+1)))}|\\
&= |e^{i\frac{\pi}{h}t-\frac{\pi}{h}(d_++(a+1))} - e^{-i\frac{\pi}{h}t+\frac{\pi}{h}(d_++(a+1))}|\\
&\ge e^{\frac{\pi}{h}(d_++a+1)} - e^{-\frac{\pi}{h}(d_++a+1)}.
\end{aligned}
$$

Hence,

$$
\begin{aligned}
&\Big|\lim_{\epsilon\to 0+}\int_{-\infty+i(d_++\epsilon)}^{\infty+i(d_++\epsilon)} \frac{f(z)}{2\sin(\frac{\pi}{h}(z+i(a+1)))}(\frac{e^{-\frac{\pi}{h}|a|}}{z+i} - \frac{e^{-\frac{\pi}{h}|a+1|}}{z} + \frac{ie^{i\frac{\pi}{h}(z+i(a+1))}}{z(z+i)})dz\Big|\\
&\le \frac{2e^{\frac{\pi}{h}d_a}}{(d_+)(e^{\frac{\pi}{h}(d_++a+1)} - e^{-\frac{\pi}{h}(d_++a+1)})}\|f\|^+ = \frac{2e^{\frac{\pi}{h}(d_a-d_+-a-1)}}{(d_+)(1-e^{-\frac{2\pi}{h}(d_++a+1)})}\|f\|^+\\
&= \frac{2e^{-\frac{\pi}{h}d'_a}}{d_+(1-e^{-\frac{2\pi}{h}(d_++a+1)})}\|f\|^+, \qquad d'_a = \min(d_+, 2(d_++a+1)).
\end{aligned}
$$

Therefore,

$$
|E_h^V(f,a)| \le \frac{2e^{-\frac{\pi}{h}|d_-+2(a+1)|}}{(-1-d_-)(1-e^{\frac{2\pi}{h}(d_-+a+1)})}\|f\|^- + \frac{2e^{-\frac{\pi}{h}d'_a}}{d_+(1-e^{-\frac{2\pi}{h}(d_++a+1)})}\|f\|^+.
$$

□

Notice that for $1 - e^{\frac{2\pi}{h}(d_- + a + 1)}$ and $1 - e^{-\frac{2\pi}{h}(d_+ + a + 1)}$ in the denominators, when we choose h relatively small, we can bound them easily. For example, when $a \leq -1$, $d_- \leq -1$, if we take $h \leq 2\pi$, we can bound $1 - e^{\frac{2\pi}{h}(d_- + a + 1)}$ by $1 - 1/e$. Then we can choose a and h properly to control the numerical error to be less than pre-selected tolerance level.

6.2 Pricing Bermudan options

For calibration purpose, we usually need to price tens or hundreds of options together. In this part, we propose a new numerical scheme based on the original work in [39] to price Bermudan options with different strike prices and maturities simultaneously.

6.2.1 Bermudan options

Different from European option which could only be exercised at maturity, American option has early exercise feature. That is, the owner of American style options has the right to exercise the option at certain pre-determined time slots or time periods. Theoretically, American options could be exercised at any time before or at option maturity. As a discrete type of American style option, Bermudan option can be exercised only on a discrete time set $\mathfrak{T} = \{t_0, t_1, ..., t_N\}$. Typically, $t_0 = 0$, $t_N = T$, the option maturity. In the following part, we assume $t_{j+1} - t_j = \Delta$ are equally spaced. And the method could be easily extended to more general situations.

Here we still illustrate by considering put options. If we denote the strike price by K, and the underlying asset price at time t by S_t, then Bermudan put option price at time 0 is given by:

$$V(S_0) = \sup_{\tau} \mathbb{E}[e^{-r\tau}(K - S_\tau)^+], \tag{6.5}$$

where stopping time τ is in $\mathfrak{T}$. We will first introduce the method proposed in [39] in the following section. And then we will extend their method to price multiple options simultaneously in later section.

6.2.2 Valuation of Bermudan options

Here we briefly introduce the method proposed by Feng and Lin in [39]. Please refer to their original paper for more details. As mentioned by Feng and Lin, the pricing problem (6.5) is essentially an optimal stopping problem, which could be solved by backward induction. Because the underlying

asset price follows geometric Lévy process, they first conduct a variable change by setting $X_t = \ln(S_t/K)$. Then if denote the payoff function at each discrete time spots $t_j = j\Delta$ by $f^j(x)$, for $j = N, ..., 0$ and define $g(x) = (K - Ke^x)^+$, then for $j = N$, the payoff function of put option is given by $f^N(x) = g(x)$, and the rest f^j are defined recursively as:

$$f^j(x) = \max(g(x), e^{-r\Delta}\mathbb{E}_{j\Delta,x}[f^{j+1}(X_{(j+1)\Delta})]), \quad 0 \le j < N,$$

where $\mathbb{E}_{j\Delta,x}$ is the conditional expectation of $X_{j\Delta} = x$. And the option price is then given by $V(S_0) = f^0(X_0)$. This problem is indeed solved in the Fourier space in order to save computational cost. And to guarantee integrability, damping factor $e^{\alpha x}$ is introduced. They defined damping version of f and g by $f^j_\alpha(x) = e^{\alpha x} f^j(x)$ and $g_\alpha(x) = e^{\alpha x} g(x)$. Then by the convolution theorem:

$$\mathcal{F}\Big(\mathbb{E}^\alpha_{j\Delta,x}[f^{j+1}_\alpha(X_{(j+1)\Delta})]\Big) = \mathcal{F}\Big(\int_{\mathbb{R}} f^{j+1}_\alpha(y) p^\alpha_\Delta(x-y)dy\Big) = \hat{f}^{j+1}_\alpha(\xi)\phi_\Delta(-\xi + i\alpha).$$

Also, by taking advantage of the following key identity of Hilbert transform:

$$\mathcal{F}(\infty_{(-\infty,x^*_j)} \cdot h)(\xi) = \frac{1}{2}\hat{h}(\xi) - \frac{i}{2}e^{i\xi x^*_j}\mathcal{H}(e^{-i\eta x^*_j}\hat{h}(\eta))(\xi)$$

their algorithm in the Fourier space can be written as:

$$\hat{f}^N_\alpha(\xi) = \hat{g}_\alpha(\xi), \tag{6.6a}$$

$$\text{find } x^*_j \text{ such that: } g_\alpha(x) = \frac{1}{2\pi}e^{-r\Delta}\int_{\mathbb{R}} e^{-i\xi x}\hat{f}^{j+1}_\alpha(\xi)\phi_\Delta(-\xi + i\alpha)d\xi. \tag{6.6b}$$

$$\begin{aligned}\hat{f}^j_\alpha(\xi) =& \mathcal{F}(g_\alpha \cdot \mathbf{1}_{(-\infty,x^*_j]})(\xi) + e^{-r\Delta}\Big(\frac{1}{2}\hat{f}^{j+1}_\alpha(\xi)\phi_\Delta(-\xi + i\alpha) \\ &+ \frac{i}{2}e^{i\xi x^*_j}\mathcal{H}(e^{-i\eta x^*_j}\hat{f}^{j+1}_\alpha(\eta)\phi_\Delta(-\eta + i\alpha))(\xi)\Big), \quad 0 \le j < N\end{aligned} \tag{6.6c}$$

$$f^0_\alpha(x) = \max\Big(g_\alpha(x), \frac{1}{2\pi}e^{-r\Delta}\int_{\mathbb{R}} e^{-\xi x}\hat{f}^1_\alpha(\xi)\phi_\Delta(-\xi + i\alpha)d\xi\Big). \tag{6.6d}$$

First they compute (6.6a). Then, the early exercise boundary x^*_j can be obtained by solving (6.6b). And then, the computation goes back one step from N to $N-1$ by solving (6.6c). And the whole process will be repeated for $N-1$ times to get $\hat{f}^1_\alpha(x)$. The option price at time 0 is then given by $V^0(S_0) = e^{-\alpha \ln(S_0/K)} f^0_\alpha(\ln(S_0/K))$.

6.2.3 Simultaneous computing of multiple Bermudan vanilla options price

The above method is good enough if we only need to compute single option price given certain parameters. However, since our goal is to calibrate American vanilla (Bermudan daily monitored) options, for which we need to compute tens or even hundreds of option prices at one time. In that case, the computing time would be multiplied by a factor of the number of options, which would be very undesirable. Hence, it is ideal if we can modify the above method so that it provides us with prices of multiple options, which written upon the same underlying asset but have different strike prices and maturities. For example, it is well known that fast Fourier transform method could be applied to compute option prices with different initial underlying asset prices S, see [16] for more detail. We expect to have similar effect here, but instead for different initial underlying asset prices S, we need different strikes K and maturities T. Similar example is Dupire type equations, which could be applied in computing price of options with different strikes and maturities through solving numerical PDE. However, it only works in local volatility model. Here, luckily, according to the special structure of this backward induction and also the homogeneity property of Lévy processes, we are able to achieve this goal with one run of backward induction. We introduce the detail as following.

First, we consider different *strikes*. Recall that in the backward reduction (6.6), the whole process is kept in the Fourier space, except the last step where we compute option price. In implementation, we only need to discretize in the Fourier space. Let ξ denote the Fourier space variable and let x denote the variable of natural state space. In the first $N-1$ time step of the backward induction, we do not use x at all. Only discretization in the Fourier space is needed, say $\xi = \{-Mh, \cdots, Mh\}$, and keep the computing process in the Fourier space. After that, we obtain $\hat{f}^1_\alpha(\xi)$, where $\xi = \{-Mh, \cdots, Mh\}$. And the option price could be obtained by the discretization form of (6.6d), which is:

$$\begin{aligned} f^0_\alpha(x) &= \max\Big(g_\alpha(x), \frac{1}{2\pi}e^{-r\Delta}\int_{\mathbb{R}} e^{-\xi x}\hat{f}^1_\alpha(\xi)\phi_\Delta(-\xi+i\alpha)d\xi\Big) \\ &\approx \max\Big(g_\alpha(x), \frac{1}{2\pi}e^{-r\Delta}\sum_{k=-M}^{M} e^{-ikhx}\hat{f}^1_\alpha(kh)\phi_\Delta(-kh+i\alpha)h\Big). \end{aligned}$$

Since $x = \ln(S_0/K)$ here, by putting in different x in the above equation, we can obtain different option prices which have same maturity but different initial underlying asset price. Notice that the extra workload increased here in the last backward induction step to compute multiple option prices with different initial underlying asset prices takes a very small part in the total computational time,

comparing to the time taken in the whole backward induction process, and can even be ignored. To make the statement easier, assume here we have a vector $\mathbf{x} = (x_1, x_2, \cdots, x_n)$. Then the option prices can also be written in a vector form:

$$\mathbf{F}_\alpha(\mathbf{x}) = \begin{pmatrix} f_\alpha^0(x_1) \\ f_\alpha^0(x_2) \\ \vdots \\ f_\alpha^0(x_n) \end{pmatrix} = \max\Big(g_\alpha(\mathbf{x}), \frac{1}{2\pi} e^{-r\Delta} \sum_{k=-M}^{M} e^{-ikh\mathbf{x}} \hat{f}_\alpha^1(kh)\phi_\Delta(-kh + i\alpha)h\Big). \tag{6.7}$$

Recall that put option price with initial underlying asset price S_0, maturity T and strike price K is given by:

$$V(S_0, T, K) = \sup_\tau \mathbb{E}[e^{-r\tau}(K - S_\tau)^+].$$

Here option strike K is a constant, hence, we can make a little modification and write it as:

$$V(S_0, T, K) = K \sup_\tau \mathbb{E}[e^{-r\tau}(1 - \frac{S_\tau}{K})^+] = KV(\frac{S_0}{K}, T, K'). \tag{6.8}$$

Here $V(S_0/K, T, K')$ gives a put option price with initial underlying asset price S_0/K and strike price $K' = 1$. Assume we have several maturities and write it as an vector $\mathbf{K} = (K_1, K_2, \cdots, K_n)$. Then, by (6.8), we can obtain the following equation:

$$\mathbf{V}(S_0, T, \mathbf{K}) = \begin{pmatrix} V(S_0, T, K_1) \\ V(S_0, T, K_2) \\ \vdots \\ V(S_0, T, K_n) \end{pmatrix} = \begin{pmatrix} K_1 V(S_0/K_1, T, 1) \\ K_2 V(S_0/K_2, T, 1) \\ \vdots \\ K_n V(S_0/K_n, T, 1) \end{pmatrix} = \mathbf{K} \cdot V(\mathbf{S}, T, 1),$$

where $\cdot$ is for vector dot product and $\mathbf{S} = (S_0/K_1, S_0/K_2, \cdots, S_0/K_n)$ is a vector. Then we obtain the following relationship:

$$\mathbf{V}(S_0, T, \mathbf{K}) = \mathbf{K} \cdot V(\mathbf{S}, T, 1) = \mathbf{K} \cdot \mathbf{F}(\mathbf{x}).$$

Here $\mathbf{F}(\mathbf{x})$ can be obtained through $\mathbf{F}_\alpha(\mathbf{x})$ divided by the damping factor $e^{\alpha\mathbf{x}}$. Since we already know how to compute $\mathbf{F}_\alpha(\mathbf{x})$ in one run of the backward reduction, then the put option price here with different strikes can be obtained easily by a dot product of $\mathbf{K}$ and $\mathbf{F}(\mathbf{x})$. ($\mathbf{F}(\mathbf{x})$ is a short

notation of $\mathbf{F}(\mathbf{x}, T, K' = 1)$, which is equal to $V(\mathbf{S}, T, 1)$.)

Then, we consider how to fast compute option prices with different *maturities*. Here we take advantage of the homogeneity property of Lévy process. In fact, we have already used this property in deducing the backward induction process (6.6). In the computing process, the real time t (or stopping time $\tau = T - t$) never appears, instead, what matters is the length of the time step Δ in the backward reduction. When we compute an option with maturity of 1 year. And assume that we use daily monitoring, i.e. $\Delta = 1/252$. As we introduced before, the option price with 1 year maturity can be obtained by (6.6d) after we have the value of $\hat{f}_\alpha^1$ in the backward reduction. On the other hand, in the middle of the backward induction, we obtained $\hat{f}_\alpha^{127}$ as a byproduct. In this case, if putting this value into the same equation (6.6d), we are able to get the value of option price with maturity of half year (or 6 months).

Moreover, assume we have several maturities $\mathbf{T} = (T_1, T_2, \cdots, T_m)$, $T_i < T_j$ if $i < j$. In the backward induction process of computing option price with maturity T_m, we are able to derive all other option prices with maturities $T_1, T_2, \cdots, T_{m-1}$, and all these computation could be done in one run of backward induction. Also, this method could be easily combined with the method introduced before to obtain multiple option prices with different strikes. Therefore, we are able to compute multiple option prices with different strikes and maturities by one run of the backward induction. This saves us a huge amount of time in model calibration. For example, if we take $K = (50, 52, \cdots, 150)$, and $T = (20, 50, 100, 126, 252, 400)$ (days), which is 306 options in total. Instead of the time cost for computing a single option price multiplied by 306, we can complete the computation within 5 times of single pricing time.

Remark 6.2.1. *There are also problems and restrictions with this approach. By the Lévy -Khintchine theorem, the c.f function of X_t has the following form:*

$$\phi_t(\xi) = \mathbb{E}[e^{i\xi X_t}] = \exp\{-t(\frac{1}{2}\sigma^2\xi^2 - i\mu\xi + \int_{\mathbb{R}}(1 - e^{i\xi y} + i\xi y\mathbf{1}_{\{|y|\leq 1\}}\Pi(dy)))\}.$$

where t is replaced by a fixed constant time step Δ in our deduction before. And here:

$$\mu = r - q - \frac{1}{2}\sigma^2 + \int_{\mathbb{R}}(1 - e^y + y\mathbf{1}_{\{|y|\leq 1\}})\Pi(dy).$$

In the previous computation of multiple option prices, we take interest rate r and dividend yield q as a constant. Otherwise, if r and q are functions of time t, the time homogeneity property would not be valid any more. However, in real world market, interest rate and dividend yield are usually not

constants and usually vary a lot for different maturities. For example, in Apr. 2010, the 1 month LIBOR rate is 0.26, 3 months rate is 0.31 while 1 year LIBOR rate is 0.94. Therefore, it is not very persuasive to take interest rate or dividend yield as a constant. In this case, we cannot use the time-homogeneity property to fast compute option prices with different maturities. On the other hand, it is still effective and efficient to use the method to compute multiple option prices with different strike prices, especially considering the fact that options usually have tens of different strikes but only couple of maturities. And in certain conditions, we can still use the time homogeneity property. For example, if we only consider options with short maturities, i.e. 1 month, 2 months, 3 months. Then the interest rate would basically be the same, and it is not very controversial to take the average interest rate as the constant.

6.2.4 Approximating American option price by Bermudan option price

According to the definition, Bermudan option is in fact the discrete version of American option. Although it is not proved theoretically, a preliminary analysis and intensive numerical study shows that the convergence rate is approximately $O(1/N)$, where N is the number of monitoring intervals. Here we use this method to compute American option prices. The Richardson extrapolation can be applied to speed up the convergence. More specifically, given two approximations P_1 (with N_1 monitoring intervals) and P_2 (with N_2 monitoring intervals) to the American vanilla put price (denote by P_∞), we have the following extrapolated value as a new approximation:

$$P_\infty \approx \frac{N_1 P_1 - N_2 P_2}{N_1 - N_2}. \tag{6.9}$$

There are three ways to do the extrapolation:

- Daily monitoring and half daily monitoring.

  This is a comparatively slower way but easier to understand. That is, we take N_1 to be the longest maturity we have in the maturity set $\mathbf{T}$. And take $N_2 = 2N_1$. In this case, we have desired accuracy, daily monitoring is very accurate for approximating American option price. However, it is too accurate for our calibration purpose (two digit after the decimal point.), especially for those long maturities. And the accuracy has a tradeoff of more computation load or longer computation time.

- Use fixed N for all options with different maturities.

  Here we compute option prices with different maturities separately. For each maturity, we

use same N_1 and N_2. And their value is chosen through intensive numerical experiments, to see whether they can provide ideal accuracy of option price. According to our experiment, when $N1 = 10$, $N2 = 5$, we can obtain most option prices with a percentage error (absolute error/option price) of 0.001 for different maturities, strikes even most of different parameters in our model. Since N_1 and N_2 are relative small, this way is much faster than the first one.

- Interpolation way.

  This way is very much alike the second way. Assuming we have the maturity set $\mathbf{T} = (T_1, T_2, \cdots, T_m)$ and $T_i < T_j$ if $i < j$. We run one backward induction to get all these prices for Bermudan options. In the backward induction, the first group of option prices we obtain has maturity T_1. While doing this, we divide the interval $(0, T_1)$ into N_1 subintervals. Later, when we do the backward induction in interval (T_1, T_2), we also divide the interval into N_1 subintervals and do the rest similarly. And then we proceed the whole backward induction one more time with N_1 replaced by N_2. And use Richardson extrapolation to obtain American option prices. Notice that for different backward induction intervals, we use same N, which basically means different subinterval length Δ. Hence, this method is in fact an interpolation method and only valid if we are able to prove the convergence of Bermudan option price to American option price. This way is the most fast one among all three ways.

6.3 Model Calibration

6.3.1 Inverse optimization

After we are able to price multiple options efficiently, now we can talk about the inverse problem, which is in fact an optimization problem. For the purpose of illustration, we consider the calibration of Kou's jump diffusion model. In this model, the Lévy density is:

$$\nu(z) = \lambda[p\eta_1 e^{-\eta_1 z}\mathbf{1}_{(z>0)} + (1-p)\eta_2 e^{\eta_2 z}\mathbf{1}_{(z<0)}].$$

Denote the set of parameters in this model by $\Theta = \{\sigma, \lambda, p, \eta_1, \eta_2\}$. We have five parameters in total to calibrate. Suppose prices of I options are available, let $P^{Mkt}(T_i, K_i)$ denote the market price of these options, and $P^{\Theta}(T_i, K_i)$ denote the corresponding option price under parameter set Θ. Then

the inverse optimization problem is formulated as following:

$$\Theta^* = \arg\inf_{\Theta\in\Omega} \sum_{i=1}^{I} \omega_i \left| \frac{P^{\Theta}(T_i, K_i) - P^{Mkt}(T_i, K_i)}{P^{Mkt}(T_i, K_i)} \right|^2 + R(\Theta). \tag{6.10}$$

where Ω is the parameter set space, which is usually a box in the variable space. ω_i is the weight for the ith option pricing difference, which represents the confidence of estimating individual data, usually determined by the liquidity of that option, i.e. volume of open interest or trading volume. And $R(\Theta)$ is the regularization term, which we added in order to change the well know ill-posed optimization problem into a convex optimization problem with good property. It is very important to notice that the objective function is a very sophisticated function of parameters which we want to calibrate. There are many computations involved such as FFT, matrix computations and iterations. Hence, it is impossible to pursue the explicit gradient of the objective function. Therefore, most of our optimization algorithms are in the category of Derivative Free Optimization (DFO).

6.3.2 The optimization problem without penalization

First we consider the calibration problem (6.10) without the penalization term $R(\Theta)$. Thus the problem is indeed a nonlinear least square problem which intends to reduce the quadratic pricing error. (See [2].) The advantage of the calibration problem is that, without the intervention of the regularization term, all information shown in the calibrated parameter Θ^* comes from the market data. On the other hand, the objective function is non-convex, so a gradient method may rashly result in a local minimum with objective value much higher than our expected fitting error, while leaving wide variable space unexplored. Also, if a global optimization method is used in order to obtain a "global minimum"(a local minimum which provides ideal objective value), no matter using heuristics methods or stochastic optimization methods, the computation time would be too large for practical use.

In order to solve this dilemma, we propose a two-stage calibration procedure. The first stage solves (6.10) without penalty, through a global optimization method, which would take a long time from several hours to around a day, depending on the dimension of calibration model and starting point. While the reward for long computing time is that we can obtain an ideal parameter set. Then, in the second stage, we use the parameters we obtained in stage one as starting point and solve (6.10) with penalty by DFO or traditional gradient method combining certain gradient deriving technique. Since the dimension of the convex optimization problem (6.10) is very small, (less than or equal to

five in Lévy process models and around ten in SVCJ model), gradient method would give the result within 5 to 10 minutes normally (more detail in later section). Hence, this method can be applied to daily or even hourly model calibration in real world market. In this part, we first introduce a global optimization method. And we put the local optimization methods in the next subsection.

There are many global optimization methods available. Most popular ones include stochastic optimization like simulated annealing and heuristic algorithms like evolutionary algorithm. In 1995, inspired by social behavior of animals, Eberhart, R. C. and Kennedy proposed a new heuristic algorithm, called Particle Swarm Optimization (PSO, see [33]). This method is getting more and more popular recently, together with its variant, being used in different areas like electric engineering, neural net work and economics. In 2007, A.I.F. Vaz and L.N.Vicente [67] incorporated PSO into the framework of Pattern-Search, which is a classic Derivative Free optimization method (For more information about pattern search, please see [6]). And they developed a hybrid global optimization method, particle swarm pattern search. It is proved to be more efficient for solving certain difficult problems than traditional methods like simulated annealing and genetic algorithm.

The two step pattern search framework includes the global search as the first step and local search as the second step. Normally, the global search is conducted on a mesh built by a spanning set in the variable space. The hybrid method incorporates PSO into this step. The second step for local search is unchanged.

6.3.3 The convex optimization problem

After we solve the non-convex optimization problem for one time, we have sets of parameters which give a very good fitting of the market data. And whenever we want to update the parameters which incorporate new market data, we can solve (6.10) with penalty, and using the parameters we obtained as starting point. Assume we have a penalty term $R(\Theta)$ added, it seems that (6.10) is easy to solve since it is convex already, and any gradient decent method would apply. However, we have one more issue left: although we are able to fast compute the function value we want to minimize, we are unable to obtain the gradient of this function yet. Recall how we obtain the American option price introduced before: lots of numerical computations are involved, like interpolation, fast Fourier transform, etc... It is inapplicable to obtain the gradient analytically. So we need to pursue numerical methods: either getting the gradient numerically or avoiding the gradient based optimization methods and seeking gradient free optimization methods. We will show later in numerical results that either way works and be able to finish the optimization problem

within ideal amount of time.

Compute gradient numerically

- *Finite difference (FD).* This is a well known method and introduced in every numerical analysis book. If we have a function $F : \mathbf{R}^n \to \mathbf{R}$, and a point $x \in \mathbf{R}^n$, then we can add a perturbation h to x and observe the function value change. According to Taylor's theorem, we have:

$$\frac{\partial F}{\partial x_i} \approx \frac{F(x + he_i) - F(x)}{h}$$

  where e_i is the ith unit vector, the vector has 1 as the ith element and 0 for all other elements. Here we need to consider two errors incurred. First one is the truncation error, $|Err_1| \leq Mh/2$, where M is the bound for $|F''(t)|$, t is in a small enough ball centering at x. The second one is the rounding error, $|Err_2| \leq 2\epsilon/h$, where ϵ is the bound for function value (machine) error. When the total error is minimized, we have $h \approx 2\sqrt{\epsilon/M}$. (See [58] for more detail.)

- *Automatic Differentiation (AD).* This is a technique used to calculate the derivative of certain function numerically provided the program code of the function. The derivative calculation is broken down into basic derivative calculation, i.e. plus, minus, multiply, dividend and then use chain rule to combine them. The advantage of AD compared to finite difference is that, the AD technique usually gives the derivative as the same accuracy ϵ of the original function, while the finite difference can only provide the accuracy of around $\sqrt{\epsilon}$. Of course, the advantage comes with its payoff. Finite difference is a lot easier to implement than automatic differentiation, and the speed of finite difference is always for AD to catch up. AD techniques can be categorized by forward accumulation and reverse accumulation. Also, the implementation of AD techniques can be categorized by Source Code Transformation(SCT) and Operator Overloading(OO). The former one automatically generates new source code computing both function value and its derivatives. While the latter one keeps the original source code unchanged but overloads the objects of real number and basic mathematical operators. Packages using SCT technique include ADIFOR, TAPENADE. While packages MAD and ADOL-C use Operator Overloading.

Once we obtained the gradient, the rest work could be done easily by applying gradient-based optimization algorithms. Certain algorithms include trust region reflective algorithm [21], interior point method [13], and active set algorithm using sequential quadratic programming (SQP) [59].

Those gradient-based optimization algorithms are sophisticated and already been used widely in practice.

Derivative free optimization

Derivative free optimization(DFO) methods have been used in practical applications where the derivative is unable or expensive to get due to the special structure of original function. Although it is not a necessity to use derivative free optimization method while the FD and AD are available in our case, we still have it in our method list for comparison purpose, since we do not know whether the possible noise incurred by function evaluation would cause unreliable issues in our calibration. One popular DFO method is pattern search, which we have already introduced in the previous section. Here we introduce another DFO method, the trust-region interpolation based method which also called model based method [10][58].

Trust-Region method is a classic method used heavily in optimization. The idea is that, at current iterate, we choose a region within which we trust that a merit function (quadratic usually) is a very good approximation of the objective function. Then we minimize the quadratic function in the region instead of the original objective function.(See [58] for more detail.) According to Taylor's theorem, the following function would be a good choice to approximate original objective function $f(x)$ at point $x_k + p$:

$$f(x_k + p) \approx m_k(x_k + p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T \nabla^2 f(x_k) p.$$

However, we do not know the gradient and Hessian here. So, we take ∇f and $\nabla^2 f$ as unknown vector and matrix variables here. Since we are able to compute the function value of $f(x)$, we can then use those function value to approximate ∇f and $\nabla^2 f$. Assuming g_k and H_k approximates ∇f and $\nabla^2 f$ at point x_k. If the dimension of the variable space is n, i.e. $x \in \mathbb{R}^n$, then we can choose $N = \frac{1}{2}(n+1)(n+2)$ sample points, x_k^j, $j = 1, 2, ...N$. And solve the linear system $c + g_k^T p + \frac{1}{2} p^T H_k p = f(x_k^j)$, for $j = 1, 2, ...N$. Since H_k is an approximation of Hessian matrix $\nabla^2 f$, it is very natural to assume it is symmetric. Hence it contains $\frac{1}{2}n(n+1)$ unknown variables, plus n unknowns in g_k and variable c, we have N unknowns and N linear equations. In this case, the linear system is actually a square system. And if choosing x_k^j properly, the system could be nonsingular, and hence has unique solution. Hence, we can obtain a good formulation of the quadratic function m_k from N samples of original function. After that, the optimization problem can be solved by modified trust region method. A complete description of the algorithm would be cumbersome, we

only give a brief description here. For more details of the algorithm, please refer to [60] and [55]. Also, in [9], the method is extended to a broader class of optimization problem with nonlinear constrains.

1. Choose N starting points properly, denote them as an interpolation set I. Select the point in I which gives the smallest objective value, denote that point as x_0. Initialize iteration timer $k = 0$. And give a trust region radius Δ_0.

2. Build the merit function m_k by quadratic interpolation as stated above.

3. Solve for p in

$$\min_{p \in \mathbb{R}^n} m_k(x_k + p), \quad \|p\|_2 \leq \Delta_k$$

   If $\|p\|_2$ is too small, check whether $f(x_k + p) - f(x_k)$ is small enough, if not, update the interpolation set I by delete the worst point and add a new point. Otherwise, decrease Δ_k and repeat this step.

   If $\|p\|_2$ is proper, compute $f(x_k + p)$. Add $x_k + p$ to set I. Also, update Δ_k and x_k accordingly.

4. If Δ_k is small enough, stop. Otherwise, $k = k + 1$ and go back to step 2.

6.4 Numerical results

In this section, we compare calibration results of different optimization methods. Also, we present numerical examples on calibration of American (Bermudan) options in various Lévy process models, including Kou's double exponential jump diffusion model ([49]), the CGMY model ([14]) and the normal inverse Gaussian (NIG) model ([7]). In our numerical experiments, first we do intensive study on the choice of $N1$, N_2, to ensure we can achieve our desired accuracy for calibration while keep computing time as short as possible. Then we use simulated data to compare different optimization methods in order to choose the most ideal one for further calibration. Then, the last step is conducted by data deriving from real world market to verify the calibration result of different models by checking whether they can fit the market data well. We use OEX and XEO options in our calibration, who have the same underlying asset S&P100 Index. *All the market data is obtained from Bloomberg Terminal.

6.4.1 Choosing N_1 and N_2

For those three Lévy models we are interested, we do intensive numerical study on how to choose N_1 and N_2, so that we can achieve percentage error of 0.001. We have some interesting findings here. In our illustration, we use Kou's model for example, the NIG and CGMY case are similar unless mentioned otherwise.

Compare different r, q

First, we want to know how r, q will affect pricing error if we fix the other conditions. According to the market situation, we choose four sets of r, q to compare, they are: $(r, q) = (0.0026, 0.024)$, $(0.05, 0.024)$, $(0.05, 0)$, $(0.0026, 0)$. The first set is the market data for 40 days OEX option on Apr.12.2010, and these values keeps at the same level recently. And for recent ten years, the 12 months interest rate is lower or approximate to the level of 5%. So, we are taking 5% as a proper large interest rate value. In table 6.1, we show the pricing error for three typical sets of parameter in Kou's model. Here we take M large enough to ensure the value of M won't incur further pricing error. Also, we present the case $N = 1$ without extrapolation, we also tested other case like $N = 2, 4, 5, 10, 20...$ and the extrapolation case like $N2 = 2N_1 = 2, 4, 10, 20, ...$, the situations are similar, so we choose the simplest one to present (Similar situation for NIG and CGMY model). We can see that in table 6.1, the value of r has tremendous influence on the pricing error. Large r would incur large pricing error. On the other hand, the difference $r - q$ also matters, although not in a tremendous way. Hence, in the following test, we fix $r = 0.05$ and $q = 0$ so that if the conclusion we draw on the upper bound of pricing error works for this r, q, it would also be valid for other r, q values.

Table 6.1: M=1e5, α=1, T= 40, $N = 1$, Largest pricing error among different strike K.

para;(r,q)	(0.0026,0.024)	(0.05, 0.024)	(0.05, 0)	(0.0026 0)
(0.9, 10, 0.01, 4, 4)	1.853E-07	0.002328392	0.003341444	4.35520E-05
(0.9, 1, 0.01, 4, 80)	3.25830E-14	0.002074508	0.003321795	7.58451E-05
(0.9, 10, 0.01, 100, 4)	1.86034E-07	0.002682455	0.003746877	0.000138182

Compare different maturity T

Then we would like to know, how maturity T would affect the pricing error in our numerical experiments. We first compare two maturities $T = 40, 182$ days. The pricing errors are illustrated

in table 6.2. We find that the pricing errors for $T = 182$ are larger than $T = 40$, although the error difference is smaller if extrapolation is used . Also, we compare the numerical result for $T = 365$ and $T = 182$ days. Our result shows that pricing errors are on the same level. Generally, options with long maturity would have slightly larger pricing error than those with short maturity. While the differences are not obvious.

Table 6.2: parameter set $(0.9, 10, 0.01, 100, 80)$, M=1e5, α=1, r=0.05, q=0

$T = 40$	Without extrapolation			Extrapolation		
N/Strike	480	545	580	480	545	580
5	0.001250846	0.001199909	0.001159857			
10	0.000696225	0.000640091	0.000607702	0.000141604	0.000080273	0.000055547
20	0.000365180	0.000328934	0.000309613	0.000034135	0.000017777	0.000011525
$T = 182$	Without extrapolation			Extrapolation		
N/Strike	480	545	580	480	545	580
5	0.003162840	0.003076123	0.003028083			
10	0.001671884	0.001599814	0.001563086	0.000180929	0.000123505	0.000098090
20	0.000859952	0.000815231	0.000793204	0.000048020	0.000030648	0.000023323

Compare different strikes K

Then we consider whether strike price K would affect the pricing error. It turns out that most moderate strikes would give the same level of pricing error, especially when there is no extrapolation used. When extrapolation is used, it seems that the lower strike prices will accompany higher pricing error. See table 6.3 and table 6.4.

Choosing M

In calibration problem, we will encounter many different parameters in our calibration. The pricing error largely depends on the parameters. For example, for one parameter, if we choose $M = 1e3$, we will get accuracy of $1e - 6$, but for some other parameter, we may need $M = 1e5$ to obtain the same precision. Also, we do not want to choose a fixed large M in our calibration process. Because if we do so, the working load may increase hundreds of time in the process. Hence, we need to give a standard to choose M according to model parameters. However, we are unable to derive a direct function describing M and pricing error of American options. Instead, we can only give approximite guess on the relationship between those two and try to verify that by intensive numerical experiments. In [39], we have the following error estimation for Hilbert transform:

$$\|\mathcal{H}f - \mathcal{H}_{h(M),M}f\|_{L^\infty} \leq C_1 M^{1/(1+\nu)} \exp(-C_2 M^{\nu/(1+\nu)})$$

where $C_1 > 0$ and $C_2 = (\pi d)^{\nu/(1+\nu)}(c_2)^{1/(1+\nu)}$. Here c_2 depends on the tail behavior of the Lévy density. (Please see [39] for more detail.) In the above error estimation, the dominated term is $E(M) = C_2 M^{\nu/(1+\nu)}$ in the exponential function. Hence, we want to have this value fixed, and solve for M accordingly. That is, we assume the error of American option price is in the form of $Ce^{-E(M)}$, where C is an unknown constant. Our numerical experiments shows that, for Kou's model and NIG model, we can let $E(M) = 10$ to guarantee the precision of $1e-4$. For CGMY model, the situation is more complex. Because for Kou's model and NIG model, we have ν as a constant. But in CGMY, $\nu = Y$ is a variable. And if we choose $E(M) = 10$ as for the other two models, we will derive huge M ($> 1e6$) in certain situation. After some tests, we fix $E(M) = 6$ here, which ensures the pricing error is below $4e-3$. Although larger than Kou and NIG's case, it is still sufficient for our calibration purpose.

Choosing N_1 and N_2

Since the value of N_1 and N_2 have significant influence on option pricing time and hence calibration time, we prefer to have them small. On the other hand, if they are too small, we will have very large pricing error, which would affect our calibration result. Since the market option price is given with accuracy of 0.01, we require our pricing error caused by N_1 and N_2 be around 0.001. Then, combining the error incurred by M, we can still keep the total pricing error within the range of 0.005, which should be enough for our calibration purpose. Since we have already taken the infulence of r, q, M, T and K into account, now we only need to check how different model parameter sets would affect our pricing error. After considering those possible values of parameter sets in real world market, we restricted our parameter sets in the following bound: $(\sigma, \lambda, p, \eta_1, \eta_2)$, $lb = [0.01, 1.0, 0.01, 4, 3]$ and $ub = [0.9, 10.0, 0.99, 200, 180]$. After intensive numerical experiment and comparing the option pricing error for different parameter sets, in table 6.3 and table 6.4, we show the largest pricing error among all different parameter sets we obtained. And we can see that, without extrapolation, we need to choose $N = 20$ to achieve the accuray of $2e-3$, while on the other hand, the extrapolation of $N_1 = 5$ and $N_2 = 10$ would provide the accuracy of $7e-4$. And their working load is in the same level with the latter with extrapolation being at least 15% faster. Hence, we will take $N_1 = 5$ and $N_2 = 10$ in our following numerical experiments.

Table 6.3: parameter set $(0.9, 10, 0.01, 200, 180)$, M=1e5, α=1, T= 365, r=0.05, q=0

	Without extrapolation					Extrapolation				
N/Strike	200	450	550	650	900	200	450	550	650	900
5	5.3e-3	5.1e-3	4.9e-3	4.8e-3	4.4e-3					
10	3.0e-3	2.6e-3	2.5e-3	2.4e-3	2.2e-3	7.0e-4	2.1e-4	1.2e-4	6.1e-5	2.8e-5
20	1.6e-3	1.4e-3	1.3e-3	1.2e-3	1.1e-3	2.1e-4	5.3e-5	2.8e-5	1.1e-5	1.3e-5

Table 6.4: parameter set $(0.9, 1, 0.01, 4, 180)$, M=1e5, α=1, T= 365, r=0.05, q=0

	Without extrapolation					Extrapolation				
N/Strike	200	450	550	650	900	200	450	550	650	900
5	4.9e-3	4.8e-3	4.7e-3	4.6e-3	4.3e-3					
10	2.7e-3	2.5e-3	2.4e-3	2.3e-3	2.1e-3	5.5e-4	1.4e-4	6.7e-5	1.8e-5	2.2e-6
20	1.4e-3	1.2e-3	1.2e-3	1.1e-3	1.1e-3	1.5e-4	2.4e-5	5.4e-6	6.6e-6	2.3e-5

6.4.2 Calibration using simulated data

Here, we want to find out which optimization method can serve our calibration purpose better. We assume that the risk free interest rate is $r = 5\%$, the continuous yield of the underlying asset is paying $q = 2\%$. Also, we choose our parameter sets as $\Theta^* = (\sigma, \lambda, p, \eta_1, \eta_2) = (0.1, 3.0, 0.3, 40, 12)$. We use these data to generate a bunch of option prices and assume these option prices are "real" market option prices. And then, the parameters we gave is the "real" parameters. Then, we try to find out whether our optimization method can find the parameters in the parameter space, starting from some initial points. Also, through these numerical experiments, we want to compare different optimization methods by their computing time and accuracy.

Compare Automatic Differentiation and Finite Difference

The advantage of Automatic Differentatiation over Finite Difference is that it provides more accurate derivatives. However, since our calibration problem itself has not much requirement on the accuracy, the precision of FD seems to be enough for us (around 1E-8). Hence, we care more about the time cost of AD and FD. If they are at the same level, of course we will choose FD over AD because of the accuracy. Otherwise, we prefer to use FD. To compare the time cost, we design some simple tests. In the calibration problem (6.10), we take

$$R(\Theta) = \alpha * \| \frac{\Theta - \Theta^*}{\Theta^*} \|_2 \tag{6.11}$$

Here we use this penalty term to pull the parameter Θ back to Θ^*. Also, we set $\alpha = 10$ big enough, so that the optimization problem is highly convex. Then no matter where our initial parameter Θ_0 is, we can find the optimal point Θ^* by gradient based optimization methods. We incorporated the AD and FD with BFGS Quasi-Newton method for the following experiment. Set $S_0 = 100$, $K = \{90, 100, 110\}$ and $T = \{20, 126, 252\}$. First we compute these option prices and use them as benchmark price. Then we give different starting points Θ_0 and compare the computing time for AD and FD. It turns out that both methods can find the benchmark parameter. In table 6.5, we show the time cost with number of iterations in brackets. We can see that, except the last starting point, FD is obvious faster than AD. Also, FD being slower than AD for the last parameter might be because the landscape between the last parameter and our benchmark parameter is very complex, hence, the accuracy of gradient plays an important role in fast finding the target. However, in our calibration, the gradient method is used for updating the new parameter from the previous one when incorporating new data. Then the parameters normally have very little change. The situation is more like calibration using the first starting point in table 6.5. Therefore, we prefer to use FD instead of AD in our calibration.

Table 6.5: Computing time for AD and FD.

Starting point/Method	Automatic Differentiation	Finite Difference
1.2*(0.1, 3.0, 0.3 , 40, 12)	385.12s (4)	240.99s (6)
1.5*(0.1, 3.0, 0.3 , 40, 12)	458.46s (5)	206.09s (5)
1.8*(0.1, 3.0, 0.3 , 40, 12)	531.59s (6)	273.53s (7)
3.0*(0.1, 3.0, 0.3 , 40, 12)	765.27s (9)	340.87s (9)
(0.5, 7.0, 0.6 , 20, 45)	602.42s (7)	782.29s (19)

Compare DFO with Gradient Based method with FD

Similarly, we can do some simple tests to compare DFO method we mentioned before with Gradient Based method combined with FD, to see which one performs better in our calibration process. Here, we use one maturity but several different strikes $\{70, 75, ..., 130\}$. Also, we choose $\alpha = 3$ to increase the difficulty of the optimization problem. In this case, the minima of the optimization problem can also be easily found. We show the computing time with number of function evaluated in bracket in table 6.6. We can see that Gradient based method is faster than DFO. And interior-point method seems to fit our calibration problem better than Active Set method.

Table 6.6: Computing time for DFO and Gradient based methods.

para/method	DFO	Active Set	Interior-point
1.2*(0.1, 3.0, 0.3 , 40, 12)	163.53s (300)	68.31s (128)	78.66s (155)
2.0*(0.1, 3.0, 0.3 , 40, 12)	130.77s (279)	114.58s (145)	73.95s (158)
(0.5, 7.0, 0.6 , 20, 45)	188.25s (389)	112.25s (142)	84.89s (169)

6.4.3 Calibration using market data

After the numerical experiments we did previously, we have a general idea about our calibration problem. Now we can use market data (S&P100 Options) to do calibration. And find out whether these Lévy process models can fit the market data well or not. In the following numerical experiments, first, we use the global optimization method introduced previously to derive some good parameters (providing small enough objective value) in our parameter space. Then we use gradient based method with finite difference to solve the optimization problem thoroughly. Be aware that although the global optimization method (particle swarm pattern search) itself could give us a local minimum, it costs large amount of time as often global optimization algorithms do. Hence, we can run the global optimization method for a certain period of time (1000-3000 function evaluation or 1-3 hours of CPU time) so that it can sufficiently explore the surface of our parameter spaces and provide us with some good points in the neighborhood of our ideal local minimum. Then we leave the rest task to gradient based method, which is much faster than the global optimization method in case of local search.

Comparison of models for fixed maturities

First, we want to calibrate a simple case with only one maturity but multi-strikes. In this part, we use the option price data from OEX market on Apr.12.2010. In the market data, option price are given in the form of bid-ask prices. For our benchmark prices, we use the average prices of bids and asks. We first calibrate the models without penalty term to fully incorporate the information in market. In table 6.7, we show the option prices obtained from calibration for $T = 40$ days. First we analyze Kou's model. We can see that, for four different sets of parameters P1, P2, P3, P4, the calibration result are all very good. Only one option price falls out of the bid-ask spread. Also, the four sets of parameters agrees on the volatility $\sigma \approx 11.5\%$ and $\eta_2 \approx 15.5$, which means that the average negative jump size is around $1/\eta_2 \approx 6.45\%$. These two parameters do agree with the market situation. Then, we calibrate OEX options on the same day with a maturity of $T = 250$ days. The

result is in table 6.8. Except the prices calibrated by P4, the rest of calibrated option prices are all in the bid-ask spread (The number of available market option price are less than that in $T = 40$). From the parameters we obtained, we find that, p in the first three parameter sets are all above 0.5, which means that positive jump is more likely than negative jump. This disagrees with the market situation. And for P4, although its $p < 0.5$, the calibration is not very ideal (4 prices out of 18 are not in the bid-ask spread). The reason might be that Kou's model fits better for short maturity options rather than long maturity options. And this agrees with the finding of R.Cont and P.Tankov in [22] when they use nonparametric method to calibrate European options. Also in table 6.7, we can see that, for CGMY model, there is also only one option price out of the bid-ask spread. Moreover, we can see that the objective value is around half of that for Kou's model. To some extent, it means that CGMY model fits the market data even better than Kou's although it has one dimension of freedom less than Kou's model.

Table 6.7: Calibrated Option price in Levy's model, $T = 40$ days

			Kou's model				CGMY	
Parameters	bid	ask	P1	P2	P3	P4	P1	P2
Strike	Obj value $->$		1.28E-04	1.23E-04	1.25E-04	1.27E-04	6.00E-05	6.35E-05
480	0.75	0.95	0.84	0.84	0.84	0.84	0.86	0.86
485	0.90	1.10	0.99	0.99	0.99	0.99	0.99	0.99
490	1.05	1.25	1.16	1.16	1.16	1.16	1.15	1.15
495	1.25	1.50	1.36	1.36	1.36	1.36	1.34	1.34
500	1.45	1.70	1.60	1.60	1.60	1.60	1.58	1.58
505	1.75	2.05	1.89	1.89	1.89	1.89	1.88	1.88
510	2.10	2.40	2.26	2.26	2.26	2.26	2.25	2.25
515	2.55	2.90	2.72	2.72	2.72	2.72	2.73	2.73
520	3.20	3.50	3.32	3.32	3.32	3.32	3.34	3.34
525	3.90	4.30	4.10	4.10	4.10	4.10	4.13	4.13
530	5.10	5.30	5.12	5.12	5.12	5.12	5.13	5.13
535	6.10	6.60	6.44	6.44	6.44	6.44	6.41	6.42
540	7.70	8.20	8.11	8.11	8.11	8.11	8.02	8.03
545	9.60	10.20	10.17	10.17	10.17	10.17	10.01	10.02
550	12.20	12.60	*12.66*	*12.65*	*12.65*	*12.66*	12.43	12.44
555	14.90	15.60	15.58	15.56	15.56	15.57	15.28	15.29
560	18.10	19.00	18.91	18.88	18.89	18.89	18.57	18.57
565	21.90	23.00	22.61	22.58	22.59	22.60	22.27	22.26
570	26.10	27.30	26.65	26.61	26.61	26.62	26.31	26.31
575	30.60	31.70	30.95	30.91	30.91	30.92	30.65	30.64
580	35.40	36.40	35.47	35.42	35.42	35.43	*35.21*	*35.20*

	$(\sigma, \lambda, p, \eta_1, \eta_2)$	(C, G, M, Y)	
P1	(0.1160 1.6554 0.0100 08.5929 15.5550)	(0.0223 2.4965 76.4292 1.4891)	
P2	(0.1164 1.6508 0.0100 21.7420 15.5341)	(0.0209 2.3625 91.5745 1.5067)	
P3	(0.1157 2.3041 0.2866 79.9999 15.5702)		
P4	(0.1154 2.3784 0.3071 69.6223 15.5866)		

Table 6.8: Calibrated Option price in Kou's model, $T = 250$ days

Parameters	bid	ask	P1	P2	P3	P4
Strike/Objective value			9.24E-05	9.04E-05	8.88E-05	1.65E-04
300	0.65	0.90	0.78	0.78	0.78	0.78
320	1.05	1.30	1.16	1.16	1.16	1.16
340	1.55	1.90	1.68	1.68	1.68	1.68
360	2.20	2.60	2.39	2.39	2.39	2.40
380	3.00	3.60	3.33	3.33	3.33	3.34
400	4.20	4.80	4.56	4.56	4.56	4.58
420	5.90	6.40	6.14	6.14	6.14	6.17
440	7.60	8.60	8.16	8.17	8.17	8.21
460	10.10	11.20	10.75	10.76	10.75	10.79
480	13.80	14.70	14.10	14.11	14.11	14.11
500	24.10	19.30	18.56	18.57	18.57	18.49
520	24.10	25.40	24.56	24.55	24.56	24.40
540	31.00	33.80	32.55	32.49	32.51	32.36
560	40.50	43.40	42.76	42.68	42.69	42.74
580	52.40	55.50	55.20	55.15	55.14	*55.57*
640	101.10	103.70	102.98	103.27	103.11	*104.88*
660	120.00	122.60	121.16	121.54	121.34	*123.32*
680	139.60	142.10	139.94	140.36	140.15	*142.19*

	$(\sigma, \lambda, p, \eta_1, \eta_2)$
P1	(0.0920 5.1876 0.8688 33.7958 6.2544)
P2	(0.1059 1.3979 0.5147 16.1934 6.2509)
P3	(0.1034 1.9570 0.6531 20.9395 6.2515)
P4	(0.0978 1.0000 0.3069 8.4223 6.2998)

Across maturity

In this part, instead of using one maturity in calibration, we consider two maturities together in the calibration. The data used is still from OEX option on S&P100 Index but on May.4.2010. The two maturities we used is $T = 18$ days and $T = 228$ days. Still we take Kou's model as example for illustration purpose. After spend huge amount of time exploring the parameter space, we are still unable to obtain an ideal parameter. In table 6.9 we show some of the calibration information. The best we have is to fit half of the options we get. For comparison purpose, we also calibrated XEO (European) options in SVCJ model. There is no fast way to compute American options under SVCJ model, so we use European options instead. The good thing is that OEX and XEO options are both on S&P100 Index, so they are parallel options sharing same underlying asset, maturities, interest rate, etc... We applied Hilbert transform method proposed in [38]. As we can see in table 6.9, the calibration result is pretty satisfactory. We have more than 90% of options in the bid-ask spread, which shows that SVCJ model successfully incorporates the rich structure of both jump and stochastic volatility so that it can better model the index change in both short term and long term.

Table 6.9: Across maturity calibration

Kou's model for $T = 18$ and $T = 228$		
Parameter	Objective value	No. of options in bid-ask spread / No. of Total options
P1	7.90E-02	32/60
P2	7.90E-02	32/60
P3	7.43E-02	22/60
P1=(0.1551, 9.3716, 0.6098, 199.9999, 14.9431)		
P2=(0.1558, 5.2443, 0.3034, 200.0000, 14.9383)		
P3=(0.1149, 4.3514, 0.0989, 4.0000, 15.3943)		

European Put, Kou's model for $T = 18$ and $T = 228$		
Parameter	Objective value	No. of options in bid-ask spread / No. of Total options
P1	1.29E-02	19/56
P2	1.29E-02	19/56
P3	1.34E-02	18/56
P1=(0.1858, 9.3207, 0.9538, 199.9996, 4.5116)		
P2=(0.1863, 5.1256, 0.9161, 200.0000, 4.5108)		
P3=(0.1833, 1.0000, 0.5612, 24.2228, 4.5503)		

European Put, SVCJ model for $T = 18$, $T = 46$ and $T = 228$.		
Parameter	Objective value	No. of options in bid-ask spread / No. of Total options
P1	8.23E-04	69/76
P2	8.01E-04	69/76
P3	7.99E-04	69/76
P4	9.59E-04	69/76
P1=(-0.4718, -0.4741, 0.7215, 2.3496, 0.0075, 1.0983, -0.0864, 0.0010, 0.1127, 0.0234)		
P2=(-0.4740, -0.6849, 0.6809, 1.4633, 0.0010, 1.2429, -0.0744, 0.0010, 0.0842, 0.0224)		
P3=(-0.4586, -0.5675, 0.6907, 1.8585, 0.0010, 1.2391, -0.0775, 0.0010, 0.0963, 0.0226)		
P4=(-0.4911, -0.4756, 0.9437, 3.3952, 0.0221, 0.9582, -0.0847, 0.0014, 0.1220, 0.0247)		

Cross Validation

In this part, we still calibrate options on single maturity. Instead of using all the available option prices on that specific day. We divide the options into two sets. Among which, the first one is called training set. We use all the data in the training set to calibrate. The second set is called validation set. We use the parameters we calibrated from training set to generate option prices for validation set and compare with the bid-ask spread. The data we used is OEX option on May.4.2010, with maturity of $T = 46$ days. As shown in table 6.10, we did three tests, for each of them, we use different training set (set1) and validation set (set2). The result seems to be very good. Except one option, the others all fall in the bid-ask spread. Hence, we again verify that Kou's model (Lévy process models) fits the market data well.

Stability

For the above numerical experiments, we only calibrate options on one specific day. But for practical purpose, the calibration should be done continuously for a period of time, like daily calibration.

Here, we will run the calibration on daily data and verify the stability of the calibration. In table 6.11, we present the option prices we calibrated on May.4th, May.5th and May.6th. To be specific, first we calibrate the American option under Kou's model on May.4.2010 by the same method we used above, then we choose one set of parameter and use it as a starting point to calibrate the data on May.5.2010. And then use the new parameter derived on May.5th to calibrate the data on May.6.2010. For May.5th and May.6th, we actually calibrated the options twice for each day. The first time is without penalty term. The calibrated parameters are denoted as P1-N and P2-N. Then we use the same data and same starting point to do the calibration again but with the penalty term (6.11). Then the result we got is shown in P1-W and P2-W. As can be observed, the calibration result is satisfactory. For May.5th, there are three option prices being out of the bid-ask spread, for May.6th, there are two options. Considering the total number of options calibrated is 32, we have more than 90% of options fall in the bid-ask spread. On the other hand, we want to mention that, the underlying asset price, which is S&P100 Index on these three days are 534.79, 531.45 and 513.80 respectively. On May.6th, the Index dropped 3.32%, which is a dramatic change for indexes. In this case, the calibration process still provides us with a good result, which to a large extent shows the stability of the calibration method. Also, we want to take a look at the objective value and penalty term value, these information is also provided in table 6.11. Here we denote the objective value by the function value given by (6.10) without the penalty term $R(\Theta)$. And the penalty term value is referred to $R(\Theta)$ in (6.11). We can see that, the objective value we obtained are similar no matter the penalty term is added or not, hence, the calibration result is similar. But on the other hand, for the value of penalty term, P1-N $>$ P1-W and P2-N $\gg$ P2-W. This means that, with penalty term, we can restrict our parameter sets within certain region, so that, the calibrated parameters will not change dramatically when underlying asset price has big movement. This is good for hedging purpose. Since a big change of parameters means a big change of hedging positions. And with the help of the penalty term, we can keep the parameter change within control while obtain the objective value on the same level as calibrations without penalty term.

Table 6.10: Cross Validation in Kou's model, $T = 46$ days

			Test 1		Test 2		Test 3	
Strike K	bid	ask	Set1	Set2	Set1	Set2	Set1	Set2
300	0.00	0.15		0.03		0.03		0.02
340	0.00	0.25		0.12		0.11		0.08
360	0.10	0.30		0.21		0.20		0.16
380	0.25	0.50	0.36			0.34		0.29
400	0.45	0.70		0.61		0.58		0.51
415	0.70	1.00	0.88			0.85		0.77
420	0.85	1.10	0.99		0.96			0.88
430	1.10	1.40	1.25		1.23			1.14
440	1.40	1.75		1.58	1.55			1.47
445	1.60	1.90	1.77		1.75			1.66
450	1.75	2.10	1.97		1.96			1.88
455	2.00	2.30	2.21		2.19			2.12
460	2.25	2.55		2.46	2.45		2.39	
465	2.50	2.85	2.74		2.74		2.69	
470	2.85	3.20	3.06		3.06		3.02	
475	3.20	3.60	3.40		3.42		3.39	
480	3.60	4.00		3.79	3.81		3.81	
485	4.00	4.40	4.22		4.25		4.27	
490	4.50	5.00	4.70		4.75		4.79	
495	5.10	5.60	5.26		5.32		5.37	
500	5.70	6.30		5.91	5.97		6.04	
505	6.50	7.10	6.67		6.72		6.80	
510	7.40	8.00	7.57		7.61		7.69	
515	8.50	9.00	8.64		8.67		8.74	
520	9.70	10.40		9.92	9.93		9.98	
525	11.20	11.90	11.44		11.43		11.45	
530	12.90	13.50	13.24		13.21		13.18	
535	14.80	15.60	15.34		15.29		15.22	
540	17.10	17.80		17.77	17.70		17.58	
545	19.70	20.50	*20.53*		20.44		20.29	
550	22.60	23.80	23.62		23.53		23.34	
555	25.90	27.10	27.03			26.93	26.73	
560	29.50	30.80		30.73		30.64	30.43	
565	33.50	34.80	34.70			34.61	34.42	
570	37.70	39.10	38.89			38.81	38.64	
575	42.20	43.50	43.28			43.21	43.06	
580	46.80	48.20		47.83		47.77	47.65	

	$(\sigma,\ \lambda,\ p,\ \eta_1,\ \eta_2)$
Test 1	(0.1344 2.0029 0.0100 22.8642 10.1116)
Test 2	(0.1292 3.5333 0.3830 79.4301 10.5364)
Test 3	(0.1232 2.6779 0.0100 22.3294 11.5679)

Table 6.11: Calibrated Option Price On Continous Days

Dates	May.4th	May.5th		May.6th	
Strike/Parameters	P	P1-N	P1-W	P2-N	P2-W
415	0.13	0.17	0.17	0.69	0.70
420	0.15	0.20	0.20	0.80	0.81
430	0.22	0.28	0.28	1.06	1.07
440	0.31	0.40	0.40	1.39	1.40
445	0.37	0.47	0.47	1.59	1.60
450	0.44	0.56	0.56	1.82	1.83
455	0.53	0.66	0.66	2.08	2.09
460	0.63	0.77	0.77	2.38	2.38
465	0.74	0.91	0.91	2.72	2.72
470	0.87	1.06	1.07	3.12	3.11
475	1.03	1.24	1.25	3.58	3.58
480	1.21	1.45	1.46	4.14	4.13
485	1.42	1.70	1.71	4.82	4.80
490	1.66	1.99	2.00	5.65	5.63
495	1.95	2.33	2.34	6.69	6.64
500	2.29	2.75	2.76	7.96	7.89
505	2.70	3.28	3.28	*9.52*	9.41
510	3.22	*3.97*	*3.95*	11.41	11.25
515	3.88	4.86	4.83	*13.65*	13.44
520	4.76	6.02	5.98	16.26	15.98
525	5.94	7.54	7.47	19.25	18.89
530	7.50	9.46	9.38	22.58	22.16
535	9.53	*11.85*	*11.75*	26.24	25.76
540	12.07	14.70	14.61	30.19	29.66
545	15.15	18.02	17.93	34.37	33.82
550	18.72	21.76	21.68	38.76	38.20
555	22.72	25.86	25.79	43.30	42.75
560	27.07	30.24	30.18	47.96	47.43
565	31.66	34.84	34.80	52.71	52.21
570	36.43	39.59	39.56	57.53	57.07
575	41.30	44.44	44.42	62.39	*61.97*
580	46.23	*49.35*	*49.34*	67.28	*66.91*

	$(\sigma, \lambda, p, \eta_1, \eta_2)$	obj value	$R(\Theta)$
P	(0.1379 4.4276 0.0100 21.7685 16.0045)	3.30E-03	
P1-N	(0.1614 4.5342 0.0100 21.7708 15.2684)	1.20E-03	0.0317
P1-W	(0.1583 4.6459 0.0100 21.7686 15.3683)	1.20E-03	0.0259
P2-N	(0.2084 8.1297 0.3273 21.7710 12.2589)	1.50E-03	1.01E+03
P2-W	(0.2199 5.3777 0.0100 21.7664 12.0565)	1.50E-03	0.2227

References

[1] J. Abate and W. Whitt. The Fourier-series method for inverting transforms of probability distributions. *Queueing Systems*, 10(1-2):5–88, 1992.

[2] L. Andersen and J. Andreasen. Jump diffusion models: volatility smile fitting and numerical methods for pricing. *Review of Derivatives Research*, 4(3):231–262, 2000.

[3] J. Andreasen. The pricing of discretely sampled Asian and lookback options: a change of numeraire approach. *Journal of Computational Finance*, 2(1):5–30, 1998.

[4] S. Asmussen and P. W. Glynn. *Stochastic Simulation*. Springer-Verlag, Berlin, Germany, 2007.

[5] S. Asmussen and J. Rosiński. Approximation of small jumps of Lévy processes with a view towards simulation. *Journal of Applied Probability*, 38(2):482–493, 2001.

[6] C. Audet and J. J. E. Dennis. Analysis of generalized pattern searches. *SIAM Journal on Optimization*, 13(3):889–903, 2003.

[7] O. E. Barndorff-Nielsen. Process of normal inverse Gaussian type. *Finance and Stochastics*, 2(1):41–68, 1998.

[8] E. Benhamou. Fast Fourier transform for discrete Asian options. *Journal of Computational Finance*, 6(1), 2002.

[9] F. V. Berghen. Optimization algorithm for non-linear, constrained, derivative-free optimization of continous, high-computing-load functions. Technical report, IRIDIA, Université Libre de Bruxelles, Belgium, 2004.

[10] F. V. Berghen and H. Bersini. Condor, a new parallel, constrained extension of powell's uobyqa algorithm: experimental results and comparison with the dfo algorithm. *Journal of Computational and Applied Mathematics*, 181(1), 2005.

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

[12] L. Bondesson. On simulation from infinitely divisible distributions. *Advances in Applied Probability*, 14(4):855–869, 1982.

[13] R. Byrd, J. Gilbert, and J. Nocedal. A trust region method based on interior point techniques for nonlinear programming. *Mathematical Programming*, 89(1):149–185, 2000.

[14] P. Carr, H. Geman, D. B. Madan, and M. Yor. The fine structure of asset returns: an empirical investigation. *Journal of Business*, 75(2):305–332, 2002.

[15] P. Carr and D. Madan. Saddlepoint methods for option pricing. *Journal of Computational Finance*, 13(1):49–61, 2009.

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

[17] A. P. Carverhill and L. J. Clewlow. Flexible convolution. *Risk*, 3:25 – 29, 1990.

[18] M. C.Fu and J.-Q. Hu. Sensitivity analysis for monte carlo simulation of option pricing. *Probability in the Engineering and Informational Sciences*, 9(03):417–446, 1995.

[19] Z. Chen and L. Feng. Monte carlo estimation of sensitivities from analytic characteristic functions. *Working paper, University of Illinois at Urbana-Champaign*, 2012.

[20] Z. Chen, L. Feng, and X. Lin. Simulating Lévy processes from their characteristic functions and financial applications. *ACM Transactions on Modeling and Computer Simulation*, 22(3), 2012.

[21] T. Coleman and Y. Li. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6:418–445, 1996.

[22] R. Cont and P. Tankov. *Financial Modelling with Jump Processes*. Chapman & Hall/CRC, Boca Raton, Florida, 2004.

[23] R. Cont and P. Tankov. Non-parametric calibration of jump-diffusion option pricing models. *Journal of Computational Finance*, 7(3):1–49, 2004.

[24] R. Cont and P. Tankov. Retrieving Lévy processes from option prices: regularization of an ill-posed inverse problem. *SIAM Journal on Control and Optimization*, 45(1):1–25, 2006.

[25] S. Crépey. Calibration of the local volatility in a generalized Black-Scholes model using Tikhonov regularization. *SIAM Journal on Mathematical Analysis*, 34(5):1183–1206, 2003.

[26] S. Crépey. Calibration of the local volatility in a trinomial tree using Tikhonov regularization. *Inverse Problems*, 19:91–127, 2003.

[27] E. Derman. Laughter in the dark - the problem of the volatility smile. Available at http://www.ederman.com, 2003.

[28] E. Derman and I. Kani. Riding on a smile. *Risk*, pages 32–39, February 1994.

[29] L. Devroye. On the computer generation of random variables with a given characteristic function. *Computers & Mathematics with Applications*, 7(6):547–552, 1981.

[30] D. Duffie, J. Pan, and K. Singleton. Transform analysis and asset pricing for affine jump-diffusions. *Econometrica*, 68(6):1343–1376, 2000.

[31] B. Dupire. Pricing with a smile. *Risk*, pages 18–20, January 1994.

[32] D. Düvelmeyer and B. Hofmann. A multi-parameter regularization approach for estimating parameters in jump diffusion processes. *Journal of Inverse and Ill-Posed Problems*, 14(9):861–880, 2006.

[33] R. C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micromachine and Human Science, Nagoya, Japan.*, pages 39–43, 1995.

[34] E.C.Titchmarsh. *Introduction to the thoery of Fourier integrals*.

[35] H. Egger and H. Engl. Tikhonov regularization applied to the inverse problem of option pricing: convergence analysis and rates. *Inverse Problems*, 21(3):1027–1045, 2005.

[36] J.-C. Evard and F. Jafari. A complex rolle's theorem. *American Mathematical Monthly.*, 99(9):858–861, 1992.

[37] L. Feng. *Computational methods for Lévy and jump-diffusion processes: applications in financial engineering*. PhD thesis, Northwestern University, Evanston, USA, 2006.

[38] L. Feng and X. Lin. Inverting analytic characteristic functions and financial applications. *Revision Submitted*, 2012.

[39] L. Feng and X. Lin. Pricing Bermudan options in Lévy process models. *Revision Submitted*, 2012.

[40] L. Feng and V. Linetsky. Pricing discretely monitored barrier options and defaultable bonds in Lévy process models: a fast Hilbert transform approach. *Mathematical Finance*, 18(3):337–384, 2008.

[41] L. Feng and V. Linetsky. Computing exponential moments of the discrete maximum of a Lévy process and lookback options. *Finance and Stochastics*, 13(4):501–529, 2009.

[42] G. Fusai and A. Meucci. Pricing discretely monitored Asian options under Lèvy processes. *Journal of Banking and Finance*, 32:2076–2088, 2008.

[43] P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer-Verlag, Berlin, Germany, 2004.

[44] P. Glasserman and Z. Liu. Sensitivity estimates from characteristic functions. In S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, editors, *Proceedings of the 207 Winter Simulation Conference*, pages 932–940. 2007.

[45] P. Glasserman and Z. Liu. Sensitivity estimates from characteristic functions. *Operations Research*, 58(6):1611–1623, 2010.

[46] T. Hein and B. Hofmann. On the nature of ill-posedness of an inverse problem arising in option pricing. *Inverse Problems*, 19(6):1319–1338, 2003.

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

[48] B. Hofmann and R. Krämer. On maximum entropy regularization for a sepcific inverse problem of option pricing. *Journal of Inverse and Ill-Posed Problems*, 13(1):41–63, 2005.

[49] S. G. Kou. A jump-diffusion model for option pricing. *Management Science*, 48(8):1086–1101, 2002.

[50] R. Lagnado and S. Osher. A technique for calibrating derivative security pricing models: numerical solution of an inverse problem. *Journal of Computational Finance*, 1(1):13–25, 1997.

[51] P. L'Ecuyer. Quasi-Monte Carlo methods in finance. In R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, editors, *Proceedings of the 2004 Winter Simulation Conference*. 2004.

[52] P. L'Ecuyer and C. Lemieux. Recent advances in randomized quasi-Monte Carlo methods. In M. Dror, P. L'Ecuyer, and F. Szidarovszki, editors, *Modeling Uncertainty: an Examination of Stochastic Theory, Methods, and Applications*, pages 419–474. Kluwer Academic Publishers, Boston, 2002.

[53] X. Lin. *The Hilbert Transform and its Applications in Computational Finance*. PhD thesis, University of Illinois at Urbana-Champaign, Urbana, IL, 2010.

[54] E. Lukacs. *Characteristic Functions*. Charles Griffin & Company Limited, London, second edition, 1970.

[55] M. Marazzi and J. Nocedal. Wedge trust region methods for derivative free optimization. *Mathematical Programming*, 91(2):289–305, 2002.

[56] R. C. Merton. Theory of rational option pricing. *Bell Journal of Economics and Management Science*, 4(1):141–183, 1973.

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

[58] J. Nocedal and S. Wright. *Numerical Optimization*. Springer-Verlag, Berlin, Germany, second edition, 2006.

[59] M. J. D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. In G. Watson, editor, *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, Germany.

[60] M. J. D. Powell. Uobyqa: unconstrained optimization by quadratic approximation. *Mathematical Programming*, 92(3):555–582, 2002.

[61] M. Reed and B. Simon. *Methods of Modern Mathematical Physics, II, Fourier Analysis, Self-Adjointness*. Academic Press, San Diego, California, 1975.

[62] C. Ribeiro and N. Webber. A Monte Carlo method for the normal inverse Gaussian option valuation model using an inverse Gaussian bridge. *Working paper*, 2003.

[63] J. Rosiński. Series representations of Lévy processes from the perspective of point processes. In O. Barndorff-Nielsen, T. Mikosch, and S. Resnick, editors, *Lévy Processes – Theory and Applications*, pages 401–415. Birkhäuser, Boston, 2001.

[64] W. T. Shaw and J. McCabe. Monte Carlo sampling given a characteristic function: quantile mechanics in momentum space. *Available from http://arxiv.org/abs/0903.1592*, 2009.

[65] F. Stenger. *Numerical Methods Based on Sinc and Analytic Functions*. Springer-Verlag, Berlin, Germany, 1993.

[66] T.H.Rydberg. The normal inverse gaussian Lévy process: simulation and apprximation. *Stochastic Models*, 13(4):887–910, 1997.

[67] A. Vaz and L. Vicente. A particle swarm pattern search method for bound constrained global optimization. *Journal of Global Optimization*, 39:197–219, 2007.

[68] A. Černý and I. Kyriakou. An improved convolution algorithm for discretely sampled asian options. *Quantitative Finance*, 11(3):381–389, 2011.

[69] J. Vecer. Unified pricing of asian options. *Risk*, 15(6):113–116, 2002.