Relativistic Brownian Motion

Yue Wang, Prof. Rajeev

April 26, 2021

Abstract

The displacements of a particle from classical Brownian motion form a Gaussian distribution. However, the distribution of variables drawn from a distribution with infinite variance as opposed to a finite variance in classical Brownian motion is not Gaussian. We show that the sum of a large number of lognormal variables tends to an α-stable Levy subordinator when $0 < \alpha < 1$. This distribution is surprisingly common in real-life stochastic systems such as stock market and relativistic Brownian motion.

1 Introduction

1.1 Background

In classical Brownian Motion, defined below, the speed of particles has been assumed to be infinite and the displacement of each step has been treated as independent.[1]

Definition 1.1. (CLASSICAL BROWNIAN MOTION) *A real-valued stochastic process $\{B(t) : t \geq 0\}$ is called a classical Brownian motion with start in $x \in \mathbb{R}$ if the following holds:*

1. $B(0) = x$,
2. *the process has independent increments, i.e. for all times $0 \leq t_1 \leq t_2 \leq ... \leq t_n$ the increments $B(t_n) - B(t_{n-1}), B(t_{n-1}) - B(t_{n-2}), ..., B(t_2) - B(t_1)$ are independent random variables,*
3. *for all $t \geq 0$ and $h > 0$, the increments $B(t+h) - B(t)$ are normally distributed with expectation zero and variance h,*
4. *almost surely, the function $t \to B(t)$ is continuous.*

We say that $\{B(t) : t \geq 0\}$ is a classical Brownian motion.

1.2 Model Construction

The velocity of particles in classical Brownian motion are assumed to be infinite. However, in reality, the speed of particles is bounded above by the speed of light c. We are interested in studying the motion of the particles moving at nearly the speed of light. Therefore, this study requires relativity.

Velocity is the derivative of position with respect to time. In relativity, a more natural variable is rapidity θ, which is related to the derivative of position X^1 and time X^0 with respect to proper time τ (time measured by a clock attached to the particle). In units with $c = 1$,

$$\frac{dX^0}{d\tau} = \cosh\theta, \quad \frac{dX^1}{d\tau} = \sinh\theta.$$

We can see that the relation of velocity to rapidity is

$$v = \frac{dX^1}{dX^0} = \tanh\theta.$$

Although θ is not bounded, v is bounded by the speed of light.

It is more convenient to introduce null co-ordinates

$$X^0 \pm X^1 = X^\pm$$

so that

$$\frac{dX^+}{d\tau} = e^\theta, \quad \frac{dX^-}{d\tau} = e^{-\theta}.$$

Thus, we can now propose a model of relativistic Brownian motion in one spatial dimension.

A particle moves along a straight line until it collides with another particle. In each collision, the rapidity of a particle is changed by the addition of a random variable. The effect of a large number of such collisions is to make rapidity into a Gaussian random variable (Central Limit Theorem). The null coordinates are incremented by the exponential of rapidity at each step:

$$X_k^+ = X_{k-1}^+ + e^{\theta_k}\Delta\tau, \quad X_k^- = X_{k-1}^- + e^{-\theta_k}\Delta\tau$$

where $\Delta\tau$ is a small interval of proper time in between two collisions. Also, θ_k are independent identically distributed (i.i.d) Gaussian random variables.

Thus we have

$$X_n^+ = \Delta\tau \sum_{k=1}^{n} e^{\theta_k}$$

and similarly for X_n^-. The mathematical question we solve is how to take the limit $n \to \infty$ and $\Delta\tau \to 0$ such that X_n tends to a random process.

We should expect that this limit is an infinitely divisible distribution, because the sum can be subdivided into independent terms. Because the terms in the sum are positive, we should expect also that this will be a "subordinate process" in the sense defined below. A tricky question is how the mean and variance of the Gaussian random variables θ_k must behave as $n \to \infty$.

If θ is Gaussian, $x = e^\theta$ is log-normal. It is useful to write its probability density function in the following form:

Proposition 1.2. *The probability measure of a log-normal variable can be written as*

$$\mu(dx) = \frac{1}{Z} \frac{e^{-\frac{\log^2 x}{2\sigma^2}}}{x^{\alpha+1}} dx$$

where

$$E(\log x) = -\alpha\sigma^2, \quad \mathrm{Var}(\log x) = \sigma^2$$

and

$$Z = \sqrt{2\pi}\sigma e^{\frac{\alpha^2\sigma^2}{2}}.$$

Proof. Make the change of variables $\theta = \log x$

$$\mu(dx) = \frac{1}{Z} \frac{e^{-\frac{\theta^2}{2\sigma^2}}}{e^{(\alpha+1)\theta}} e^\theta d\theta$$

$$= \frac{1}{Z} e^{-\frac{1}{2\sigma^2}\theta^2 - \alpha\theta} d\theta$$

$$= \frac{e^{\frac{\alpha^2\sigma^2}{2}}}{Z} e^{-\frac{1}{2\sigma^2}\left[\theta+\alpha\sigma^2\right]^2} d\theta$$

$$= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}\left[\theta+\alpha\sigma^2\right]^2} d\theta.$$

This is the probability density of a normal variable with mean $-\alpha\sigma^2$ and variance σ^2. □

We will hold α fixed as $n \to \infty$. The question then is how the variance σ^2 must depend on n in order that the sum of n identical log-normal variables tends to a sensible limit as $n \to \infty$. We will show below that we need

$$\sigma_n^2 \sim \frac{\log\left[\frac{\alpha^2 n^2}{2\pi}\right]}{\alpha^2}.$$

Since the variance tends to infinity, the central limit theorem does not apply and the limiting distribution for X_n is not a Gaussian, unlike for Brownian motion. Instead, it will be an α-stable subordinator Levy distribution. This is our main result.

In the next section, we will review some useful results from the theory of stochastic processes.

1.3 Impact

A few studies were published on related problems. The distribution of the sum of log-normal variables was studied as early as in 1960.[2] A comparison of four methods to approach the sum of log-normal variables was published in 1995.[3] In 2003, a paper focusing on using moment-matching approximation to study the distribution of the sum of log-normal variables was published.[4]

The distribution of log-normal variables with additional parameters from financial models was studied in 1998 and the distribution of independent but not necessarily identically distributed log-normal variables was studied in 2005.[5][6] In 2008, a paper was published on the asymptotics of the sum of log-normal variables.[7]

In this paper, we will show that the sum of log-normal variables tends to an infinitely divisible α-stable Levy subordinator when the number of terms of in the sum goes to infinity. α is the ratio of the mean to the variance of the normal distribution and we will focus on the case when $0 < \alpha < 1$. However, this approach is very slow and the number of steps is expected to be much larger than 500.

The resulting distribution of relativistic Brownian motion is quite common in real-life stochastic systems including stock market. In the future, this work can be generalized into financial field.

We will first present our theoretical derivation, with special emphasize on the major approximations and normalizations. In the following section, we will present a numerical verification of the proposed distribution. Finally, our methods will be demonstrated using a simulation program.

2 Theoretical Derivation

First, we state the definition of a Levy process.[8]

Definition 2.1. (Levy Process) *Let $X = (X(t), t \geq 0)$ be a stochastic process defined on a probability space $(\Omega, \mathcal{F}, P)$. We say that it has independent increments if for each $n \in \mathbb{N}$ and each $0 \leq t_1 < t_2 < ... < t_{n+1} < \infty$ the random variables $(X(t_{j+1}) - X(t_j), 1 \leq j \leq n)$ are independent and that it has stationary increments if each $X(t_{j+1}) - X(t_j) \stackrel{d}{=} X(t_{j+1} - t_j) - X(0)$.*

We say that X is a Levy process if:

1. *$X(0) = 0$ (almost surely);*
2. *X has independent and stationary increments;*
3. *X is stochastically continuous, i.e. for all $a > 0$ and for all $s \geq 0$*

$$\lim_{t \to s^+} P(|X(t) - X(s)| > a) = 0.$$

The third property of Levy process implies that Levy processes are not necessarily continuous. In fact, classical Brownian motion is the only Levy process that is continuous. Our model for relativistic Brownian motion is not continuous. However, it is infinitely divisible.[8]

Definition 2.2. (INFINITELY DIVISIBLE) *Let X be a random variable taking values in $\mathbb{R}^d$ with law μ_X. We say that X is infinitely divisible if, for all $n \in \mathbb{N}$, there exist independent and identically distributed (i.i.d.) random variables $Y_1^{(n)}, ..., Y_n^{(n)}$ such that*

$$X \overset{d}{=} Y_1^{(n)} + ... + Y_n^{(n)}.$$

2.1 Problem Reduction

The following theorem guarantees the infinite divisibility of relativistic Brownian motion once the motion is shown to be a Levy process.[8]

Theorem 2.3. *If X is a Levy process, then $X(t)$ is infinitely divisible for each $t \geq 0$.*

Proof. For each $n \in \mathbb{N}$, define

$$Y_k^{(n)}(t) = X\Big(\frac{kt}{n}\Big) - X\Big(\frac{(k-1)t}{n}\Big).$$

$Y_k^{(n)}(t)$ are i.i.d. for each $1 \leq k \leq n$ by the second property from the definition for Levy process.

Therefore, for such n, we have

$$X(t) = Y_1^{(n)}(t) + ... + Y_n^{(n)}(t).$$

By definition, $X(t)$ is infinitely divisible.

□

Therefore, we have reduced the problem to showing the sum of log-normal variables tends to an α-stable subordinator Levy distribution.

We now introduce five definitions.[8][9]

Definition 2.4. (STABLE RANDOM VARIABLES) *We consider the general central limit problem in dimension $d = 1$, so let $(Y_n, n \in \mathbb{N})$ be a sequence of real-valued i.i.d. random variables and construct the sequence $(S_n, n \in \mathbb{N})$ of rescaled partial sums*

$$S_n = \frac{Y_1 + Y_2 + ... + Y_n - b_n}{\sigma_n},$$

where $(b_n, n \in \mathbb{N})$ is an arbitrary sequence of real numbers and $(\sigma_n, n \in \mathbb{N})$ an arbitrary sequence of positive numbers. We are interested in the case where there exists a random variable X for which

$$\lim_{n\to\infty} P(S_n \leq x) = P(X \leq x) \tag{1}$$

for all $x \in \mathbb{R}$, i.e. $(S_n, n \in \mathbb{N})$ converges in distribution to X.

A random variable is said to be stable if it arises as a limit as in Equation 1.

Definition 2.5. (Stable Levy Processes) *A stable Levy process is a Levy process X in which each $X(t)$ is a stable random variable*

Definition 2.6. (α-stable Levy Processes) *Let $\{X_t : t \geq 0\}$ be a Levy process on $\mathbb{R}^d$. It is called α-stable if, for any $a > 0$, there is α such that*

$$\{X_{at} : t \geq 0\} \overset{d}{=} \{a^{\frac{1}{\alpha}} X_t : t \geq 0\}.$$

Definition 2.7. (Subordinator) *A subordinator is a one-dimensional Levy process that is non-decreasing (almost surely).*

More specifically, α-stable subordinators are defined as the following.

Definition 2.8. (α-stable Subordinators) *An α-stable subordinator X is a Levy process with $\mathbb{E}\left(e^{-uX}\right) = e^{-u^{\alpha}}$ for $0 < \alpha < 1, u \geq 0$.*

Using the identity (the proof of which we will recall in section 2.7),

$$u^{\alpha} = \frac{\alpha}{\Gamma(1-\alpha)} \int_0^{\infty} \left(1 - e^{-ux}\right) \frac{dx}{x^{1+\alpha}}.$$

It follows that the problem has been reduced to showing

$$\log \mathbb{E}\left(e^{-uX}\right) = \int_0^{\infty} \left(1 - e^{-ux}\right) \frac{dx}{x^{1+\alpha}}. \tag{2}$$

2.2 Definition of Variables

We are interested in the expected value for the exponential of displacement. In the following derivation, we will use the following notation:

n: the number of steps;
x: step, the log-normal variable;
X_n: displacement after n steps;
$X = \lim_{n\to\infty} X_n$;
s: time window;
σ^2: the variance of rapidity;
$-\alpha\sigma^2$: the mean of rapidity.
$\mu_n(dx)$: the probability measure.

Since every step is independent

$$\mathbb{E}(e^{-sX_n}) = [\int_0^{\infty} e^{-sx} \mu_n(dx)]^n.$$

$\int_0^{\infty} e^{-sx}\mu_n(dx)$ is the expected value for the exponential of each step x. The summation of the expected value of each step becomes multiplication when taking the exponential.

If we choose $\int_0^{\infty} n\mu_n(dx) = 1$,

$$\mathbb{E}(e^{-sX_n}) = [1 + \frac{1}{n}\int_0^{\infty} (e^{-sx} - 1) n\mu_n(dx)]^n$$

$$\mathbb{E}(e^{-sX_n}) = \exp[n\log(1+\frac{1}{n}\int_0^\infty (e^{-sx}-1)n\mu_n(dx))]. \tag{3}$$

This equation allows us to further simplify the expression for $\mathbb{E}(e^{-sX_n})$ using approximations, which will be explained in the following sections.

2.3 Large Number Approximation

Assuming n is large so that $\frac{1}{n}$ is small.

Consider the Taylor expansion around $x = 0$,

$$\log(1+x) = x - \frac{x^2}{2} + ...$$

When x is sufficiently small, the first two or one terms is a good approximation for $\log(1+x)$. We now apply this approximation to Equation 3.

The Taylor expansion for Equation 3 is

$$\exp[\int_0^\infty (e^{-sx}-1)n\mu_n(dx) - \frac{1}{2n}[\int_0^\infty (e^{-sx}-1)n\mu_n(dx)]^2] + ...$$

Since $\frac{1}{n}$ is sufficiently small, only keeping first two terms in its Taylor expansion gives a good approximation for $\mathbb{E}(e^{-sX_n})$, as shown in Equation 4.

$$\mathbb{E}(e^{-sX_n}) = \exp[\int_0^\infty (e^{-sx}-1)n\mu_n(dx) - \frac{1}{2n}[\int_0^\infty (e^{-sx}-1)n\mu_n(dx)]^2]. \tag{4}$$

Moreover, if we only keep the first term,

$$\mathbb{E}(e^{-sX_n}) = \exp[\int_0^\infty (e^{-sx}-1)n\mu_n(dx)]. \tag{5}$$

2.4 Normalization

Before deriving Equation 5 further, it is critical to study its properties in limit cases. If μ_n does not depend on n, the integral in Equation 5 goes to infinity as n goes to infinity, which is not physical. To eliminate the n dependence in Equation 5, we choose μ_n such that $n\mu_n$ does not depend on n. Physically, the change of n means a change of the number of steps in a fixed time window. By dividing the given time window into finer bins, we do not expect to receive larger and larger total displacement happening within the time window.

Recall the log-normal probability density function. This measure is used since the variable x follows a log-normal distribution.

$$\mu_n(dx) = \frac{1}{Z_{\sigma,\alpha}} \frac{\exp(-\frac{\log^2 x}{2\sigma^2})}{x^{\alpha+1}} dx$$

where

$$Z_{\sigma,\alpha} = \sqrt{2\pi}\sigma e^{\frac{\alpha^2\sigma^2}{2}}$$

We choose σ_n so that

$$Z_{\sigma_n,\alpha} = n.$$

Then

$$n\mu_n(dx) = \frac{n}{Z_{\sigma_n,\alpha}} \frac{\exp(-\frac{\log^2 x}{2\sigma_n^2})}{x^{\alpha+1}} dx = \frac{\exp(-\frac{\log^2 x}{2\sigma_n^2})}{x^{\alpha+1}} dx \tag{6}$$

which is independent of n. This normalizes $\mathbb{E}(e^{-sX_n})$.

By solving $Z_{\sigma_n,\alpha} = n$, we have

$$\sigma_n^2 = \frac{W(\frac{\alpha^2 n^2}{2\pi})}{\alpha^2}. \tag{7}$$

where $W(z)$ is the Lambert function, defined as the principal branch solution of $z = W(z)e^{W(z)}$.

2.5 Very Large Number Approximation

We simplify Equation 7 by assuming n is even larger so that $\frac{1}{\log n}$ is small. Expanding the Lambert function introduced above,

$$W(z) = \log z - \log\log z + O(1)$$

Since $\frac{1}{\log n}$ is small, we can approximate $W(z)$ only keep the first term in the expansion. Substituting $W(\frac{\alpha^2 n^2}{2\pi})$ by the first term from its expansion in Equation 7,

$$\sigma_n^2 = \frac{\log(\frac{\alpha^2 n^2}{2\pi})}{\alpha^2}.$$

Applying this result and the expression for $n\mu_n$ in Equation 6 to Equation 5, we have

$$\mathbb{E}(e^{-sX}) = \exp[\int_0^\infty (e^{-sx} - 1) \frac{\exp(-\frac{\log^2 x}{2\sigma_n^2})}{x^{\alpha+1}} dx]. \tag{8}$$

2.6 Another Large Number Approximation

Equation 8 can be further simplified assuming $n \to \infty$, then $\frac{\log^2 x}{2\sigma_n^2}$ is small. Therefore, we can approximate $\exp(-\frac{\log^2 x}{2\sigma_n^2})$ by 1. With this approximation, Equation 8 becomes

$$\mathbb{E}(e^{-sX}) = \exp[\int_0^\infty (e^{-sx} - 1) \frac{1}{x^{\alpha+1}} dx]. \tag{9}$$

Thus,

$$\log \mathbb{E}(e^{-sX}) = \int_0^\infty (e^{-sx} - 1) \frac{1}{x^{\alpha+1}} dx.$$

This agrees with Equation 2, so the sum of log-normal variables in relativistic Brownian motion is an infinitely divisible α-stable Levy subordinator as long as the integral $\int_0^\infty [1 - e^{-sx}] \frac{dx}{x^{\alpha+1}}$ converges. Notice that the integral on the right hand side converges when $0 < \alpha < 1$.

2.7 Alternative Large Number Approximation: First Order

Instead of applying the approximation in the previous section, which is $\exp(-\frac{\log^2 x}{2\sigma_n^2}) \to 1$, consider the following Taylor expansion.

$$\int_0^\infty (e^{-sx}-1)\frac{\exp(-\frac{\log^2 x}{2\sigma_n^2})}{x^{\alpha+1}}dx = \int_0^\infty (e^{-sx}-1)\frac{1}{x^{\alpha+1}}dx - \frac{1}{2\sigma_n^2}\int_0^\infty \frac{\log^2 x}{x^{\alpha+1}}[e^{-sx}-1]dx + ...$$

Furthermore, following the derivation by Applebaum, we can prove the identity mentioned in section 2.1.[8] For $0 < \alpha < 1$,

$$\begin{aligned}\int_0^\infty (e^{-sx}-1)\frac{1}{x^{\alpha+1}}dx &= \int_0^\infty (\int_0^x se^{-sy}dy)\frac{dx}{x^{\alpha+1}} \\ &= \int_0^\infty (\int_y^\infty \frac{dx}{x^{\alpha+1}})se^{-sy}dy \\ &= -\frac{s}{\alpha}\int_0^\infty e^{-sy}y^{-\alpha}dy \\ &= -\frac{s^\alpha}{\alpha}\int_0^\infty e^{-x}x^{-\alpha}dx \\ &= -\frac{s^\alpha}{\alpha}\Gamma(1-\alpha) \\ &= -\frac{s^\alpha}{\alpha}(-\alpha)\Gamma(-\alpha) \\ &= s^\alpha\Gamma(-\alpha)\end{aligned}$$

Therefore,

$$\log \mathbb{E}(e^{-sX}) = s^\alpha\Gamma(-\alpha).$$

Moreover, since

$$-\frac{s^\alpha}{\alpha}\Gamma(1-\alpha) = s^\alpha\Gamma(-\alpha),$$

$$s^\alpha\Gamma(-\alpha) = \int_0^\infty [e^{-sx}-1]\frac{dx}{x^{\alpha+1}}.$$

Hence,

$$\int_0^\infty (e^{-sx}-1)\frac{\exp(-\frac{\log^2 x}{2\sigma_n^2})}{x^{\alpha+1}}dx \to \Gamma(-\alpha)s^\alpha - \frac{1}{2\sigma_n^2}\int_0^\infty \frac{\log^2 x}{x^{\alpha+1}}[e^{-sx}-1]dx + ...$$

If we only keep the first order in this expansion

$$\mathbb{E}(e^{-sX}) = \exp[\Gamma(-\alpha)s^\alpha]. \tag{10}$$

In other words,

$$\log \mathbb{E}(e^{-sX}) = \int_0^\infty (e^{-sx} - 1)\frac{1}{x^{\alpha+1}}dx.$$

This matches Equation 2. Hence, we conclude that the sum of log-normal variables in relativistic Brownian motion is an infinitely divisible α-stable Levy subordinator as long as the integral $\int_0^\infty [1 - e^{-sx}]\frac{dx}{x^{\alpha+1}}$ converges.

When $\alpha = \frac{1}{2}$, there is an analytic expression for the Γ function. Then Equation 10 becomes

$$\mathbb{E}(e^{-sX}) = \exp(-2\pi^{\frac{1}{2}}s^{\frac{1}{2}}). \tag{11}$$

Thus, we have shown that the sum of log-normal variables in relativistic Brownian motion is an infinitely divisible α-stable Levy subordinator for $0 < \alpha < 1$.

2.8 Alternative Large Number Approximation: Second Order

To include the second order term in the expansion, Recall for $0 < \alpha < 1$,

$$s^\alpha\Gamma(-\alpha) = \int_0^\infty [e^{-sx} - 1]\frac{dx}{x^{\alpha+1}}.$$

Also

$$-\frac{\partial}{\partial\alpha}(\frac{1}{x^{\alpha+1}}) = \frac{\log x}{x^{\alpha+1}}.$$

Furthermore,

$$\frac{\partial^2}{\partial\alpha^2}(\frac{1}{x^{\alpha+1}}) = \frac{\log^2 x}{x^{\alpha+1}}.$$

Then

$$\frac{\partial^2}{\partial\alpha^2}(s^\alpha\Gamma(-\alpha)) = \int_0^\infty \log^2 x[e^{-sx} - 1]\frac{dx}{x^{\alpha+1}}.$$

So

$$\mathbb{E}(e^{-sX_n}) = \exp[\Gamma(-\alpha)s^\alpha - \frac{1}{2\sigma_n^2}\frac{\partial^2}{\partial\alpha^2}(s^\alpha\Gamma(-\alpha))].$$

This is an infinitely divisible α-stable Levy subordinator with a correction term.

3 Numerical Verification

We study the case when $\alpha = \frac{1}{2}$ using numerical verification. $\mathbb{E}(e^{-sX_n})$ has been calculated analytically and numerically when $n = 10^4$.

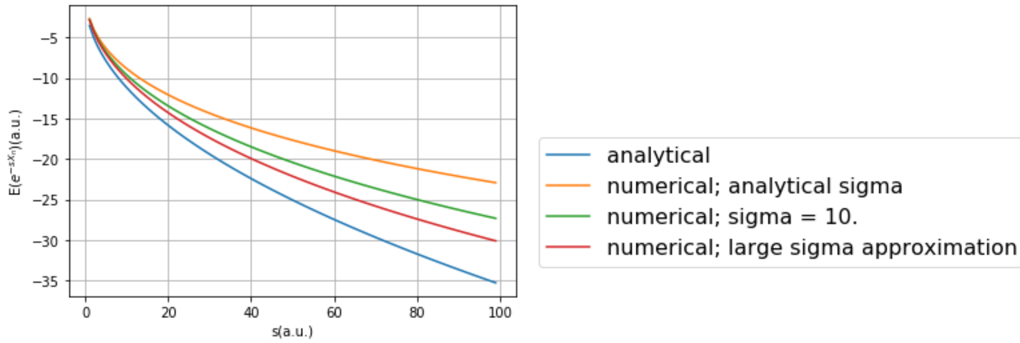

Figure 1: Analytical and Numerical Calculation for $\mathbb{E}(e^{-sX_n})$

In the curve "analytic", $\mathbb{E}(e^{-sX_n})$ was calculated using Equation 11. The curve "numerical; analytical sigma" was calculated by integrating the exponent in Equation 8 where σ was calculated using Equation 7. The curve "numerical; sigma = 10." was calculated using Equation 8 where sigma was the fixed value 10. The curve "numerical; large sigma approximation" was calculated from Equation 9.

We conclude that the numerical calculation using Equation 9 gives the best approximation. Moreover, all three approximations are valid when s is small.

4 Simulation

In the simulations, we add a new variable t to our model to control the time span we considered in each random process. Notice that $t = 1$ in the derivation. In general,

$$\mathbb{E}(e^{-sX_n}) = \exp[\int_0^\infty (e^{-sx} - 1)t \frac{\exp(-\frac{\log^2 x}{2\sigma^2})}{x^{\alpha+1}} dx].$$

We first generate $m = 200$ random copies of $n = 100$ log-normal variables x_{ij} with the standard deviation σ calculated using equation

$$\sigma = \frac{\sqrt{W(\frac{\alpha^2 n^2}{2\pi t^2})}}{\alpha}$$

which was derived from Equation 7 and the mean

$$\mu = -\alpha\sigma^2.$$

4.1 Distribution of the Steps

Let x_{ij} be the displacement each step. The actual distribution of 200×100 log-normal random variables is shown in the figure below.

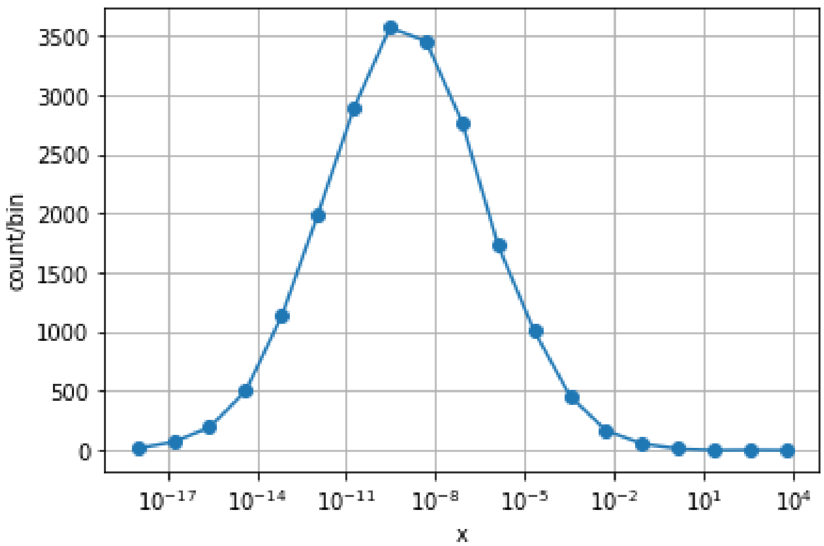

Figure 2: Distribution of log-normal variable x

4.2 Sample Path

We then sum the displacement per step (i.e. summing over i) in each copy to get the total displacement per trial X_j. An example displacement vs. number of steps curve is shown as below.

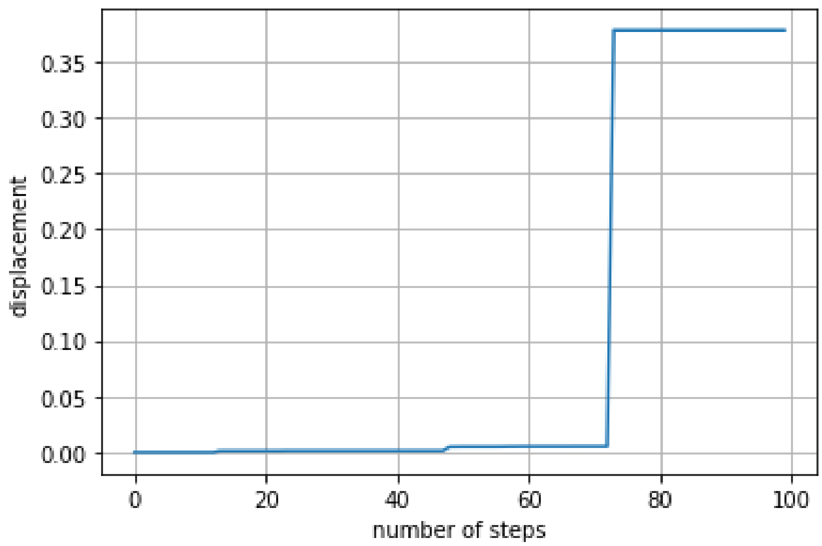

Figure 3: Displacement vs. Number of Steps

As shown in the figure above, the total displacement X_j is dominated by large jumps. We define large jumps as any step with a size larger than $\frac{\epsilon}{n}$. Therefore, the summed displacement with the rest of the steps is less than or equal to ϵ. We choose $\epsilon = 10^{-2}$.

4.3 Distribution of Large Steps

Then we count N_j the number of large steps per trial. We expect the distribution of N_j follows Poisson distribution.[8]

Definition 4.1. (The Poisson Process) *The Poisson process of intensity $\lambda > 0$ is a Levy process N taking values in $\mathbb{N} \cup \{0\}$ wherein each $N(t) = \pi(\lambda t)$, so that we have*

$$P(N(t) = n) = \frac{(\lambda t)^n}{n!} e^{-\lambda t} \tag{12}$$

for each $n = 0, 1, 2, ...$

We anticipate the distribution of N_j follows Poisson distribution since whether a big jump is happening at a given step is random.

$$P_{N_j = r} = \frac{\lambda^r}{r!} e^{-\lambda}$$

where λ is the mean of N_j. The distribution of N_j from simulation comparing with the distribution predicted by Poisson distribution from Equation 12 where $t = 1$ is shown as below.

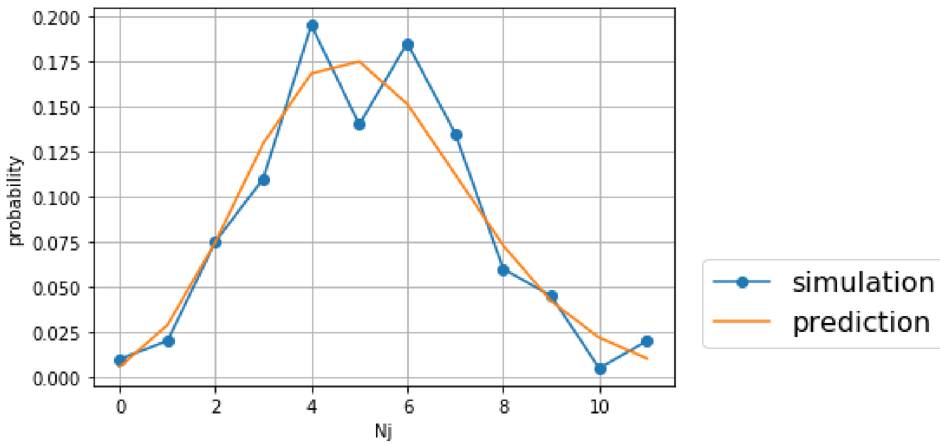

Figure 4: Distribution of the Number of Large Steps

From Figure 4, we can see N_j indeed follows Poisson distribution.

Figure 5 verifies that the decay of the probability of step follows a power law as the size of the step increases. This agrees with our expectation since we have shown that $\mathbb{E}(e^{-sX})$ is an infinitely divisible α-stable Levy subordinator. Notice that the linear correspondence in a log-log graph means a power law.

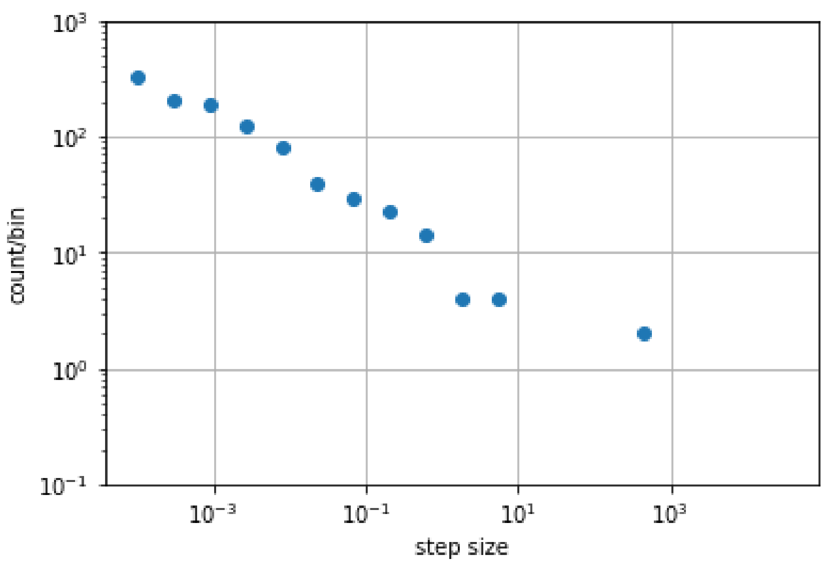

Figure 5: Distribution of the Size of Large Steps

5 Acknowledgements

Thank you to my faculty advisors, mentors, and role models at University of Rochester. Prof. Bigelow, Prof. Bodek, Prof. Demina, Prof. Eberly, Prof. Ferguson, Prof. Geba, Prof. Gonek, Prof. Haessig, Prof. Herman, Prof. Iosevich, Prof. Jochnowitz, Prof. Kleene, Prof. Madhu, Prof. Orr, Prof. Pakianathan, Prof. Rajeev, Prof. A. Tucker, Prof. Waddell, and Prof. Wolfs, I could not become the person I am today without your guidance or support.

Thank you to all of my instructors. You turned me into a physicist and mathematician.

Thank you to the positive and supportive staffs at University of Rochester who always tried their best to help me. Connie, Cynthia, Jeff, Jimmy, Laura, Linda, Lysa, and Mike, you made me feel valued.

Thank you to my mentors at SLAC National Accelerator Laboratory. Dr. Barklow, Dr. Coslovich, Dr. Dixon, Dr. Friedland, Dr. Fry, Dr. Hanuka, Ms. Hu, Dr. Zohar, the QMI community, and the machine learning group, you opened a door to a broader world for me.

Thank you to my peer mentors, mentees, friends, students, the University of Rochester SPS, the DSP group, teaching assistants and teaching interns who have taught me. I felt so lucky for meeting all of you.

Thank you to my family. You are my first ever role models.

References

[1] Peter Morters and Yuval Peres. *Brownian Motion.* Cambridge University Press, 2010.

[2] L. Fenton. The sum of log-normal probability distributions in scatter transmission systems. *IRE Transactions on Communications Systems*, 8(1):57–67, 1960.

[3] N. C. Beaulieu, A. A. Abu-Dayya, and P. J. McLane. Estimating the distribution of a sum of independent lognormal random variables. *IEEE Transactions on Communications*, 43(12):2869, 1995.

[4] S. M. Haas and J. H. Shapiro. Capacity of wireless optical communications. *IEEE Journal on Selected Areas in Communications*, 21(8):1346–1357, 2003.

[5] Moshe Arye Milevsky and Steven E. Posner. Asian options, the sum of lognormals, and the reciprocal gamma distribution. *The Journal of Financial and Quantitative Analysis*, 33(3):409–422, 1998.

[6] Jingxian Wu, N. B. Mehta, and Jin Zhang. Flexible lognormal sum approximation method. In *GLOBECOM '05. IEEE Global Telecommunications Conference, 2005.*, volume 6, pages 3413–3417, Nov 2005.

[7] Søren Asmussen and Leonardo Rojas-Nandayapa. Asymptotics of sums of lognormal random variables with gaussian copula. *Statistics Probability Letters*, 78(16):2709 – 2714, 2008.

[8] David Applebaum. *Lévy Processes and Stochastic Calculus*, volume 93. Cambridge University Press, Cambridge, 2004.

[9] Ken-iti Sato. *Lévy processes and infinitely divisible distributions.* Cambridge studies in advanced mathematics ; 68. Cambridge University Press, Cambridge, United Kingdom, rev. edition, corrected paperback edition. edition, 2013.

Appendices

Numerical Verification Program

```
#!/usr/bin/env python
# coding: utf-8

# In[ ]:


#Authors: Yue Wang, Prof. Rajeev


# In[2]:


#packages
import math
import numpy as np
import matplotlib.pyplot as plt
import scipy as sp
from scipy.optimize import curve_fit


# In[3]:


#alpha in the derivation and in alpha-stable Poisson process
alpha = 0.5


# In[4]:


#integrate the integral in the alpha-stable subordinator equation
    numerically
#fixed sigma = 10.
def zexp(m,s,alpha):
    #integration method
    sigma =
        10.#np.sqrt(sp.special.lambertw(alpha**2*m**2/(2*math.pi)))/alpha
    #print(sigma)
    binNum = 10**4
    binSize = 10**(-3)
    integral = 0.0
    for i in range(1,binNum):#iterate through the waves collected
        x = i*binSize
        integral = integral +
            (np.exp(-s*x)-1)*(np.exp(-(np.log(x))**2/(2*sigma**2))/(x**(alpha+1)))*binSize
```

```
    return integral


# In[5]:


#integrate the integral in the alpha-stable subordinator equation
    numerically
#sigma without approximation
#analytic sigma
def zexpAnalytic(m,s,alpha):
    #integration method
    sigma = np.sqrt(sp.special.lambertw(alpha**2*m**2/(2*math.pi)))/alpha
    #print(sigma)
    binNum = 10**4
    binSize = 10**(-3)
    integral = 0.0
    for i in range(1,binNum):#iterate through the waves collected
        x = i*binSize
        integral = integral +
            (np.exp(-s*x)-1)*(np.exp(-(np.log(x))**2/(2*sigma**2))/(x**(alpha+1)))*binSize
    return integral


# In[8]:


#integrate the integral in the alpha-stable subordinator equation
    numerically
#large sigma approximation
def z(m,s,alpha):
    #integration method
    binNum = 10**4
    binSize = 10**(-3)
    integral = 0.0
    for i in range(1,binNum):#iterate through the waves collected
        x = i*binSize
        integral = integral + (np.exp(-s*x)-1)/(x**(1+alpha))*binSize
    return integral



# In[15]:


#Integrate Z with different sigma approximations
binNumSplot = 10**2
binSize = 1
plotListZ = []
plotListS = []
```

```
plotListA = []
plotListZExp = [] #fixed sigma
plotListZExpA = [] #analytic
plotListZExp1 = [] #1st approximation
plotListZExp2 = [] #2nd approximation
for i in range (3,4):
    m = 10**i
    sList = []#list of s values
    zList = []
    aList = []
    zExpList = []
    zExpAList = []
    zExp1List = []
    zExp2List = []
    for j in range(1,binNumSplot):
        sList.append(j*binSize)
        zList.append(z(1,j*binSize,alpha))
        aList.append(-2*np.pi**(1/2)*(j*binSize)**(1/2))
        zExpList.append(zexp(m,j*binSize,alpha))
        zExpAList.append(zexpAnalytic(m,j*binSize,alpha))
        zExp1List.append(zexp1(m,j*binSize,alpha))
        zExp2List.append(zexp2(m,j*binSize,alpha))
    plotListS.append(sList)
    plotListZ.append(zList)
    plotListA.append(aList)
    plotListZExp.append(zExpList)
    plotListZExpA.append(zExpAList)
    plotListZExp1.append(zExp1List)
    plotListZExp2.append(zExp2List)


# In[19]:


#Plot Z integrated using different sigma approximations
for i in range(1):
    plt.plot(plotListS[i],plotListA[i],label = 'analytical')
    plt.plot(plotListS[i],plotListZExpA[i],label = 'numerical;
        analytical sigma')
    plt.plot(plotListS[i],plotListZExp[i],label = 'numerical; sigma =
        10.')
    plt.plot(plotListS[i],plotListZ[i],label = 'numerical; large sigma
        approximation')
plt.xlabel('s(a.u.)')
plt.ylabel('z(a.u.)')
plt.legend(loc=(1.05,0.1),fontsize=16.0)
plt.grid()
#plt.show()
plt.savefig('numericalZ.png',bbox_inches='tight')
```

Simulation Program

```
#!/usr/bin/env python
# coding: utf-8

# In[ ]:


#Authors: Yue Wang, Prof. Rajeev


# In[43]:


#packages
import math
import numpy as np
import matplotlib.pyplot as plt
import scipy as sp
from scipy.optimize import curve_fit
import numpy.ma as ma


# In[44]:


#epsilon in the derivation: used to define large steps
epsilon = 10**(-2)
#t: time range
t = 0.05


# In[45]:


#alpha in the derivation and in alpha-stable Poisson process
alpha = 1/2
n=100 #number of steps
m = 200 #number of copies
sigma =
    np.sqrt(sp.special.lambertw((alpha**2)*(n**2)/(2*math.pi*(t**2))))/alpha
mu = -alpha*sigma**2 #from the mathematica program


# In[47]:


#generate steps
xList = [] #X_{ij}
distList = [] #the list of summed x for each copy; X_j
```

```
for i in range(0,m):
    theta = np.random.normal(0., 1.0, n)
    x = np.exp(sigma*theta+mu)
    xList.append(x)
    distList.append(sum(x))


# In[48]:


#boolean list indicating if the each step in xList is large
booleanList = np.array(xList)[:][:]>epsilon/n


# In[49]:


NList = [] #N_j
for j in range(0,m):
    NList.append(sum(booleanList[j]))


# In[56]:


#histogram for number of large steps
Nhist,Nbin= np.histogram(NList,bins =12)
plt.plot(Nbin[:-1],Nhist,'o-')
plt.xlabel('Nj')
plt.ylabel('count/bin')

#plt.yscale("log")
#plt.xlim(0,10)
#plt.ylim(top=1000)
#plt.ylim(top=2200)
plt.grid()


# In[58]:


#define the Poisson distribution
#suppose the Poisson process is called N
#lambda_poisson is a float, which is the mean of of N
#r_possion is a float. It matches with the r in the main text
def poisson(lambda_poisson, r_possion):
    return
        ((lambda_poisson**r_possion)/(math.factorial(r_possion)))*np.exp(-lambda_poisson)
```

```
# In[59]:


#plot the expectated curve using values from Nbin as r values
#Nbin contains the x axis values in the histogram for number of large
    steps
poissonPlot = []
for Nj in Nbin[:-1]:
   poissonPlot.append(poisson(np.mean(NList),math.floor(Nj)))


# In[76]:


#histogram for number of large steps with expected poisson distribution
plt.plot(Nbin[:-1],Nhist/sum(Nhist),'o-',label = 'simulation')
plt.plot(Nbin[:-1],poissonPlot,label = 'prediction')
plt.xlabel('Nj')
plt.ylabel('probability')
plt.legend(loc=(1.05,0.1),fontsize=16.0)
#plt.yscale("log")
#plt.xlim(0,10)
#plt.ylim(top=50)
#plt.ylim(top=2200)
plt.grid()
plt.savefig('largestep.png',bbox_inches='tight')


# In[75]:


#histogram for step sizes
#Lognormal
disthist, distbin = np.histogram(np.array(xList).flatten(),bins
    =np.logspace(-18,5,20))
plt.plot(distbin[:-1],disthist,'o-')
plt.xlabel('x')
plt.ylabel('count/bin')

plt.xscale("log")
#plt.xlim(0,10)
#plt.ylim(top=1000)
#plt.ylim(top=2200)
plt.grid()
plt.savefig('xhistogram.png',bbox_inches='tight')


# In[62]:
```

```
#boolean matrix transpose
booleanListI = np.invert(booleanList)


# In[63]:


#select large steps in xList
mx = ma.masked_array(np.array(xList), mask=np.array(booleanListI))
print(mx)


# In[64]:


#select large steps
compressedMx = []
compressedMxFlattened = []
for j in range(len(mx)):
    compressedX = mx[j].compressed()
    buffer = []
    for i in range(len(compressedX)):
        buffer.append(float(compressedX[i]))
        compressedMxFlattened.append(float(compressedX[i]))
    compressedMx.append(buffer)


# In[79]:


#histogram for the sizes of large steps
#log-log normal
largehist, largebin = np.histogram(compressedMxFlattened,bins
    =np.logspace(-4,5,20))
plt.plot(largebin[:-1],largehist,'o')
plt.xlabel('step size')
plt.ylabel('count/bin')

plt.xscale("log")
plt.yscale("log")
#plt.xlim(0,10)
#plt.ylim(top=1000)
plt.ylim(10**(-1),10**3)
plt.grid()
plt.savefig('largestepsize.png',bbox_inches='tight')


# In[66]:
```

```
#plot a sample path
samplePath = []
for i in range(0,n):
    samplePath.append(sum(xList[0][:i]))


# In[67]:


#plot one path
plt.plot(samplePath)
plt.xlabel('number of steps')
plt.ylabel('displacement')

#plt.yscale("log")
#plt.xlim(0,10)
#plt.ylim(top=50)
#plt.ylim(top=2200)
plt.grid()
plt.savefig('examplePath.png',bbox_inches='tight')
```