Imperial College London

–

Department of Mathematics

Applied Mathematics MSc project

Stationary densities of stable Lévy flights in external potentials

Author:
Mathilde Léval

Supervisor:
Dr. Yanghong Huang

September 4, 2015

Summary

Lévy flights in external potentials are studied. The aim of this project is to provide numerical estimations of the stationary densities of such systems, i.e. estimating the solutions of the stationary Fokker-Planck equation. Some exact results can be derived in the case of symmetric Lévy flights and certain types of potentials. Here symmetric as well as asymmetric Lévy flights will be considered. The method developed consists in solving numerically the corresponding stochastic differential equations by an implicit Euler scheme, and using a large number of realisations to estimate the steady-state density. First, the method will be tested on cases where exact or approximated results are available to compare with our own results. Then, the method will be used to estimate the stationary densities on cases with no known results.

Acknowledgments

I would like to express my gratitude to Dr. Huang, my supervisor, for giving me the opportunity to work on this project and for his very valuable guidance and help throughout the development of this work. I also wish to thank my family and friends for their support and encouragement during my studies.

Contents

List of Figures

1 Introduction

Stochastic processes constitute a very powerful modelisation tool for areas such as signal processing, chemistry, quantitative finance, statistical mechanics, meteorology or economics, where the high number of parameters and data makes it impractical to deterministically describe the system. The study of stochastic processes originates in the 19th century with Brownian motion. This object was first mathematically described by Thorval N. Thiele in order to model particles in suspension in a fluid, as the botanist Robert Brown had observed under his microscope. Then, in the early 20th century Brownian motion began to interest several scientists (as Louis Bachelier and Albert Einstein) because of the possible applications such objects could have in finance or in physics. The prospect of applying randomness to electrical noise inspired Norbert Wiener to theorise a proper definition of Brownian Motion, which is now also called Wiener process or Standard Brownian Motion. As the theory of stochastic processes expanded, the mathematician Paul Lévy studied a family of stochastic processes that present independent and stationary increments, are continuous in probability and almost surely *càdlàg* (right-continuous with left limits). This important class of stochastic processes form a generalisation of the Wiener process and are now known as Lévy processes. One important subcategory of these processes are α-stable stochastic processes, which we will present and study in this thesis.

The 20th century also saw the developpement of stochastic calculus, mainly through the works of Andreï Kolmogorov and Kiyoshi Itō. This field introduces the very useful objects that are stochastic differential equations. They consist of differential equations with an additional term that represents the randomness of the system. At first, this randomness term was modelled via the Standard Brownian Motion. This proved to be a great tool for modelling financial markets, as Fischer Black, Myron Scholes and Robert C. Merton did in their theory of option pricing. However, it has been noticed that using Lévy processes (also called *jump processes* in this context) can be more appropriate than Brownian Motion to model certain systems; in particular when very large increments (or jumps) can be observed.

In statistical mechanics, an important equation linked to stochastic processes emerged, the Fokker-Planck equation, named after the physicists Adriaan Fokker and Max Planck. This equation was first derived to describe the temporal behaviour of the probability density function of a particle's velocity under the influence of an external potential and random forces. However, the Fokker-Planck equation can be used to describe the time evolution of probability density functions in different types of contexts. In this project we will specifically study the steady-state solutions of such probability density functions, i.e. solutions of the stationary Fokker-Planck equation. Traditionally, the Fokker-Planck equation is derived from stochastic differential equations involving Brownian motion. But, a generalisation, called the fractional Fokker-Planck equation, can be derived using equations driven by α-stable Lévy processes, which will be the object of our study. The precise goal of this project is to estimate numerically steady-state solutions of the fractional Fokker-Planck equation for certain types of external potentials.

To achieve this goal, the method will be to numerically solve the desired stochastic differential equation a very high number of times and to realise a density estimation given the final states of all the numerical solutions of the equation. In order to choose good parameters in the numerical scheme and in the density estimation, we will first apply the method to cases where analytical solutions of the stationary Fokker-Planck equation are available. Naturally, we will start by studying the Brownian motion case, for which we will have closed-form solutions to compare

to our estimations. Then, we will apply the algorithms to equations driven by symmetric Lévy motions. In this case, some exact solutions can be found, and some can be numerically approximated by the fast Fourier transform algorithm. Finally, equations driven by asymmetric Lévy motions, for which there are no exact solutions (in regular space or in Fourier space), will be studied. In order to be able to carry out such computations, a preliminary study of α-stable Lévy motion and their simulation is necessary, as well as the derivation and solving of the regular and stationary Fokker-Planck equations.

2 The α-stable Lévy motions and their simulation

2.1 α-stable Lévy random variables

2.1.1 Definition and first properties

Before defining the standard α-stable Lévy motions we have to define α-stable random variables. A random variable X is said to be stable if (according to [8]), for two independent copies X_1 and X_2 of X and any $a > 0$, $b > 0$, there exists $c > 0$ and $d \in \mathbb{R}$ such that $aX_1 + bX_2 \sim cX + d$. There are several ways to define α-stable random variables. The most common one can be found in [8], where they are defined via their characteristic function $\phi(\theta) = \mathbb{E}[e^{i\theta X}]$ in the following way: it involves four parameters; $\alpha \in (0, 2]$ which is the stability index (i.e. the random variables are stable *with respect to* α), $\beta \in [-1, 1]$ the skewness parameter, $\sigma > 0$ the scale parameter and $\mu \in \mathbb{R}$ the shift. Thus we call such random variables α-stable random variables, and they have a characteristic function of the form:

$$\phi(\theta) = \exp\left(-\sigma^\alpha |\theta|^\alpha (1 - i\beta \mathrm{sgn}(\theta)\Phi) + i\mu\theta\right),$$

where Φ is defined by:

$$\Phi = \begin{cases} \tan(\alpha\pi/2) & \text{if} \quad \alpha \neq 1 \\ -\frac{2}{\pi}\log(|\theta|) & \text{if} \quad \alpha = 1 \end{cases}.$$

When $\beta = 0$ the random variable is symmetric and when $\beta = \pm 1$ the random variable is said to be *totally skewed* to the right or to the left. We will use the notation $X \sim S_\alpha(\sigma, \beta, \mu)$ to denote an α-stable random variable with parameters α, σ, β, μ. And, to shorten the notation of the simplest case, we will denote $S_\alpha(1, 0, 0)$ by S_α.

We can check that this quantitative definition is consistent with the very fist definition we gave of stable random variables. Indeed, consider two independent α-stable random variables $X_1 \sim S_\alpha(\sigma_1, \beta_1, \mu_1)$ and $X_2 \sim S_\alpha(\sigma_2, \beta_2, \mu_2)$ and two positive constants a and b, we have that the characteristic function of $aX_1 + bX_2$ is given by, if $\alpha \neq 1$:

$$\begin{aligned} \mathbb{E}[e^{i\theta(aX_1+bX_2)}] &= \mathbb{E}[e^{i\theta aX_1}]\mathbb{E}[e^{i\theta bX_2}] \\ &= \phi_{X_1}(a\theta)\phi_{X_2}(b\theta) \\ &= \exp\left(-(a^\alpha\sigma_1^\alpha + b^\alpha\sigma_2^\alpha)|\theta|^\alpha\left(1 - i\frac{a^\alpha\sigma_1^\alpha\beta_1 + b^\alpha\sigma_2^\alpha\beta_2}{a^\alpha\sigma_1^\alpha + b^\alpha\sigma_2^\alpha}\mathrm{sgn}(\theta)\Phi\right) + i(a\mu_1 + b\mu_2)\theta\right). \end{aligned}$$

Which means that $aX_1 + bX_2$ is the α-stable random variable:

$$S_\alpha\left(((a\sigma_1)^\alpha + (b\sigma_2)^\alpha)^{1/\alpha}, \frac{a^\alpha\sigma_1^\alpha\beta_1 + b^\alpha\sigma_2^\alpha\beta_2}{a^\alpha\sigma_1^\alpha + b^\alpha\sigma_2^\alpha}, a\mu_1 + b\mu_2\right).$$

In the case where $\alpha = 1$, one can see that the formula is similar but the mean has some extra terms due to the form of Φ. Precisely we have that:

$$aX_1 + bX_2 \sim S_1\left(a\sigma_1 + b\sigma_2, \frac{a\sigma_1\beta_1 + b\sigma_2\beta_2}{a\sigma_1 + b\sigma_2}, a\mu_1 + b\mu_2 - \frac{2}{\pi}(\sigma_1\beta_1 a\log(a) + \sigma_2\beta_2 b\log(b))\right).$$

This proves the stability with respect to α of the above definition of α-stable random variables. The previous calculations allows us to derive an important arithmetic property of these variables, which is that, if $Y \sim S_\alpha(1, \beta, 0)$ then:

$$X = \begin{cases} \sigma Y + \mu & \text{if} \quad \alpha \neq 1 \\ \sigma Y + \mu + \beta \frac{2}{\pi} \sigma \log(\sigma) & \text{if} \quad \alpha = 1 \end{cases},$$

is such that $X \sim S_\alpha(\sigma, \beta, \mu)$.

2.1.2 Probability density functions

According to the inversion theorem for characteristic functions, if a random variable X has an integrable characteristic function $\varphi(\theta)$, then X has a probability density function and it is given by:

$$f(x) = \frac{1}{2\pi} \int_{\mathbb{R}} \varphi(\theta) e^{-i\theta x} \, \mathrm{d}\theta.$$

In the case of α-stable random variables, since the scale parameter σ is strictly positive, it is clear that the characteristic function is integrable. Thus, we know that all α-stable random variables have a probability density function $f(\alpha, \beta, \sigma, \mu; x)$.

According to [8] and [10], one can show, using a series representation of α-stable random variables, that the densities have the asymptotic behaviour, for $\alpha < 2$,

$$f(x) \sim \frac{1}{|x|^{\alpha+1}}.$$

From this we can see the main difference between α-stable random variables (with $\alpha < 2$) and Gaussian random variables, which is that non-Gaussian α-stable random variables all have infinite variances. More precisely, we can see that for the quantity $\mathbb{E}[|X|^p]$ to be finite, p must be strictly less than α.

Furthermore, all these densities can be expressed, as shown in [5], using Fox's H function in the following way (assuming $\mu = 0$ for simplicity):

$$f(\alpha, \beta, \sigma, 0; x) = \frac{1}{\alpha x} H^{1,2}_{3,3}\left[\frac{1}{x} \middle| \begin{matrix} (0, 1/\alpha) & (0, 1) & (0, \frac{\alpha-\beta}{2\alpha}) \\ (0, 1/\alpha) & (0, \sigma/\alpha) & (0, \frac{\alpha-\beta}{2\alpha}) \end{matrix}\right] \quad \text{if} \quad \alpha < \sigma,$$

$$f(\alpha, \beta, \sigma, 0; x) = \frac{1}{\alpha x} H^{2,1}_{3,3}\left[\frac{1}{x} \middle| \begin{matrix} (1, 1/\alpha) & (1, \sigma/\alpha) & (1, \frac{\alpha-\beta}{2\alpha}) \\ (1, 1/\alpha) & (1, 1) & (1, \frac{\alpha-\beta}{2\alpha}) \end{matrix}\right] \quad \text{if} \quad \alpha > \sigma,$$

and the case $\alpha = \sigma$ can be reduced to:

$$f(\alpha, \beta, \sigma, 0; x) = \frac{1}{\pi} \frac{x^{\alpha-1} \sin(\pi(\alpha - \beta)/2)}{1 + 2x^\alpha \cos(\pi(\alpha - \beta)/2) + x^{2\alpha}}.$$

Which can be seen as a generalisation of the usual symmetric Cauchy density. In the general case where $\mu \neq 0$, the density is given by $f(\alpha, \beta, \sigma, \mu; x) = f(\alpha, \beta, \sigma, 0; x - \mu)$.

Fox defined the H function in [4] as the integral:

$$H_{p,q}^{m,n}\left[z\left|\begin{matrix}(a_j,\alpha_j)_{j=1\ldots p}\\(b_j,\beta_j)_{j=1\ldots q}\end{matrix}\right.\right]=\frac{1}{2\pi i}\int_T\frac{\prod_{j=1}^m\Gamma(b_j+\beta_j s)\prod_{j=1}^n\Gamma(1-a_j-\alpha_j s)}{\prod_{j=m+1}^q\Gamma(1-b_j-\beta_j s)\prod_{j=n+1}^p\Gamma(a_j+\alpha_j s)}z^{-s}\mathrm{d}s,$$

where the contour T is a contour seperating the poles of $\Gamma(b_j+\beta_j s)$ from the poles of $\Gamma(1-a_j-\alpha_j s)$.

This special function is very interesting but does not generally have an analytic expression which can be computable. Thus it is not possible in the general case to find an analytic expression for the densities of α-stable random variables. Among exceptions, three notable ones are:

- $S_2(\sigma,0,\mu)$ which has a Gaussian distribution,

$$f(x)=\frac{1}{2\sigma\sqrt{\pi}}\exp\left(-\frac{(x-\mu)^2}{4\sigma^2}\right). \tag{5}$$

  Indeed, in this case the characteristic function is given by:

$$\phi(\theta)=e^{-\sigma^2\theta^2+i\mu\theta}.$$

  Which verifies the following differential equation:

$$\phi'(\theta)=-2\sigma^2\theta\phi(\theta)+i\mu\phi(\theta).$$

  So, by applying the Fourier transform $\mathcal{F}$ we get:

$$\mathcal{F}(\phi'(\theta))(x)=-2\sigma^2\mathcal{F}(\theta\phi(\theta))(x)+i\mu f(x),$$

  i.e.

$$ixf(x)=2\sigma^2 if'(x)+i\mu f(x),$$

  which has the simple form:

$$f'(x)=\frac{x-\mu}{2\sigma^2}f(x).$$

  So, there exists a constant K such that:

$$f(x)=K\exp\left(\frac{(x-\mu)^2}{4\sigma^2}\right).$$

  Finally, by using the definition of the Fourier transform we note,

$$f(\mu)=\frac{1}{2\pi}\int_{\mathbb{R}}e^{-\sigma^2\theta^2}\mathrm{d}\theta=\frac{1}{2\sigma\sqrt{\pi}},$$

  which finishes to prove equation (5).

- $S_1(\sigma,0,\mu)$ which has a Cauchy distribution,

$$f(x)=\frac{\sigma}{\pi\left((x-\mu)^2+\sigma^2\right)}. \tag{6}$$

Indeed, in this case we can straightforwardly compute:

$$\begin{aligned} f(x) &= \frac{1}{2\pi}\int_{\mathbb{R}} e^{-\sigma|\theta|+i(\mu-x)\theta}\mathrm{d}\theta \\ &= \frac{1}{2\pi}\left(\int_{-\infty}^{0} e^{\theta(\sigma+i(\mu-x))}\mathrm{d}\theta + \int_{0}^{\infty} e^{\theta(-\sigma+i(\mu-x))}\mathrm{d}\theta\right) \\ &= \frac{1}{2\pi}\left(\frac{1}{\sigma+i(\mu-x)} + \frac{1}{\sigma-i(\mu-x)}\right) \\ &= \frac{\sigma}{\pi\left((x-\mu)^2+\sigma^2\right)}, \end{aligned}$$

which proves equation (6).

- $S_{1/2}(\sigma, 1, \mu)$ which has a Lévy distribution,

$$f(x) = \begin{cases} \left(\frac{\sigma}{2\pi}\right)^{1/2}(x-\mu)^{-3/2}\exp\left(-\frac{\sigma}{2(x-\mu)}\right) & \text{if} \quad x \in (\mu, \infty) \\ 0 & \text{if} \quad x \in (-\infty, \mu] \end{cases}.$$

For any other case, we can compute numerical approximations of density values by using the Cooley-Tukey Fast Fourier Transform algorithm which is a Matlab built-in function. To do so, we must choose the number N of points (which should be even) and the range $[-K, K]$ of the spectrum we want to consider.

Then, we should build a vector containing the values of the characteristic function at the points $k_j = -K + \frac{2K(j-1)}{N}$ with $j = 1, \ldots, N$. Then we will compute the probability density on the range $[-L, L]$, with $L = \frac{N\pi}{2K}$, at the points $x_j = -L + \frac{2L(j-1)}{N}$. Thus, the approximation we want to compute is the following:

$$\begin{aligned} f(x_j) &= \frac{1}{2\pi}\sum_{n=1}^{N}\phi(k_n)e^{-ik_n x_j}\Delta k \\ &= \frac{K}{\pi N}\sum_{n=1}^{N}\phi(k_n)\exp\left(-i\left(-K+\frac{2K(n-1)}{N}\right)\left(-L+\frac{2L(j-1)}{N}\right)\right) \\ &= \frac{1}{2L}\sum_{n=1}^{N}\phi(k_n)\exp\left(-iN\frac{\pi}{2}+\pi i(j-1)+\pi i(n-1)-\frac{2\pi i}{N}(n-1)(j-1)\right). \end{aligned}$$

But, the formulæ that Matlab uses to compute the approximated values X_n of the Fourier transform of the values x_j in the `fft` and `ifft` functions are:

$$X_n = \sum_{j=1}^{N} x_j \exp\left(-\frac{2\pi i}{N}(n-1)(j-1)\right),$$

$$x_j = \frac{1}{N}\sum_{n=1}^{N} X_n \exp\left(\frac{2\pi i}{N}(n-1)(j-1)\right).$$

There are also the functions `fftshift` and `ifftshift` which do the same calculations but rearrange the outputs so that the zero-frequency component is at the center. So, we can see that, given a vector `pk` containing the values of the characteristic functions, the way to get the wanted approximated values of the probability density `fx` explicited previously is to use the following line of code:

```
fx = real(ifftshift(fft(ifftshift(pk))))/(L*2);
```

The real part is taken as some small numerical errors returning complex values can occur. Also, the vector `pk` will be previously obtained using:

```
k = [0:N-1]*dk-K;
pk = exp(-(abs(sigma*k).^(alpha)).*(1-beta*1i*sign(k)*tan(alpha*pi/2)));
```

assuming $\alpha \neq 1$ and $\mu = 0$.

2.2 Simulation methods

2.2.1 Symmetric case

To generate normally distributed points, one common way is to use the Box-Muller method, which performs a transformation from uniformly distributed points. Given two independent random variables $U, V \sim \mathcal{U}(0,1)$, the random variables given by:

$$Z_1 = \sqrt{-2\ln(U)}\cos(2\pi V), \tag{16}$$

$$Z_2 = \sqrt{-2\ln(U)}\sin(2\pi V), \tag{17}$$

are normally distributed and independent. According to [9], these formulæ can be derived by noting that the 2-dimensional Gaussian probability density verifies, for x, y in the unit circle:

$$\frac{1}{2\pi}e^{-\frac{x^2+y^2}{2}}\mathrm{d}x\,\mathrm{d}y = \left(\frac{1}{2}e^{-s/2}\mathrm{d}s\right)\left(\frac{1}{2\pi}\mathrm{d}\theta\right),$$

with $s = r^2 = x^2 + y^2$ and $\theta = \arg(x, y)$, the angle between the points x and y. So, the polar coordinates (R, Θ) of the normally distributed points in the unit circle (X, Y) are such that R^2 follows an exponential law of parameter $1/2$ and Θ is normally distributed in $[0, 2\pi]$, i.e. $R^2 \sim -2\ln(U)$, by inverse transform sampling, and $\Theta \sim 2\pi V$. Then, by using the fact that $X = R\cos(\Theta)$ and $Y = R\sin(\Theta)$ we obtain the formulæ(16) and (17).

A generalisation of this transform for symmetric α-stable random variable is presented in [2] and will be the method we will use to simulate symmetric α-stable variables. Given two random variables $V \sim \mathcal{U}(-\frac{\pi}{2}, \frac{\pi}{2})$ and $W \sim \mathcal{U}(0,1)$, the random variable given by:

$$X = \frac{\sin(\alpha V)}{(\cos(V))^{1/\alpha}}\left(\frac{\cos((1-\alpha)V)}{-\ln(W)}\right)^{\frac{1-\alpha}{\alpha}}, \tag{18}$$

is such that $X \sim S_\alpha$.

Then for any $\sigma > 0$ and $\mu \in \mathbb{R}$, we have $Y = \sigma X + \mu \sim S_\alpha(\sigma, 0, \mu)$. The following Matlab function will be used to simulate S_α:

```
function S = alpha_stable_sym(alpha, N)

% Simulation of N realisations of an alpha-stable stochastic process of the ...
    type S(1,0,0) (mean zero, scale parameter 1, symmetric)
% alpha must be in (0, 2]

% V uniformly distributed on (-pi/2, pi/2)
% W uniformly distributed on (0,1)
V = -pi/2 + pi*rand(N,1);
W = rand(N,1);

% X following S_alpha(1,0,0)
alphabis = (1-alpha)/alpha;
X1 = (sin(alpha.*V))./((cos(V)).^(1/alpha));
X2 = ((cos((1-alpha).*V))./(-log(W))).^(alphabis);
S = X1.*X2;
```

2.2.2 Asymmetric case

A generalisation of formula (18) is given in [2] in order to simulate skewed α-stable variables. Given two random variables $V \sim \mathcal{U}(-\frac{\pi}{2}, \frac{\pi}{2})$, $W \sim \mathcal{U}(0,1)$ and $\beta \in [-1,1]$,

$$Y = D_{\alpha,\beta} \frac{\sin(\alpha(V + C_{\alpha,\beta}))}{(\cos V)^{1/\alpha}} \left(\frac{\cos(V - \alpha(V + C_{\alpha,\beta}))}{-\log(W)} \right)^{\frac{1-\alpha}{\alpha}},$$

is such that $Y \sim S_\alpha(1, \beta, 0)$, with:

$$C_{\alpha,\beta} = \frac{\arctan(\beta \tan(\alpha\pi/2))}{1 - |1 - \alpha|},$$

$$D_{\alpha,\beta} = (\cos(\arctan(\beta \tan(\alpha\pi/2))))^{-\frac{1}{\alpha}}.$$

This formula is properly defined only when $\alpha \neq 1$. In order to use $\alpha = 1$ we can refer to [8], where the formula is extended to the case $\alpha = 1$ by computing:

$$Y = \frac{2}{\pi} \left(\left(\frac{\pi}{2} + \beta V \right) \tan(V) - \beta \log \left(-\frac{\frac{\pi}{2} \log(W) \cos(V)}{\frac{\pi}{2} + \beta V} \right) \right).$$

With V and W defined as above, $Y \sim S_1(1, \beta, 0)$.

From there we can simulate any Lévy motion. Indeed, for any given $\alpha \in (0, 2]$, $\beta \in [-1, 1]$, $\sigma \geq 0$ and $\mu \in \mathbb{R}$, we can simulate $X \sim S_\alpha(\sigma, \beta, \mu)$ by simulating $Y \sim S_\alpha(1, \beta, 0)$ with the previous method and computing $X = \sigma Y + \mu$; or $X = \sigma Y + \mu + 2\beta\sigma \log(\sigma)/\pi$, if $\alpha = 1$.

However, we can see that, for $\alpha > 1$, this formula might produce complex numbers – as a result of some negative numbers taken to a non-integer power. So, in order to simulate only real α-stable random variables, we will simply ignore the complex results that appear in the simulation. Thus, as to be able to control the number of realisations simulated, the Matlab function will have a recursive loop which ends when the desired number of realisations are all real numbers. The following code, which corresponds to such a Matlab function, will be used in order to simulate $S_\alpha(1, \beta, 0)$:

```
function X = alpha_stable(alpha, beta, N)
% Simulation of N realisation of an alpha-stable Levy motion of the type
% S_alpha(1, beta, 0)

% Test wether the process is symmetric and use the symmetric simulation
% function in that case
if (beta == 0)
    X = alpha_stable_sym(alpha, N);
% Test wether alpha = 1 and use the specific formula in that case
elseif (alpha == 1)
    % V uniformly distributed on (-pi/2, pi/2)
    % W uniformly distributed on (0,1)
    V = -pi/2 + pi*rand(N,1);
    W = rand(N,1);

    X = 2*((0.5*pi + beta.*V).*tan(V) - ...
        beta.*log((-0.5*pi.*log(W).*cos(V))./(0.5*pi + beta.*V)))./pi;

else
    % Computation of the constants
    C = (atan(beta*tan(0.5*alpha*pi)))/(1 - abs(1 - alpha));
    D = (cos(atan(beta*tan(0.5*alpha*pi))))^(-1/alpha);

    % V uniformly distributed on (-pi/2, pi/2)
    % W uniformly distributed on (0,1)
    V = -pi/2 + pi*rand(N,1);
    W = rand(N,1);

    alphabis = (1 - alpha)/alpha;
    X1 = (sin(alpha.*(V + C)))./((cos(V)).^(1/alpha));
    X2 = ((cos(V - alpha.*(V + C)))./(-log(W))).^(alphabis);
    % Simulation of S(1, beta, 0)
    X3 = D.*X1.*X2;

    % Select real results
    RealIndex=zeros(N, 1);
    for k=1:N
        RealIndex(k) = isreal(X3(k));
    end
    % Store real results
    Xreal = X3(logical(RealIndex));

    % Make recursive call if there is not enough real results
    if (length(Xreal)==N)
        X = Xreal;
    else
        X = [Xreal ; alpha_stable(alpha, beta, N - length(Xreal))];
    end
end
```

2.2.3 Results

In order to test this direct simulation method, we will perform simulations of a high number of realisations of α-stable random variables and realise a density estimation, which we will compare to the values obtained by computing the Fourier transform of the characteristic function. To estimate the density from the simulation data, we will use the simplest method described in [2] which is to make a normalised histogram, and to linearly interpolate between the points at

the middle of the histogram's bins. The code used in Matlab in order to obtain the following graphs is presented in the first script of the appendix.

In this script we start by choosing the parameters α, β and σ of the random variable (we restrict ourselves to the case $\mu = 0$ without loss of generality since it only performs a translation), then we compute one million realisations of the random variable, using the function `alpha_stable` described earlier. Then we choose the parameters N and K for the FFT approximation, which fix all the other parameters (Δk, L, Δx). N should be chosen large in order to have a good approximation of the Fourier transform. K can be chosen in order to adjust the wanted space range $[-L, L]$. After the FFT approximation is computed, we create a normalised histogram of the simulated data with a hundred bins on the range $[-L, L]$. The number of bins is chosen to have a good approximation around zero (where the density varies rapidly) and for the linear interpolation to be smooth. Indeed, a very high number of bins generates a linear interpolation that oscillates and which depends heavily on the particular simulation used. Finally, we perform the linear interpolation on the graphs by using the default settings of the function `plot`.

The first test was made on the case: $S_2(1, 0, 0) \sim \mathcal{N}(0, 2)$, whose results are presented in figure 1. The right-side graph shows the estimations in logarithmic scale in order to have a better visibility of the errors in the tails.

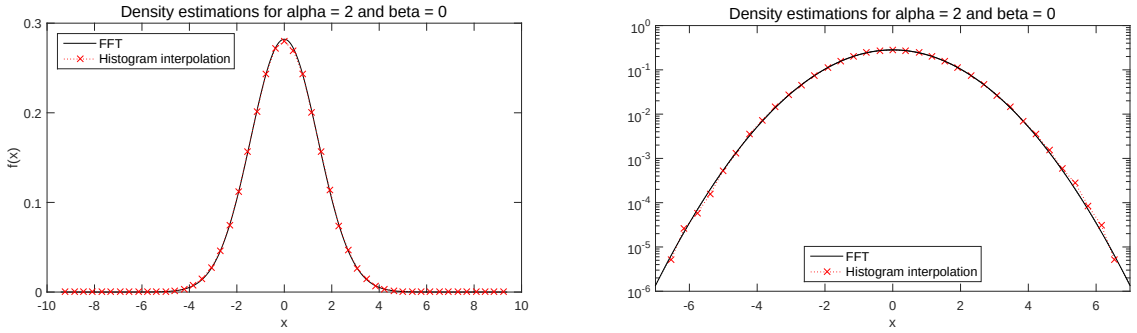

Figure 1 – Density estimations of $S_2(1, 0, 0)$ in linear and logaritmic scales

The simulation allows us to produce a very good approximation of the probability density. Then, we tested the impact of the parameter α on the quality of the simulation. The figures 2 and 3 present the results obtained by keeping $\beta = 0$ and $\sigma = 1$ and progressively decreasing α. The results are shown in logarithmic scale in figure 3 for better visibility at the tails.

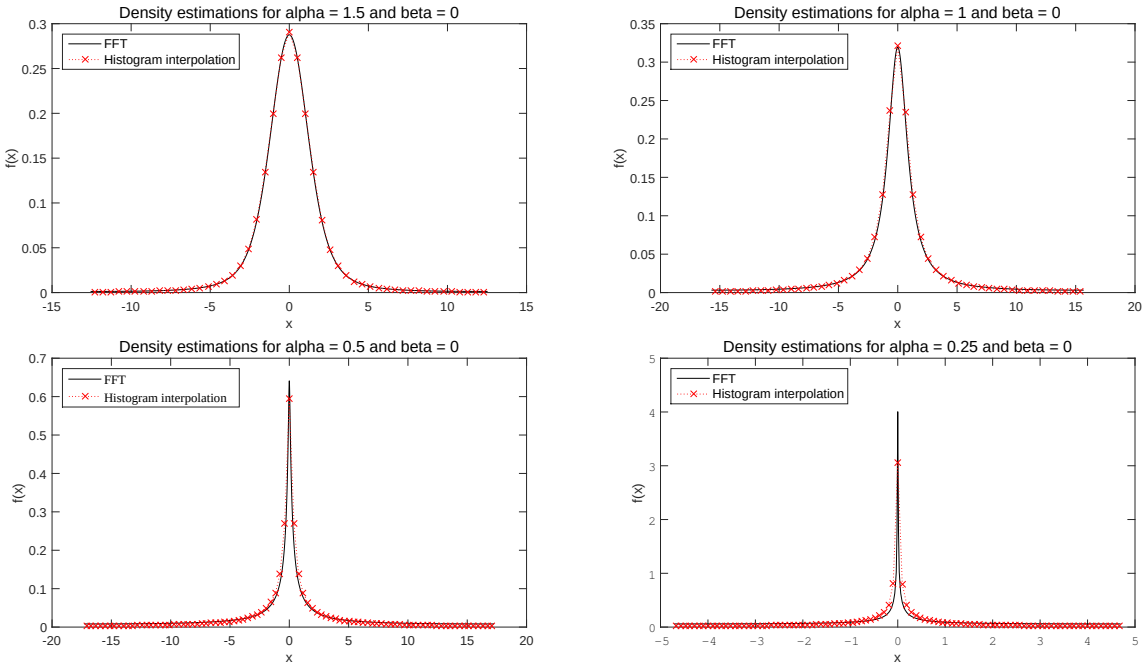

Figure 2 – Density estimations of symmetric α-stable random variables

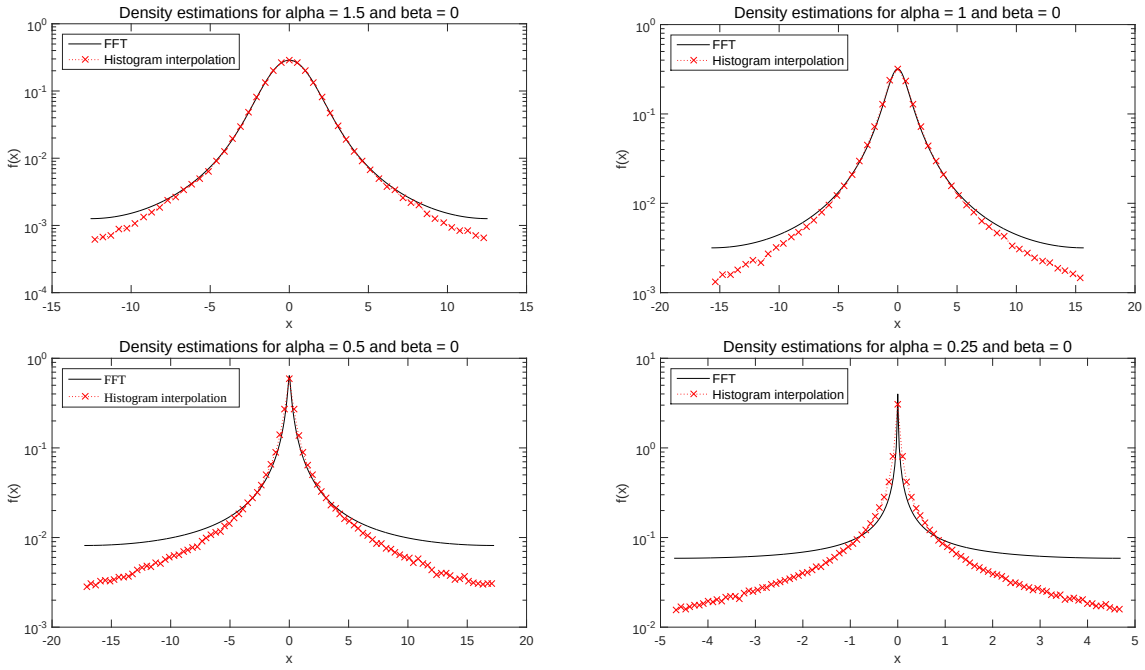

Figure 3 – Density estimations of symmetric α-stable random variables in logarithmic scale

As α goes to zero, the density estimation loses precision in the tails. The method is less accurate when it comes to simulating random variables that are very heavy-tailed. However, the estimation remains quite correct around zero, even for small values of α.

Finally, the effect of the parameter β has been tested. The figures 4 and 5 present the effect of β on the quality of the simulation. The parameter β can take values in $[-1, 1]$ but since its effect is symmetrical, we will only test a few values of β in $[0, 1]$. The following graphs show the results when $\alpha = 0.75$ and β takes values in $\{0.25, 0.5, 0.75, 1\}$.

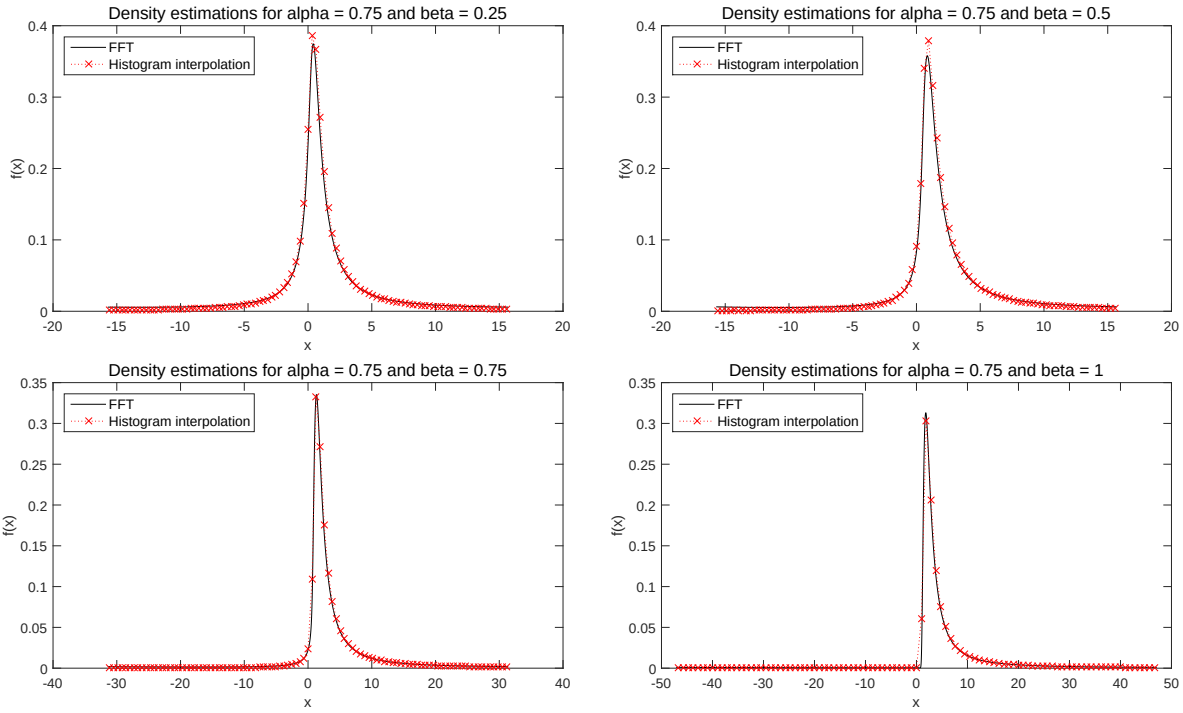

Figure 4 – Density estimations of asymetric α-stable random variables with $\alpha < 1$

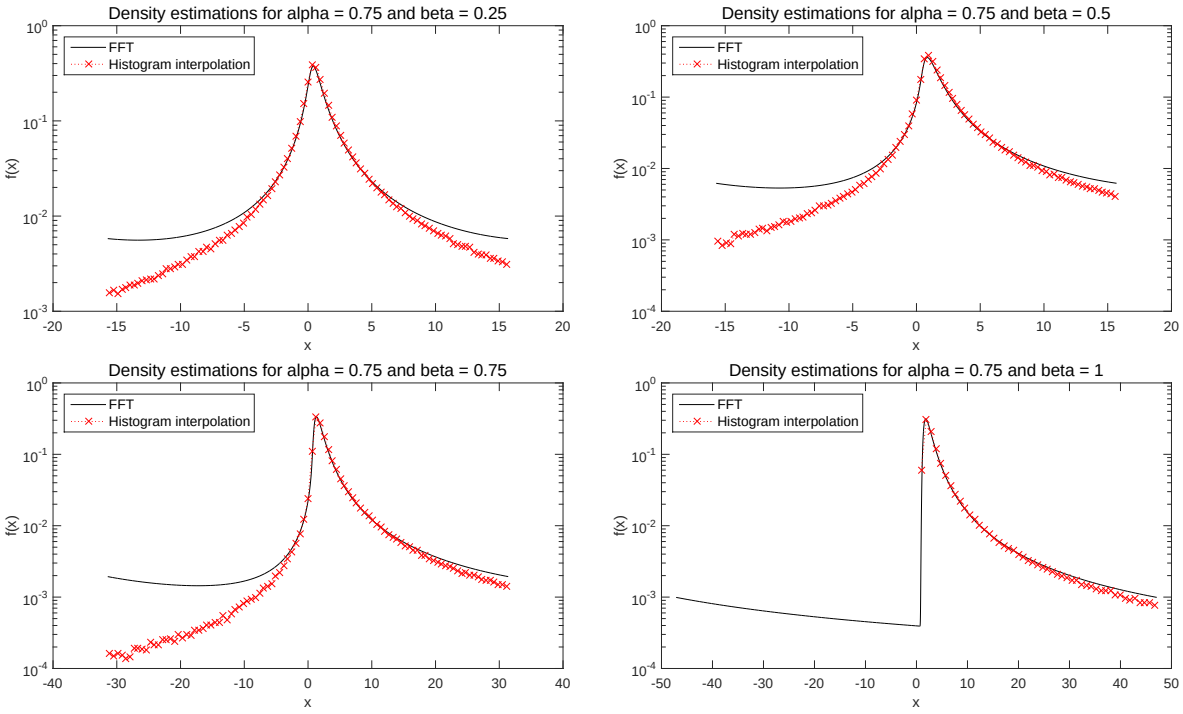

Figure 5 – Density estimations of asymetric α-stable random variables with $\alpha < 1$ in logarithmic scale

Once again, the results are also shown in logarithmic scales in order to get a better visibility of the errors on figure 5. It is interesting to note that in this case, the simulation performs better than the FFT on the non-preferred side. Indeed, for $\beta = 1$, the probability density is supposed to be zero for all negative values. While the FFT returns values of the order of 10^{-3} at -50, the simulation actually does not produce any negative data point. It's also interesting to see that, on the preferred side, the simulation generates data that agrees more with the FFT as β increases to 1.

In order to test the effect of ignoring the complex results that are appearing in the simulation method when $\alpha > 1$ and $\beta \neq 0$ we present here a few density comparisons of these cases for different values of α and β.

We can see clearly that the algorithm performs density estimations that can be quite different from reality. Indeed, choosing to simply ignore the complex values that arise in the formula is easy to implement but introduces a bias that alter the quality of the simulation. So, such Lévy motions must be used carefully in other computations.

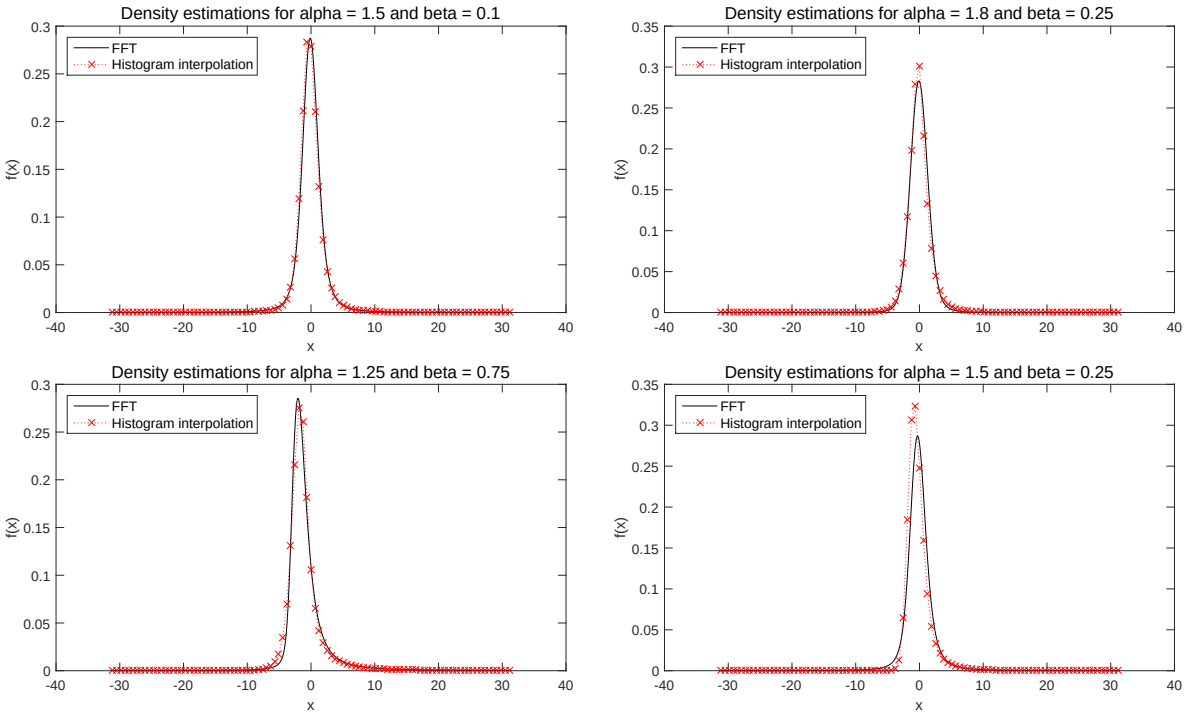

Figure 6 – Density estimations of asymetric α-stable random variables with $\alpha > 1$

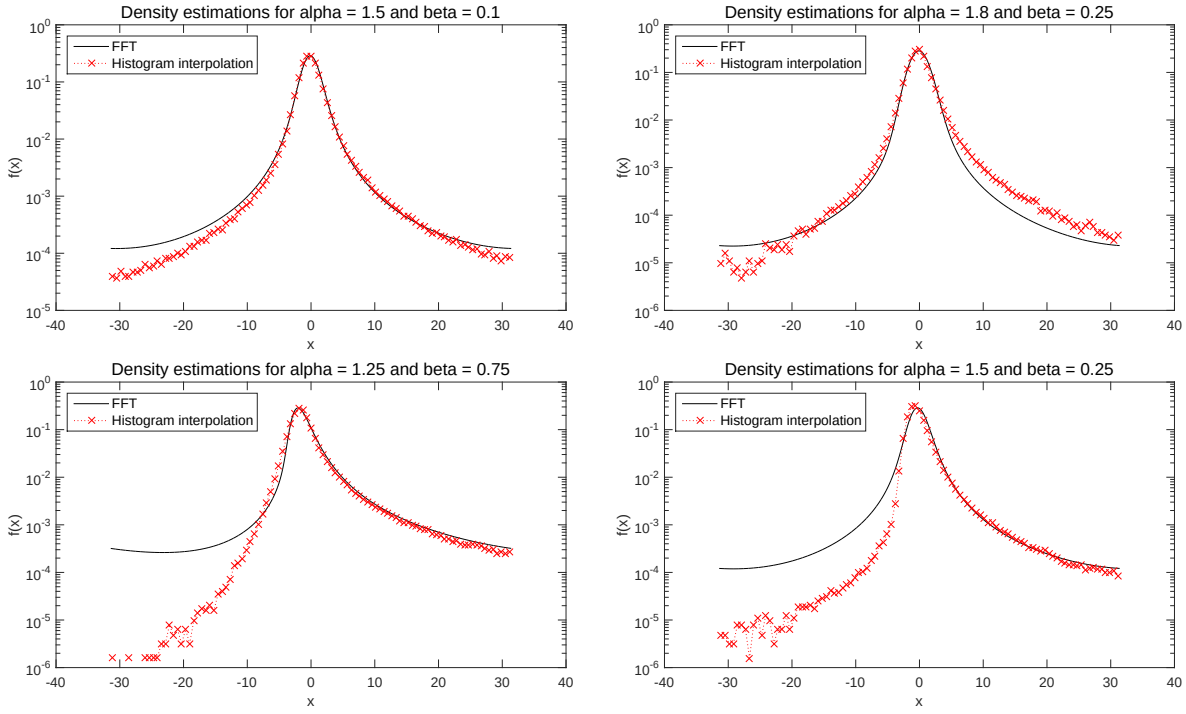

Figure 7 – Density estimations of asymetric α-stable random variables with $\alpha > 1$ in logarithmic scale

2.3 The Standard α-stable Lévy motions

2.3.1 Definition

The standard α-stable Lévy motions (or commonly called Lévy flights), are a generalisation of the standard Brownian motion, and can be used to drive stochastic differential equations. A

stochastic process $\{X(t), t \in \mathbb{R}_+\}$ is a standard α-stable Lévy motion if, for some $\alpha \in (0, 2]$ and $\beta \in [-1, 1]$:

1. $X(0) = 0 \quad a.s.$
2. non-overlapping increments are independent
3. for $0 \leq s < t < \infty, \quad X_t - X_s \sim S_\alpha \left((t-s)^{1/\alpha}, \beta, 0\right)$.

Such motions are $1/\alpha$ self-similar, i.e. if $X(t)$ is an α-stable Lévy motion, then for $c > 0$, $c^{-1/\alpha}X(ct)$ is the same motion. Indeed, the two first conditions are clearly satisfied and $c^{-1/\alpha}X(ct) - c^{-1/\alpha}X(cs) \sim c^{-1/\alpha}S_\alpha((ct-cs)^{1/\alpha}, \beta, 0) \sim S_\alpha((t-s)^{1/\alpha}, \beta, 0)$.

As a comparision, the standard Brownian motion $\{W(t), t \in \mathbb{R}_+\}$ is defined by:

1. $W(0) = 0 \quad a.s.$
2. non-overlapping increments are independent
3. for $0 \leq s < t < \infty$, $W_t - W_s \sim \mathcal{N}(0, t-s)$.

Since $\mathcal{N}(0, (t-s)) \sim \sqrt{2}S_2\left(\sqrt{t-s}, 0, 0\right)$, we see how the Lévy flights consistute a generalisation of the Brownian motion.

2.3.2 Simulation and comparison with Brownian motion

Lévy flights can be easily simulated by an increments approximation, using α-stable random variables, as in the Euler-Maruyama scheme. The following graphs represent Lévy flights obtained with this method, first, figure 8 was obtained using $\beta = 0$ and different values of α, with the case $\alpha = 2$ representing the standard Brownian motion, and then, figure 9 was obtained setting $\alpha = 0.8$ and using different values of β. The values of α chosen to be compared to Brownian motion are relatively close to 2, as to have trajectories with maximal values of the same order. With $\alpha = 1.25$ we can start to see why Lévy motions are often viewed as *flights*, or *jumps*, in opposition of the "walks" Brownian motions are called.

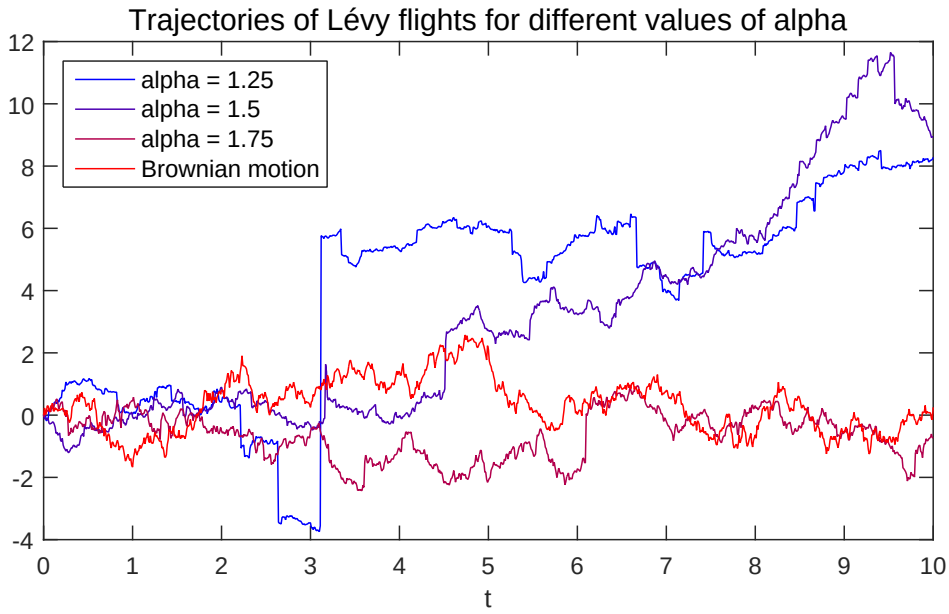

Figure 8 – Trajectories of Lévy flights compared to Brownian motion

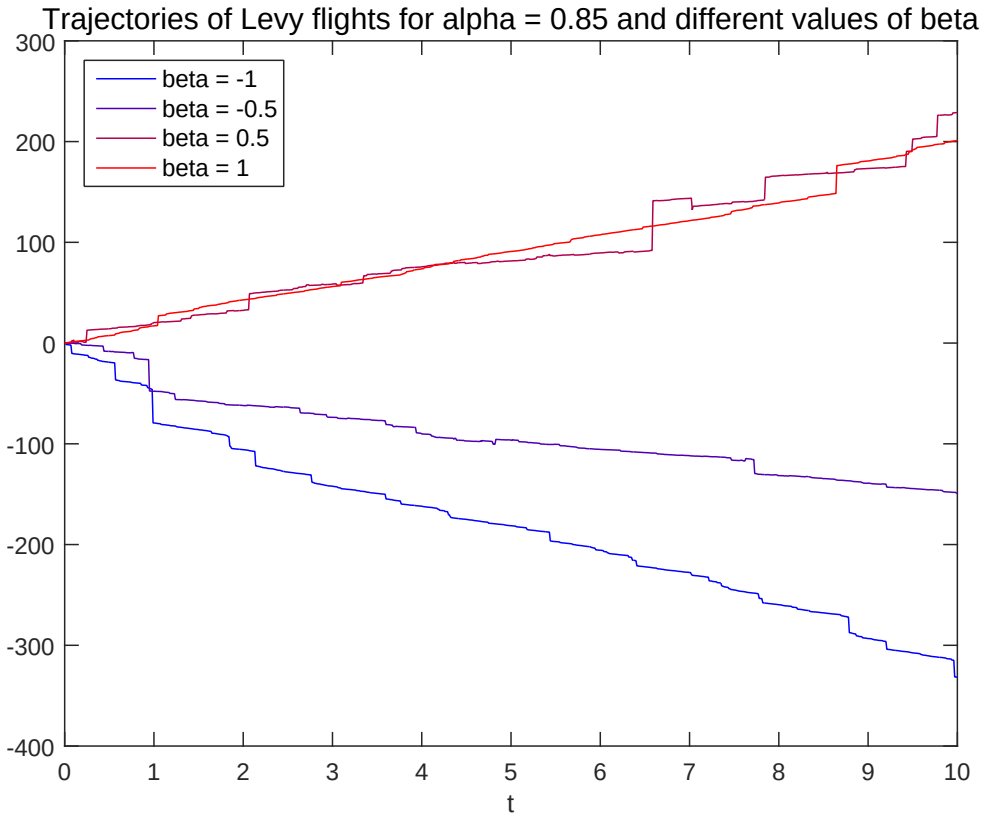

Figure 9 – Trajectories of Lévy flights with β varying

3 The Fokker-Planck equation with an external potential

In this part we will study the Fokker-Planck equation, which describes the behaviour of the transition probability density of a stochastic process, associated with the process $X(t)$ defined by:

$$dX_t = -V'(X_t)dt + dL_{\alpha,\beta}(t), \tag{19}$$

with $L_{\alpha,\beta}(t)$ a standard α-stable Lévy motion with parameters α and β. We will denote the transition probability density by:

$$p(x,t|x_0,t_0) = \mathrm{P}\left(X_t = x | X_{t_0} = x_0\right).$$

The traditional case where $\alpha = 2$ and $\beta = 0$ will be studied first, and the general analytical solution of the stationary Fokker-Planck equation will be derived. Then, the more general fractional Fokker-Planck equation will be derived, as well as some analytical solutions of the stationary equation.

3.1 SDEs driven by Brownian motion

3.1.1 Derivation of the Fokker-Planck equation

Here we study the following stochastic differential equation:

$$dX_t = -V'(X_t)dt + dW_t, \tag{20}$$

with W_t the standard Brownian motion. We will show that the Fokker-Planck equation for the Brownian case is given by:

$$\frac{\partial p}{\partial t}(x,t|x_0,t_0) = \mathcal{L}^* p(x,t|x_0,t_0), \tag{21}$$

with $\mathcal{L}^*$ the adjoint generator of the process X_t. We recall that for the stochastic equation (20) the generator and the adjoint generator of the process are given by, for $f \in C^2(\mathbb{R})$,

$$\mathcal{L}f(x) = -V'(x)f'(x) + f''(x),$$

$$\mathcal{L}^* f(x) = \frac{\partial}{\partial x}\left(V'(x)f(x) + \frac{1}{2}f'(x)\right).$$

In order to derive the Fokker-Planck equation, the quantity $\mathbb{E}_{x_0,t_0}\left[\mathcal{L}f(X_t)\right]$ will be interpreted in two different ways. First, by definition of an expectation and the definition of an adjoint operator:

$$\begin{aligned}
\mathbb{E}_{x_0,t_0}\left[\mathcal{L}f(X_t)\right] &= \mathbb{E}\left[\mathcal{L}f(X_t)|X(t_0) = x_0\right] \\
&= \int_{\mathbb{R}} (\mathcal{L}f)(y) p(y,t|x_0,t_0)\,\mathrm{d}y \\
&= \int_{\mathbb{R}} f(y)(\mathcal{L}^* p)(y,t|x_0,t_0)\,\mathrm{d}y.
\end{aligned}$$

We recall Itō's formula for $f(X_t)$:

$$df(X_t) = f'(X_t)dX_t + f''(X_t)dt = (-V'(X_t)f'(X_t) + f''(X_t))dt + f'(X_t)dW_t,$$

which can be expressed with the generator $\mathcal{L}$:

$$df(X_t) = \mathcal{L}f(X_t)dt + f'(X_t)dW_t.$$

So by using the integral form of the previous formula,

$$f(X_t) - f(X_{t_0}) = \int_{t_0}^{t} \mathcal{L}f(X_s)\,\mathrm{d}s + \int_{t_0}^{t} f'(X_s)\,\mathrm{d}W_s.$$

By applying the expectation operator $\mathbb{E}_{x_0,t_0}$, using the fact that Itō integrals (with respect to Brownian motion) are martingales, and taking derivatives with respect to time we get the following relation:

$$\partial_t \mathbb{E}_{x_0,t_0}\left[f(X_t)\right] = \mathbb{E}_{x_0,t_0}\left[\mathcal{L}f(X_t)\right].$$

And, by using the definition of the expectation, and assuming p satisfies the conditions for Leibniz integral rule,

$$\begin{aligned}\partial_t \mathbb{E}_{x_0,t_0}\left[f(X_t)\right] &= \partial_t \int_{\mathbb{R}} f(y)p(y,t|x_0,t_0)\,\mathrm{d}y \\ &= \int_{\mathbb{R}} f(y)\partial_t p(y,t|x_0,t_0)\,\mathrm{d}y.\end{aligned}$$

So, by using both interpretations of $\mathbb{E}_{x_0,t_0}\left[\mathcal{L}f(X_t)\right]$,

$$\forall f \in C^2(\mathbb{R}), \quad \int_{\mathbb{R}} f(y)\left(\partial_t p(y,t|x_0,t_0) - \mathcal{L}^* p(y,t|x_0,t_0)\right)\,\mathrm{d}y = 0,$$

which is enough to prove the Fokker-Planck equation (21), since probability densities only need to be defined in the almost everywhere sense.

3.1.2 Analytical solution of the stationary Fokker-Planck equation

The aim of this part is to derive the stationary probability density (or invariant measure) ρ_∞ of the process, which solves the stationary Fokker-Planck equation:

$$\mathcal{L}^*\rho_\infty = \frac{\partial}{\partial x}\left(V'(x)\rho_\infty(x) + \frac{1}{2}\rho'_\infty(x)\right) = 0. \tag{29}$$

Since the detailed balance condition,

$$V'(x)\rho_\infty(x) + \frac{1}{2}\rho'_\infty(x) = 0,$$

is solved by:

$$\rho_\infty(x) = \frac{1}{Z}\exp\left(-2V(x)\right) \quad \text{with} \quad Z = \int_{-\infty}^{\infty} e^{-2V(x)}\,\mathrm{d}x, \tag{30}$$

which is clearly a probability density function, this proves X_t has a unique invariant measure given by ρ_∞.

This simple formula will allow us to test our stationary density estimation in the case of stochastic differential equations driven by Brownian motion for any potential V.

3.2 SDEs driven by Standard Lévy motion

3.2.1 Derivation of the fractional Fokker-Planck equation

Now the fractional Fokker-Planck equation will be derived for the stochastic process:

$$dX_t = -V'(X_t)dt + dL_{\alpha,\beta}(t). \tag{31}$$

We will show that in this case, the Fokker-Planck equation is given by:

$$\frac{\partial p}{\partial t}(x,t|x_0,t_0) = \frac{\partial}{\partial x}\left(V'(x)p(x,t|x_0,t_0)\right) - \left[(-\Delta)^{\alpha/2} + \beta\Phi\frac{\partial}{\partial x}(-\Delta)^{(\alpha-1)/2}\right]p(x,t|x_0,t_0). \tag{32}$$

There are several ways to define fractional derivatives, here it is appropriate to consider Riesz's definition of the fractional Laplacian by using the inverse Fourier transform with respect to space $\mathcal{F}^{-1}$:

$$\forall f \in C^2(\mathbb{R}), \quad (-\Delta)^{\alpha/2}(f) = \mathcal{F}^{-1}\left[|k|^\alpha \hat{f}(k)\right].$$

Here, as it is done in [6], we will derive the proof for the case where the stochastic process (31) has stationary and independent increments. In this case we have:

$$p(x,t|x_0,t_0) = p(x-x_0,t-t_0) = p(x,t),$$

by assuming $x_0 = t_0 = 0$ without any loss of generality.

In order to prove equation (32), we will use the characteristic function in the following forms:

$$Z_X(k,t) = \mathcal{F}[p(x,t)] = \mathbb{E}[e^{iX(t)}],$$

$$K_X(k,t) = \log\left(Z_X(k,t)\right).$$

We will also use the incremental characteristic functions $\delta Z_X(k,\delta t|x,t)$ and $\delta K_X(k,\delta t|x,t)$ of the stochastic process $X(t)$, which are defined by:

$$\delta Z_X(k,\delta t) = \mathbb{E}\left[e^{ik(X(t+\delta t)-X(t))}\right],$$

$$\delta K_X(k,\delta t) = \log\left(\delta Z_X(k,\delta t)\right).$$

The stationarity and independence of the increments implies that δK_X has a Taylor expansion of the form:

$$\delta K_X(k,\delta t) = \delta t \sum_{n\in J} \frac{(ik)^n}{n!} C_n + o(\delta t), \tag{33}$$

with C_n constants and J a set of indices. Then, noting that:

$$\begin{aligned}
Z_X(k,t+\delta t) - Z_X(k,t) &= \mathbb{E}[e^{iX(t+\delta t)}] - \mathbb{E}[e^{iX(t)}] \\
&= \mathbb{E}[e^{iX(t)}]\left(\mathbb{E}[e^{i(X(t+\delta t)-X(t))}] - 1\right) \\
&= Z_X(k,t)\left(\delta Z_X(k,\delta t) - 1\right) \\
&= Z_X(k,t)\left(e^{\delta K_X(k,\delta t)} - 1\right) \\
&= Z_X(k,t)\delta K_X(k,\delta t) + o(\delta t),
\end{aligned}$$

and applying the inverse Fourier transform with respect to space we get the following convolution:

$$p(x,t+\delta t)-p(x,t)=\int_{\mathbb{R}}\mathcal{F}^{-1}[\delta K_X(k,\delta t)](x-y)p(y,t)\,\mathrm{d}y+o(\delta t). \tag{40}$$

Also, using equation (33) and the Dirac distribution δ_D,

$$\begin{aligned}\mathcal{F}^{-1}[\delta K_X(k,\delta t)](x) &= \delta t\sum_{n\in J}\frac{C_n}{n!}\mathcal{F}^{-1}[(ik)^n]+o(\delta t)\\ &= \delta t\sum_{n\in J}\frac{C_n}{n!}\delta_D^{(n)}(x)+o(\delta t),\end{aligned}$$

where the nth derivative of the Dirac delta satisfies, for $\varphi\in C^n(\mathbb{R})$,

$$\int_{\mathbb{R}}\delta_D^{(n)}(x-y)\varphi(y)\,\mathrm{d}y=(-1)^n\int_{\mathbb{R}}\delta_D(x-y)\varphi^{(n)}(y)=(-1)^n\varphi^{(n)}(x).$$

Thus, equation (40) becomes:

$$\begin{aligned}p(x,t+\delta t)-p(x,t) &= \delta t\sum_{n\in J}\frac{C_n}{n!}\int_{\mathbb{R}}\delta_D^{(n)}(x-y)p(y,t)\,\mathrm{d}y+o(\delta t)\\ &= \delta t\sum_{n\in J}\frac{(-1)^n}{n!}C_n\frac{\partial^n p}{\partial x^n}(x,t)+o(\delta t).\end{aligned}$$

And, by taking the limit $\delta t\longrightarrow 0$ we get:

$$\frac{\partial p}{\partial t}(x,t)=\sum_{n\in J}\frac{(-1)^n}{n!}C_n\frac{\partial^n p}{\partial x^n}(x,t). \tag{47}$$

This last equation can be generalised to the case where the stochastic process (31) does not have stationary and independent increment, using the Chapman-Kolomogorov identity for general Markov processes which can be written as:

$$p(x,t+\delta t|x_0,t_0)=\int_{\mathbb{R}}p(x,t+\delta t|y,t)p(y,t|x_0,t_0)\,\mathrm{d}y.$$

However, in this case, the coefficients of the Taylor expansion of $\delta K_X(k,\delta t|x,t)$ are no longer constants, and the equation (47) thus becomes:

$$\frac{\partial p}{\partial t}(x,t|x_0,t_0)=\sum_{n\in J}\frac{(-1)^n}{n!}C_n(x,t)\frac{\partial^n p}{\partial x^n}(x,t|x_0,t_0). \tag{48}$$

In order to get from equation (48) to equation (32) we first need to develop δK_X. For a standard Lévy motion L we have by definition:

$$\delta K_L(k,\delta t)=-\delta t\left(|k|^\alpha\left(1-i\beta\frac{k}{|k|}\Phi\right)\right)+o(\delta t).$$

And, given our stochastic differential equation (19):

$$\delta Z_X(k,\delta t|x,t)=\mathbb{E}[e^{i(X(t+\delta t)-x)}|X(t)=x]=e^{-ikV'(x)}\delta Z_L(k,\delta t)+o(\delta t).$$

So,

$$\delta K_X(k,\delta t|x,t)=-\delta t\left(ikV'(x)+|k|^\alpha\left(1-i\beta\frac{k}{|k|}\Phi\right)\right)+o(\delta t),$$

which can be identified as a fractional Taylor expansion. So, by replacing the coefficients in equation (48) we get the fractional Fokker-Planck equation (32).

3.2.2 Analytical solutions of the stationary fractional Fokker-Planck equation

In the case where $\beta = 0$, and for certain types of potentials, it is possible to derive an analytical solution of the stationary Fokker-Planck equation. As in [7], with $V(x) = x^2/2$, we can derive a formula for the solution. Indeed, in this case the Fokker-Planck equation (32) reduces to:

$$\frac{\partial p}{\partial t}(x,t|x_0,t_0) = \frac{\partial}{\partial x}\left(xp(x,t|x_0,t_0)\right) - (-\Delta)^{\alpha/2}\, p(x,t|x_0,t_0).$$

Thus, the stationary Fokker-Planck equation is given by:

$$\frac{\partial}{\partial x}\left(x\rho_\infty(x)\right) - (-\Delta)^{\alpha/2}\,\rho_\infty(x) = 0.$$

In Fourier space, this equation becomes:

$$ik(\mathcal{F}[x\rho_\infty(x)]) - |k|^\alpha \hat{\rho}_\infty(k) = 0,$$

i.e.

$$\frac{\partial \hat{\rho}_\infty}{\partial k}(k) = -\frac{|k|^\alpha}{k}\hat{\rho}_\infty(k),$$

which is solved by:

$$\hat{\rho}_\infty(k) = A\exp\left(-\frac{|k|^\alpha}{\alpha}\right),$$

for some real contant A. As stated in [7], in the general case, the corresponding (normalised) solution in the x-space can be given in terms of Fox's H-function:

$$\rho_\infty(x) = \frac{\pi}{|x|}H^{1,1}_{2,2}\left[\alpha|x|^\alpha \middle| \begin{matrix}(1,1) & (1,\alpha/2)\\ (1,\alpha) & (1,\alpha/2)\end{matrix}\right].$$

But, as for the probability densities of the α-stable random variables, an analytical formula is not always obtainable. However, in the case where $\alpha = 1$ we can obtain a simple formula for ρ_∞ by calculating the inverse Fourier transform of $\hat{\rho}_\infty$ as below:

$$\begin{aligned}\rho_\infty(x) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-|k|}e^{ikx}\mathrm{d}k\\ &= \frac{1}{2\pi}\left(\frac{1}{1-ix}+\frac{1}{1+ix}\right)\\ &= \frac{1}{\pi(x^2+1)}.\end{aligned}$$

For other values of V, we can refer to [1], in which the derivation of the solution in the case where $\alpha = 1$ is presented. For instance, we will use $V(x) = x^4/4$ for which the stationary density is:

$$\rho_\infty(x) = \frac{1}{\pi(x^4-x^2+1)}.$$

4 Numerical simulations and density estimations

4.1 Methods

4.1.1 Numerical simulations of the stochastic differential equations

In order to compute numerical solution of stochastic differential equations driven by Brownian or Lévy motions, we will use the *Split-Step Backward Euler scheme* presented in [3]. Indeed, when the gradient of the potential in the equation (19) is non-linear, using the forward Euler-Maruyama scheme to solve the equation can lead to numerical instability. Hence the use of an implicit scheme instead. In the case of our particular form of stochastic differential equation (19), the Split-Step Backward Euler scheme computes the solution of the equation X_n as well as an intermediate process X_n^* in the following way:

$$\begin{aligned} X_n^* &= X_n - V'(X_n^*)\Delta t, \\ X_{n+1} &= X_n^* + \Delta L_{\alpha,\beta}, \end{aligned} \tag{53}$$

with $\Delta L_{\alpha,\beta} \sim S_\alpha\left((\Delta t)^{1/\alpha}, \beta, 0\right)$.

This method treats the deterministic part of the equation implicitly and the random part explicitly. Except for the case $V(x) = x^2/2$, the first step of the scheme (equation (53)), requires to solve a nonlinear equation, which makes the computations significantly longer than the ones involving the forward Euler-Maruyama scheme (but reduces the risk of numerical instability, thus produces better results).

The three potentials that we are going to study are the following:

- The simple parabolic monostable potential, for which we can get simple analytical solutions of the stationary Fokker-Planck equation,
$$V_1(x) = \frac{x^2}{2}.$$
- The simple quartic monostable potential, for which we can get some analytical solutions in certain cases,
$$V_2(x) = \frac{x^4}{4}.$$
- The quartic double well-potential which is often used to model systems in chemistry or physics,
$$V_3(x) = \frac{x^4}{4} - \frac{x^2}{2}.$$

The first potential we will use in our study is the simple symmetric monostable potential $V(x) = x^2/2$, for which the forward Euler-Maruyama scheme does not present any risk of numerical instability, as the derivative of V is globally Lipschitz. Thus, in order to speed up the computations, when this case is studied, the forward Euler-Maruyama scheme will be used instead of the split-step scheme. The solution X_n is computed explicitly in the following way:

$$X_{n+1} = X_n - V'(X_n)\Delta t + \Delta L_{\alpha,\beta}.$$

However, for the two other potentials, we need to use the split-step backward Euler scheme in order to avoid instability. To solve the non-linear equation (53), the easiest way on Matlab is to use a numerical solver like the functions `roots` (since the equation is polynomial in our case) or `fzero` (for more general cases). But, the use of these function on each realisation, for every time-step of the simulations makes the computations extremely long, such that it is impractical to realise them on a single computer. So, as the equations we are trying to solve in these particular cases are third degree polynomials (which only have one real root), we can find an arithmetic formula of the solution. Moreover, these cubic equations are already in the depressed cubic form, so we can easily use Cordano's method to solve them. For the potential V_2 the equation to solve in $\mathbb{R}$ is:

$$\Delta t X_n^{*3} + X_n^* - X_n = 0,$$

whose solution is given by:

$$X_n^* = \frac{\gamma}{3^{2/3}\sqrt[3]{2}\Delta t} - \frac{\sqrt[3]{2/3}}{\gamma}, \tag{54}$$

with:

$$\gamma = \sqrt[3]{9\Delta t^2 X_n + \sqrt{3}\sqrt{27\Delta t^4 X_n^2 + 4\Delta t^3}}.$$

For the equation involving the potential V_3, the equations is slightly different:

$$\Delta t X_n^{*3} + (1 - \Delta t) X_n^* - X_n = 0,$$

and the solution in $\mathbb{R}$ is:

$$X_n^* = \frac{\Lambda}{3\sqrt[3]{2}\Delta t} - \frac{\sqrt[3]{2}(1 - \Delta t)}{\Lambda}, \tag{55}$$

with:

$$\Lambda = \sqrt[3]{27\Delta t^2 X_n + \sqrt{729\Delta t^4 X_n^2 + 108(1 - \Delta t)^3 \Delta t^3}}.$$

For these solutions to actually be in $\mathbb{R}$, there is a condition on the value of X_n. However, in either case, the condition is that X_n should be greater than a number of the order of -10^9. So, in the context of our study the solutions (54) and (55) are suitable.

4.1.2 Density estimations

To estimate the steady-state density from the numerical solutions of the stochastic differential equation, we will as before use a linear interpolation of an histogram. What is important in order to get a precise estimation is to choose good bins for the histogram. The histogram is created in Matlab by specifying the edges of the bins. Indeed, if we do not specify bins, the ones created by default will not be optimal to our goal in the general case. For instance, the Lévy noises can scatter the data over a range of width 10^5, while most of the weigth will be contained in a much smaller range, so the default histogram will have one bin with almost all the weigth of the distribution around zero, and very small and sparse bins on the rest of the whole range.

Thus we first need to choose the boundaries of the edges. The way used here is to use an interval of the form $[m - L, m + L]$ where m is the median value of the results. L can take different values given the potential used and the value of α. Indeed, for small values of α, the densities are heavy-tailed and thus are interesting to study on a larger range than for values of α close to 2. Also, the potentials V_2 and V_3 will much more concentrate the data in the interval $[-1, 1]$ than the potential V_1.

Finally, when the boundaries are chosen, we need to choose how many bins to take in that interval. An histogram with too little bins will not result in a precise enough estimation, but an histogram with too many bins will result in an estimation that oscillates and that depends heavily on the particular simulation used. When the potential V_1 is used, the density we are trying to estimate will be unimodal, thus a satisfying precision is attainable with about bins. But, the other two potentials will produce bimondal densities, which requires much more bins to be estimate precisely the density around the modes.

4.2 Results

4.2.1 Equations driven by Brownian motion

Here we present the solutions of the traditional stationary Fokker-Planck equation (29) obtained by the exact formula (30) and by linear interpolation of an histogram of the steady-state numerical solutions of the stochastic differential equation (20). To compute the normally distributed increments, we used the `randn` in Matlab which uses the ziggurat algorithm which is significantly faster than the Box-Muller transform. For the potential V_1 and V_2, good results were obtained using 800,000 realisations of the equation and 10,000 time-steps of size $\Delta t = 0.001$. However, for V_3 we needed 1 million realisations for the estimation around the modes to be precise enough. The time-step and the number of steps did not need to be changed, as the potentials did not affect the speed at which the steady-state was obtained.

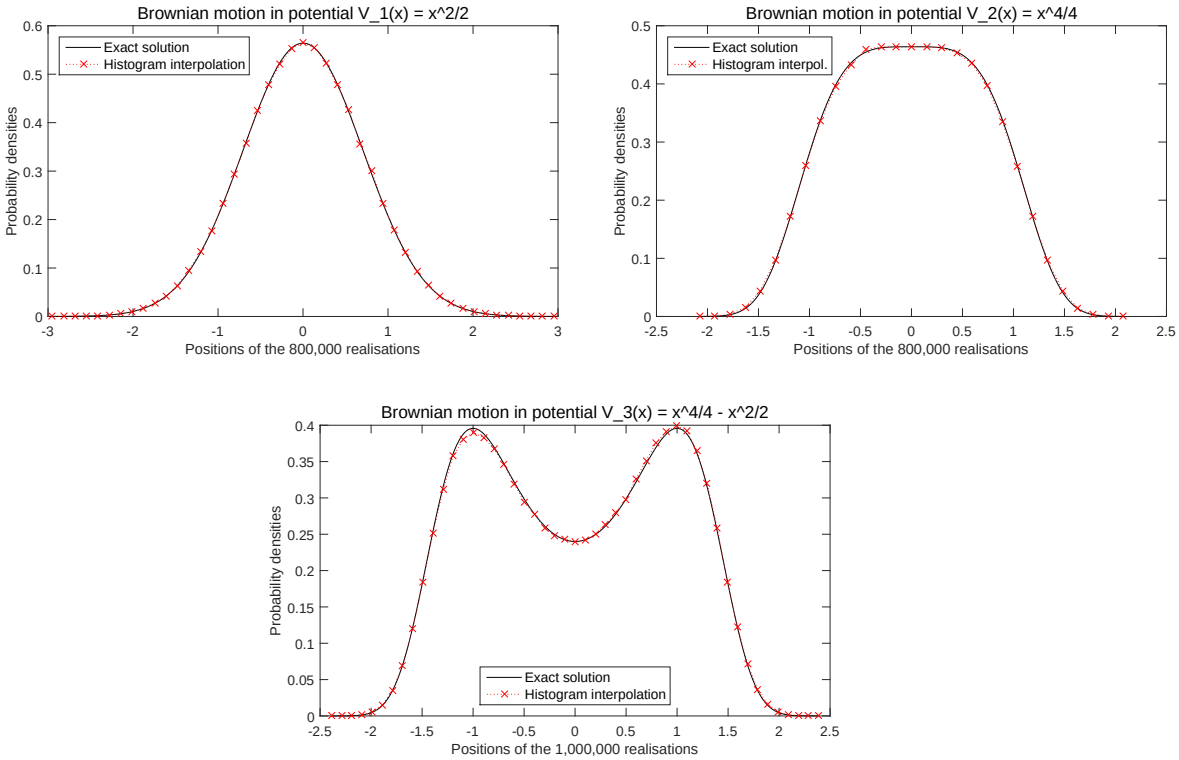

Figure 10 – Theoretical and estimated stationary density functions of Brownian motion in different potentials

4.2.2 Equations driven by symmetric Lévy motions

Quadratic potential $\mathbf{V_1(x) = x^2/2}$ For the stochastic differential equation (19) using this potential and a symmetrical Lévy motion, we can get, for all values of α, an approximate result of the stationary density by using the FFT algorithm (and even the exact result (49) when $\alpha = 1$). Hence the following results comparing the estimated densities obtained by histogram interpolation to the FFT-approximated (or exact) densities.

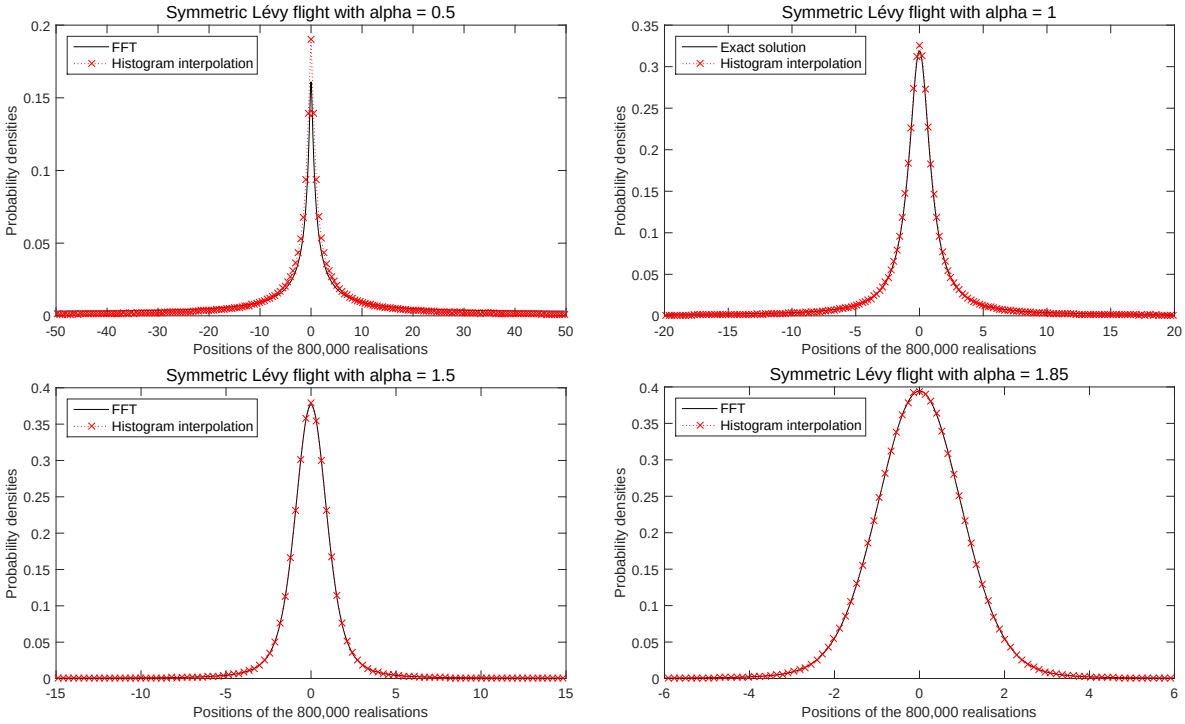

Figure 11 – Stationary density functions of different symmetric Lévy flights in potential V_1

In this case, the densities are very well estimated through numerical simulation. Indeed, as we were able perform good quality simulations of symmetric α-stable random variables in part 2 and as the potential V_2 does not change the type of the Lévy flight (as seen in part 3) this could be expected.

Quartic potential $\mathbf{V_2(x) = x^4/4}$ For this potential we used simulations of 800,000 particles as it was needed in order to get a precise enough density estimation in the Brownian motion case.

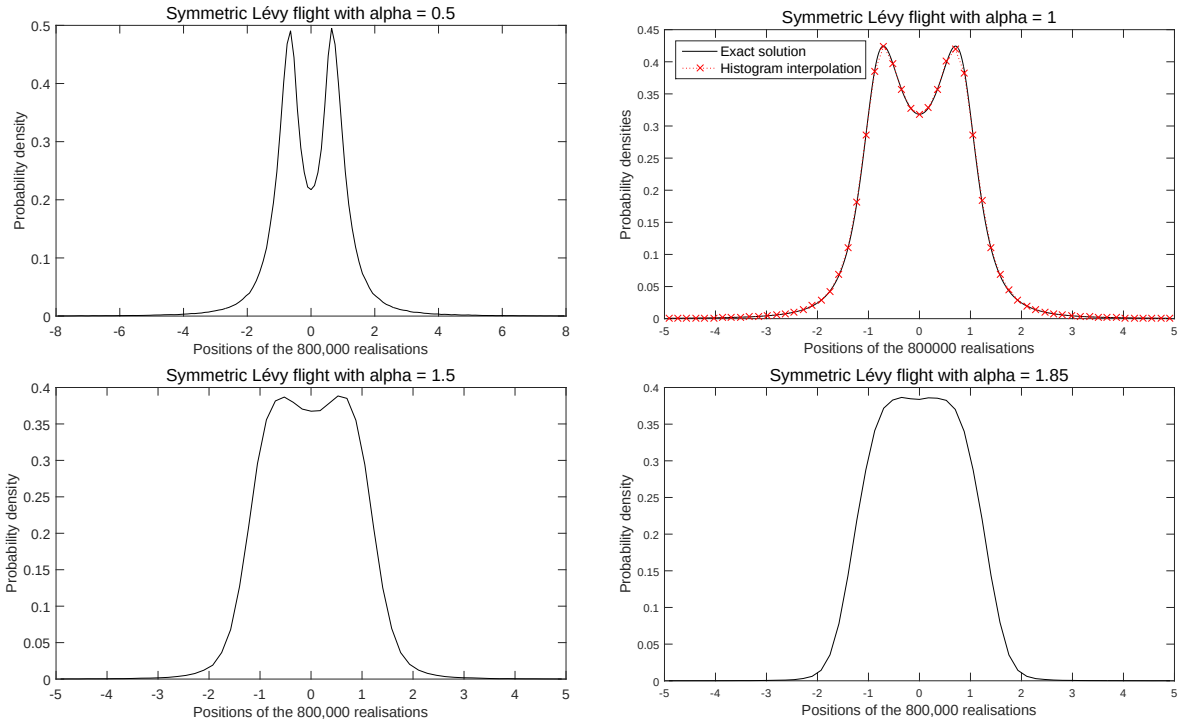

Figure 12 – Stationary density functions of different symmetric Lévy flights in potential V_2

Here, as the result obtained for $\alpha = 1$ is quite correct compared to the exact solution, we can assume that the results for other values of α are also quite precise. We can clearly see the densities going from having two clear, distinct modes to a single, very large mode as α increases.

Double-well potential $\mathbf{V_3(x) = x^4/4 - x^2/2}$ For this potential we do not have any theoretical (or FFT-approximated) result for the stationary density to compare with the interpolations' results.

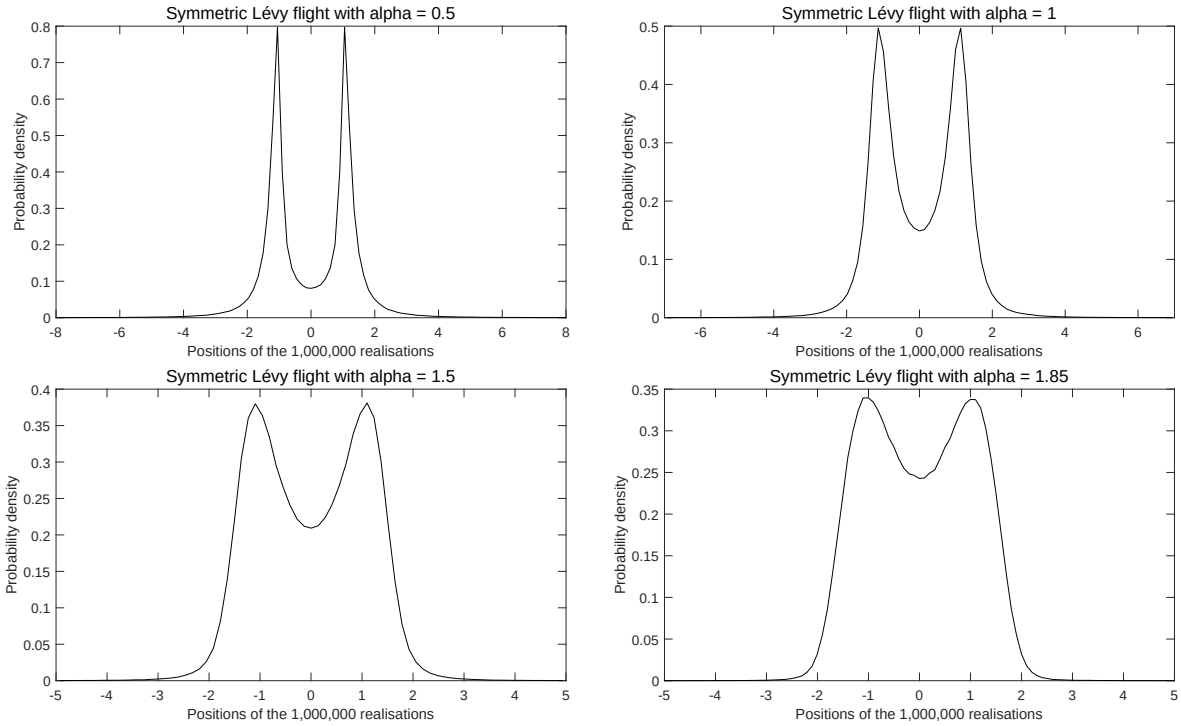

Figure 13 – Stationary density functions of different symmetric Lévy flights in potential V_3

In this case the densities remain bimodal for every α, but the bimodality becomes less pronounced as α increases.

4.2.3 Equations driven by asymmetric Lévy motions

Here we will run the simulations using skewed Lévy motions, for which solutions of the stationary fractional Fokker-Planck equation are not known. We will thus only present the interpolations of the histograms generated by the numerical solutions of the stochastic differential equations. The choices of the parameters of the simulations will be influenced by the ones that were found to be optimal in cases where we had exact or approximated results to compare to our simulations' results.

Considering the study of the simulation of α-stable random variables, the parameter α will be set to 0.85 in this part, and the parameter β will vary between 0 and 1. Indeed, we saw that our method of simulation performed less well for skewed α-stable Lévy motions with $\alpha > 1$, and that the simulation lost precision as α became close to 0 – hence our choice of a value of α that is close to but less than 1.

Quadratic potential $\mathbf{V_1(x) = x^2/2}$

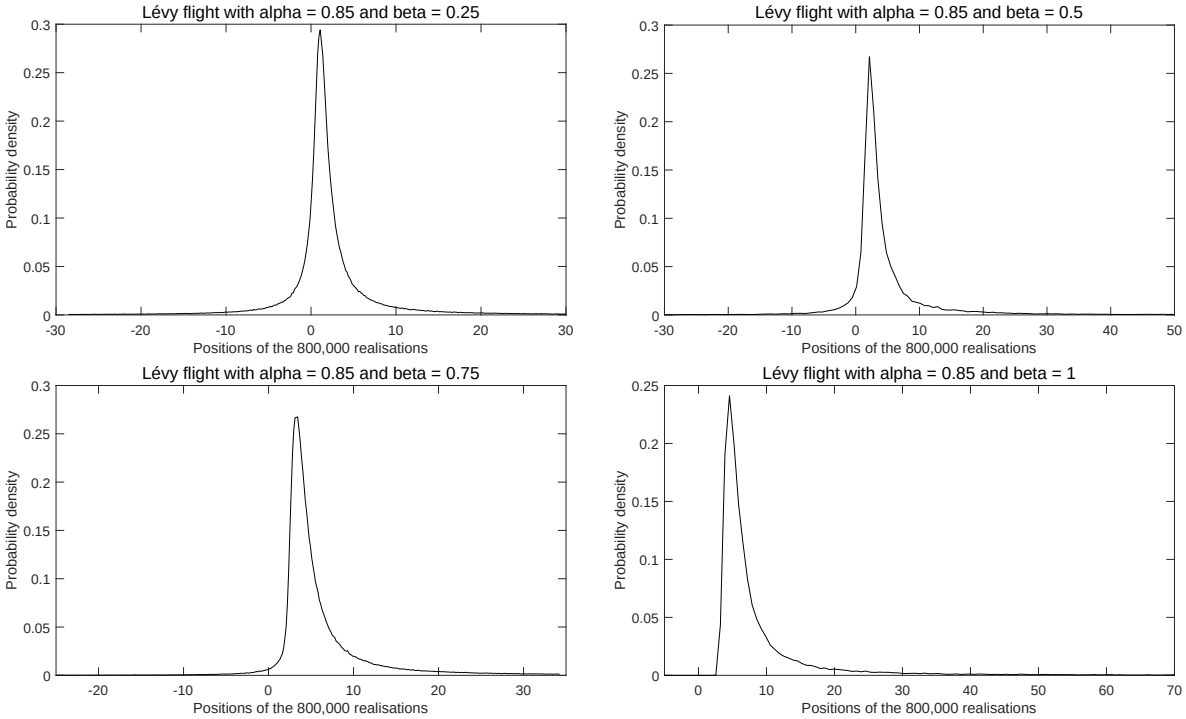

Figure 14 – Stationary density functions of different Lévy flights in potential V_1

Here we see that naturally, as β increases, the stationary density becomes more skewed. This generalises what we observed for symmetrical Lévy flights in this potential, i.e. that this external potential does not significantly change the nature of the Lévy flights driving the equation. Moreover, we can see that the skewness of the noise shifts the mode (which is usually the minimum of the potential) to the right (or the left, had we used negative values of β).

Quartic potential $\mathbf{V_2(x) = x^4/4}$

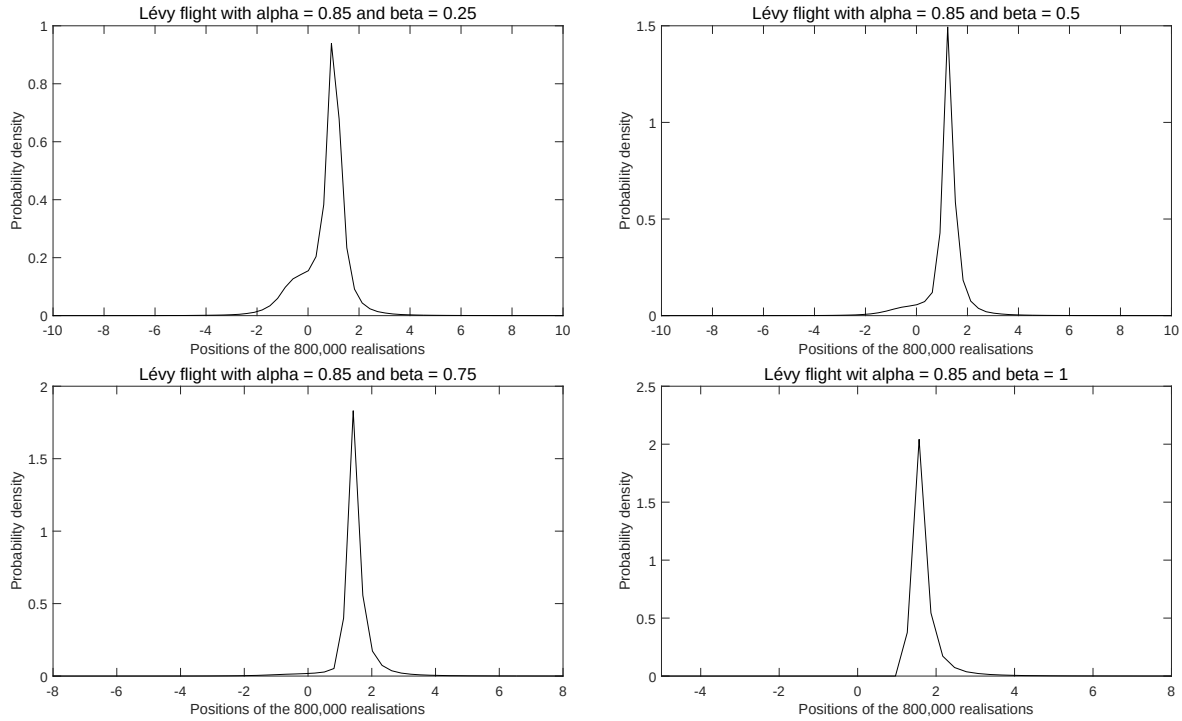

Figure 15 – Stationary density functions of different Lévy flights in potential V_2

For this potential, we had clear bimodal stationary desities for $\alpha \leq 1$, and we see that the skewness of the Lévy flights gradually erases the left mode, and for $\beta = 1$, the stationary density is even totally skewed to the right. It's also intersting to note that the right mode is shifted from about 1 to about 1.5 as β goes to 1.

Double-well potential $\mathbf{V_3(x) = x^4/4 - x^2/2}$

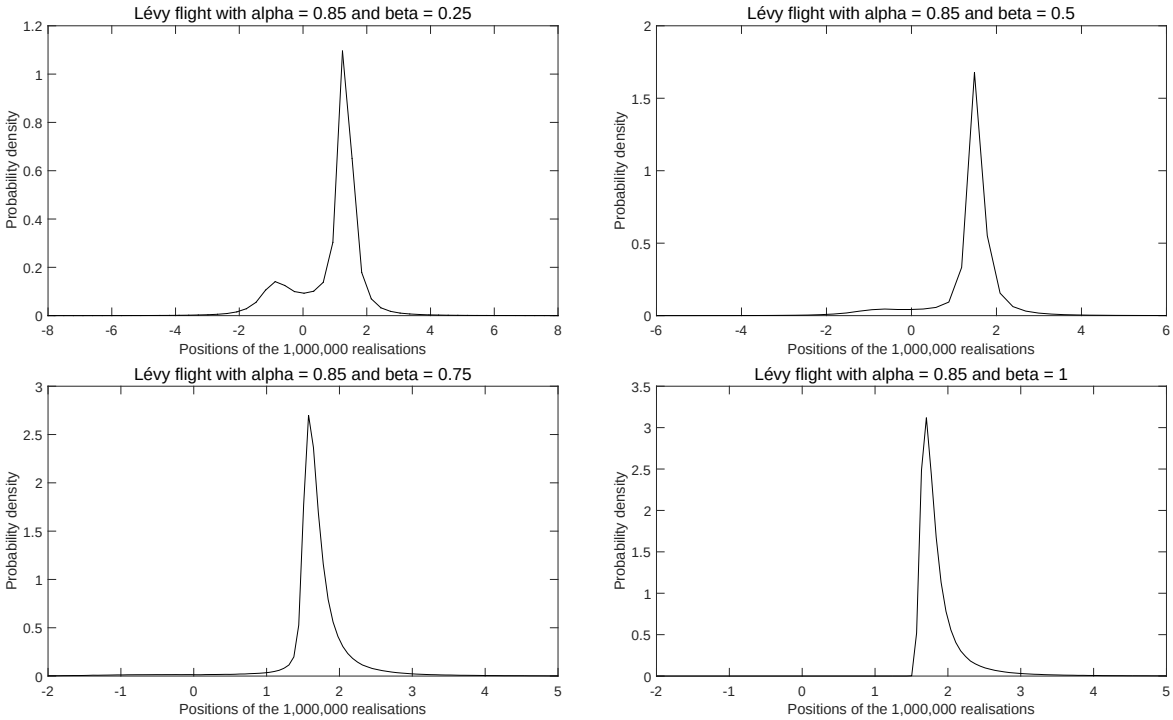

Figure 16 – Stationary density functions of different Lévy flights in potential V_3

The results are quite similar to the previous case, except that the left mode is erased more slowly than before. This was expectable as the bimodality came from the shape of the potential as well as the value of α. Also, the right mode is shifted from its original value of 1 to almost 2.

5 Conclusion

The first step of this project was to study an important class of random variables called α-stable. Such random variables are defined via their characteristic functions and thus their density functions are not always available in closed-form or analytic expressions. However, they can be easily simulated from uniformly distributed random variables by using a transform method similar to the Box-Muller one. Then the Lévy flights, or standard α-stable Lévy motions, were studied. We saw that they constituted a generalisation of the standard Brownian motion, as the α-stable random variables constituted a generalisation of Gaussian random variables. By realising a few simulations of such stochastic processes, we saw why the adjectives flights, or jumps, were often used to describe α-stable Lévy motions, whereas the adjective walk is used to describe Brownian motion. Then, we studied the Fokker-Planck equation for stochastic differential equations involving conservative forces (which are the negative derivative of a potential) and additive α-stable noise. We first derived the Fokker-Planck equation for both Brownian motion case and Lévy motion case, then we derived some solutions of the stationary Fokker-Planck equation. Finally, a method was conceived and used to estimate the solution of the stationary Fokker-Planck equation for any case by numerically solving the corresponding stochastic differential equation a large number of times and estimating the density by doing a linear interpolation of a well-chosen histogram of the steady-state solutions.

During the first part of the study, the transform method which simulates α-stable random variables was tested for different values of α and β. To test the quality of the simulation we compared a linear interpolation of an histogram of the result of the transform with values of the wanted density approximated by the FFT algorithm. Although we saw that the transform method works very well in most cases, there are certain cases (in particular when α is very close to zero, of when $\alpha > 1$ and $\beta \neq 0$) where the simulation is not very precise.

Then, during our study of the Fokker-Planck equation with an external potential, we saw that for the Brownian motion case, the stationary Fokker-Planck was easily solvable with a closed-form solution. However, for the Lévy motions cases, it was only possible to get analytic solutions in certain cases. But, in some cases with no analytic solution, it was still possible to perform an approximation of the solution using the FFT algorithm, as the stationary Fokker-Planck equation was easily solvable in Fourier space.

Finally, we first used our density estimation method in cases where we could compare our results to exact solutions of solutions approximated by FFT and we saw there was a good agreement between the two types of results. For the first quadratic potential, we expected in the symmetric case that the solution of the stationary Fokker-Planck equation was very similar to the distribution of the Lévy flight. We saw that this was also true for asymmetric Lévy flights. When using the quartic monostable potential and symmetric Lévy flights the solutions were strongly bimodal when alpha was small, and that this characteristic of the solution was less and less pronounced as alpha went to 2, and actually tended to a very flat distribution on the interval [-0.5, 0.5]. Then, by using asymmetric Lévy flights, we saw that the mode on the non-preferred side was progressively erased as the skewness parameter increased. When we used the quartic bistable potential (or double-well potential), we saw that the solutions in symmetric cases are always bimodal, with this characteristic being most emphasised as alpha was close to zero. Again, increasing the skewness parameter gradually erased one of the modes, but slower than with the quartic monostable potential. Also, in any case, we noted that as beta went to 1, the principal mode was shifted to the right.

A Matlab Scripts

A.1 Tests of the simulation of α-stable random variables

```
% Comparison of the distribution of a simulated alpha-stable Levy motion with ...
    the FFT of the characteristic fuction (approximating the pdf)

% Choice of parameters
alpha = 0.5;
Beta = 0.8;
sigma = 1;
M = 1000000; % Number of realisations
% For simplicity we only consider mu=0 since it only performs a translation

% Generation of data
X = sigma*alpha_stable(alpha, Beta, M);

% Generation of reference values of the density from the characteristic
% function
N = 4000; % Number of points
K = 200; % Range of spectral space
dk = 2*K/N;
k = [0:N-1]*dk-K; % Discretisation of spectral space
L = N*pi/2/K; % Range of space
dx = pi/K;
x = [0:N-1]*dx-L; % Discretisation of space

pk = exp(-(abs(sigma*k).^(alpha)).*(1-Beta*1i*sign(k)*tan(alpha*pi/2))); % ...
    Characteristic function values
fx = real(fftshift(fft(ifftshift(pk))))/L/2; % FFT of the characteristic function


% Creation of an histogram to estimate the density on the range [-L, L]
edges = linspace(-L, L, 100);
h = histogram(X, edges);
h.Normalization = 'pdf';

% Sample points for interpolation: middles of bins
nb = h.NumBins;
xs = zeros(1,nb);
for n=1:nb
    xs(n) = (h.BinEdges(n) + h.BinEdges(n+1))/2;
end

% Plot the results
vs = h.Values;
plot(x, fx, 'k', xs, vs,'r:x') % 'plot' linearly interpols the data by default
% Formatting the plot
xlabel('x','fontsize',12,'fontname','Arial');
ylabel('f(x)','fontsize',12,'fontname','Arial');
title(['Density estimations for alpha = ', num2str(alpha) and ', beta = ' ...
    num2str(beta)'], 'fontsize', 14, 'fontname', 'Arial')
lg = legend('FFT','Histogram interpolation');
set(lg,'fontsize',12,'fontname','Arial','Location','NorthWest');
set(gca,'fontsize',12,'fontname','Arial');
```

A.2 Simulation of symmetric standard α-stable Lévy motions

```
% Symmetric Levy flights simulations

% Choice of parameters
alpha = [1.25 1.5  1.75 2];
dt = 0.01; % time step
M = 500; % number of steps
T = 0:dt:(M*dt); % time line

% Initialise data vector
X = zeros(length(alpha), M+1);

% Compute increments vector for each alpha value
Y = zeros(length(alpha), M);
for k=1:length(alpha)
    Y(k, :) = ((dt)^(1/alpha)).*alpha_stable_sym(alpha(k), M);
end

% Compute the flights' trajectories
for n=1:M
    X(:, n+1) = X(:, n) + Y(:, n);
end

% Plot results
h = plot(T, X, 'LineWidth', 1);
set(h(1),'Color',[0 0 1]);
set(h(2),'Color',[0.3 0 0.7]);
set(h(3),'Color',[0.7 0 0.3]);
set(h(4),'Color',[1 0 0]);
xlabel('t','fontsize',12,'fontname','times roman');
title('Trajectories of Levy flights for different values of alpha', 'fontsize', ...
    14, 'fontname', 'Arial')
lg = legend('alpha = 1.25','alpha = 1.5', 'alpha = 1.75', 'Brownian motion');
set(lg,'fontsize',12,'fontname','Arial');
set(gca,'fontsize',12,'fontname','Arial');
```

A.3 Simulation of asymmetric standard α-stable Lév

```
% Levy flights simulations for alpha = 1

% Choice of parameters
alpha = 0.8;
Beta = [-1 -0.5 0.5 1];
dt = 0.01; % time step
M = 500; % number of steps
T = 0:dt:(M*dt); % time line

% Initialise data vector
X = zeros(length(Beta), M+1);

% Compute increments vector for each alpha value
Y = zeros(length(Beta), M);
for k=1:length(Beta)
    Y(k, :) = ((dt)^(alpha)).*alpha_stable(alpha, Beta(k), M);
end

% Compute the flights' trajectories
for n=1:M
    X(:, n+1) = X(:, n) + Y(:, n);
end

% Plot results
h = plot(T, X, 'LineWidth', 1);
set(h(1),'Color',[0 0 1]);
set(h(2),'Color',[0.3 0 0.7]);
set(h(3),'Color',[0.7 0 0.3]);
set(h(4),'Color',[1 0 0]);
xlabel('t','fontsize',12,'fontname','Arial');
title('Trajectories of Levy flights for alpha = 0.8 and different values of ...
    beta', 'fontsize', 14, 'fontname', 'Arial')
lg = legend('beta = -1','beta = -0.5', 'beta = 0.5', 'beta = 1');
set(lg,'fontsize',12,'fontname','Arial');
set(gca,'fontsize',12,'fontname','Arial');
```

A.4 Numerical solution of the stationary Fokker-Planck equation using the forward Euler-Maruyama scheme

```
function P_inf = fastFPapproxLM(gradV, N, dt, M, alpha, Beta)

% gradV must be a function handle
% N is the number of realisations used
% dt is the time step
% M is number of steps wanted
% alpha is the stability index of the Levy noise driving the process
% Beta is the skewness parameter

% Computes a discrete approximation of the stationary probability density
% P_inf of the process dX = -gradV(X)*dt + dS_alpha, as well as the values of the
% space points, using the forward Euler-Maruyama scheme and initial normal
% distribution over [0,1]
```

```
% initial condition : N particules uniformly distributed on [0, 1]
X = rand(N,1);

% compute X after M time steps
for k=1:M
    X = X - gradV(X)*dt + ((dt)^(1/alpha))*alpha_stable(alpha, Beta, N);
end

% Sample the data that is in the most interesting range
X = X(logical(abs(1 - isnan(X)))); % Cleaning the data
m = median(X);
% Choice of range of the histogram
Min = m - 30;
Max = m + 30;

% Creation of a pdf histogram for final state of X
edges = linspace(Min, Max, 200); % edges of the histogram
h = histogram(X, edges);
h.Normalization = 'pdf';

% Sample points for interpolation: middles of bins
K = h.NumBins;
xs = zeros(1,K);
for k=1:K
    xs(k) = (h.BinEdges(k) + h.BinEdges(k+1))/2;
end

% Sample values
vs = h.Values;

% P_inf contains the sample points and the corresponding values of the
% approximated density
P_inf = [xs ; vs];
```

A.5 Numerical solution of the stationary Fokker-Planck equation using the split-step backward Euler scheme for potential V_2

```
function P_inf = FPapproxLMV2(N, dt, M, alpha, beta)

% N is the number of realisations used
% dt is the time step
% M is number of steps wanted
% alpha is the stability index of the Levy noise driving the process
% beta is the skewness parameter

% Computes a discrete approximation of the stationary probability density
% P_inf of the process dX = -(X^3)*dt + dS_alpha, as well as the values of the
% space points, using the Split-step backward Euler-Maruyama scheme and initial ...
    normal
% distribution over [0,1]


% initial condition : N particules uniformly distributed on [0, 1]
X = rand(N,1);


% compute X after M time steps
```

```
for k=1:M
    % For the implicit part, use the arithmetic formula in the particular
    % case where gradV(x) = x^3
    Gamma = real((((3*dt)^(2)).*X + sqrt(3).*sqrt((27*(dt)^(4)).*(X.^2) + ...
        4*(dt)^3)).^(1/3));
    Xstar = Gamma./(((2)^(1/3))*((3)^(2/3))*dt) - ((2/3)^(1/3))./Gamma;

    X = Xstar + ((dt)^(1/alpha))*alpha_stable(alpha, beta, N);
end

% Cleaning the data
X = X(logical(abs(1 - isnan(X))));
% Choice of range of the histogram
m = median(X)
Min = m - 30
Max = m + 30

% Creation of a pdf histogram for final state of X
edges = linspace(Min, Max, 250); % edges of the histogram
h = histogram(X, edges);
h.Normalization = 'pdf';

% Sample points for interpolation: middles of bins
K = h.NumBins;
xs = zeros(1,K);
for k=1:K
    xs(k) = (h.BinEdges(k) + h.BinEdges(k+1))/2;
end

% Sample values
vs = h.Values;

% P_inf contains the sample points and the corresponding values of the
% approximated density
P_inf = [xs ; vs];
```

A.6 Comparison of the numerical solution of the stationary Fokker-Planck equation with the exact solution

```
% Using the function FPapproxBM to plot a comparison between the
% approximated density and the exact theoretical result

% Choice of potential and parameters
V = @(x) 0.25*x.^4 - 0.5*x.^2; % Potential and its gradient
gradV = @(x) x.^3 - x;
N = 1000000; % Number of realisations
dt = .001; % Time step
M = 10000; % Number of steps
% Estimated values of p_inf
A = FPapproxBMV3(N, dt, M);

Width = A(1,2) - A(1,1); % distance between points in A
K = length(A); % Number of points in A

% Discretization of space and query points
dx = Width*0.01;
xq = A(1,1):dx:A(1,K); % finer abscisse axis
```

```
% Values of the theoretical distribution
Y = exp(-V(xq)*2);
% Normalization
Y = Y./(sum(Y*dx));

% Plotting both interpolated (by 'plot') pdf and theoretical pdf
plot(xq, Y, 'k', A(1,:), A(2, :),'r:x')
% Formatting the plot
xlabel(['Positions of the ', num2str(N),' realisations'], 'fontsize', 12, ...
    'fontname', 'Arial')
ylabel('Probability densities','fontsize',12,'fontname','Arial');
title('Brownian motion in potential V\_3(x) = x\^4/4 - x\^2/2', 'fontsize', 14, ...
    'fontname', 'Arial')
lg = legend('Exact solution','Histogram interpolation');
set(lg,'fontsize',12,'fontname','Arial','Location','NorthWest');
```

A.7 Comparison the numerical solution of the stationary Fokker-Planck equation with an FFT approximation of the exact result

```
% Using the function FPapproxLM when beta=0 to plot a comparison between the
% approximated density and the FFT of the theoretical result in Fourier
% space

% Choice of potential and parameters
%gradV = @(x) x;
N = 50000;
dt = .001;
M = 10000;
alpha = 1.85;
% Discrete (coarse) set of values of p_inf
A = fastFPapproxLM(gradV, N, dt, M, alpha, 0);

Width = A(1,2) - A(1,1); % distance between points in A
Nbins = length(A); % Number of points in A

% Discretization of space and query points
dx = Width*0.001;
xq = A(1,1):dx:A(1,Nbins); % finer abscisse axis

% Values of the theoretic distribution in Fourier space
% Definition of Fourier space
L = max(abs(A(1,1)), abs(A(1,Nbins))); % As beta=0 we can choose a space range ...
    centered on 0 ([-L,L])
Nfft = 4000; % Number of points to be estimated by FFT
K = (Nfft*pi)/(2*L); % Fourier space range
dx = 2*L/Nfft;
dk = 2*K/Nfft;
x = [0:Nfft-1]*dx - L; % regular space axis
k = [0:Nfft-1]*dk - K; % Fourier space axis
% Density values in Fourier space
Yhat = exp((-abs(k).^(alpha))/alpha);
% FFT values
Y = real(ifftshift(fft(ifftshift(Yhat))))/(2*L);
% Normalization
Y = Y./(sum(Y*dx));
```

```
% Plotting both interpolated pdf and theoretical pdf
plot(x, Y, 'k', A(1,:), A(2, :),'r:x')
% Formatting the plot
xlabel(['Positions of the ', num2str(N),' realisations'], 'fontsize', 12, ...
    'fontname', 'Arial')
ylabel('Probability densities','fontsize',12,'fontname','Arial');
title(['Symmetric Levy flight with alpha = ', num2str(alpha)], 'fontsize', 14, ...
    'fontname', 'Arial')
lg = legend('FFT','Histogram interpolation');
set(lg,'fontsize',12,'fontname','Arial','Location','NorthWest');
```

A.8 Plots of the numerical solution of the stationary Fokker-Planck equation when no approximation or exact result is known

```
% Using the function FPapproxLM when beta is not null to plot approximated
% density of the steady-state solution

% Choice of potential and parameters
%gradV = @(x) x;
N = 800000;
dt = .001;
M = 10000;
alpha = 1.5;
beta = 0;
% Discrete (coarse) set of values of p_inf
A = FPapproxLMV2(N, dt, M, alpha, beta);

% Plotting the interpolated pdf
plot(A(1,:), A(2, :), 'k')
% Formatting the plot
xlabel(['Positions of the ', num2str(N),' realisations'], 'fontsize', 12, ...
    'fontname', 'Arial')
ylabel('f(x)','fontsize',12,'fontname','Arial');
title(['Levy flight with alpha = ', num2str(alpha), ' and beta = ', ...
    num2str(beta)], 'fontsize', 14, 'fontname', 'Arial')
%lg = legend('FFT','Histogram interpolation');
%set(lg,'fontsize',12,'fontname','Arial','Location','NorthWest');
```

References

[1] Dubkov A. and Spagnolo B. Langevin approach to lévy flights in fixed potentials: exact results for stationary probability distributions. *Acta Physica*, 38:1745–1758, 2007.

[2] Janicki A. and Weron A. *Simulation and chaotic behavior of [alpha]-stable stochastic processes.* Marcel Dekker, New York, 1994.

[3] Xuerong Mao Desmond J. Higham and Andrew M. Stuart. Strong convergence of euler-type methods for nonlinear stochastic differential equations. *SIAM Journal on Numerical Analysis*, 40:1041–1063, 2003.

[4] Charles Fox. The g and h functions as symmetrical fourier kernels. *Trans. Amer. Math. Soc.*, 98:395–429, 1961.

[5] R.K. Saxena Francesco Mainardi, Gianni Pagnini. Fox h functions in fractional diffusion. *Journal of Computational and Applied Mathematics*, 178:321 – 331, 2005.

[6] M. Larcheveque V.V. Yanovsky J. Duan, D. Schertzer and S. Lovejoy. Fractional fokker–planck equation for nonlinear stochastic differential equations driven by non-gaussian levy stable noises. *J. Math. Phys.*, 42:200–2012, 2001.

[7] Metzler R. Jespersen S. and Fogedby H.C. Lévy flights in external force fields: Langevin and fractional fokker–planck equations, and their solutions. *Phys. Rev.*, E59:2736–2745, 1999.

[8] J. P. Nolan. *Stable Distributions - Models for Heavy Tailed Data.* Birkhauser, Boston, 2015. In progress, Chapter 1 online at academic2.american.edu/∼jpnolan.

[9] Sheldon M. Ross. *A First Course in Probability.* Pearson Prentice Hall; 8th edition, New Jersey, 2008.

[10] Gennay Samorodnitsky and Murad S. Taqqu. *Stable Non-Gaussian Random processes: Stochastic Models with Infinite Variance.* Chapman and Hall, New York, 1994.