Efficient Rare-event Simulation for Multiple Jump Events of Heavy-tailed Lévy Processes with Infinite Activities

Xingyu Wang   Chang-Han Rhee

Northwestern University

Introduction

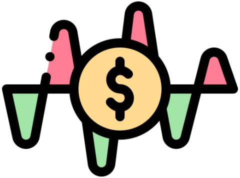
Introduction

Rare-event simulation for Lévy processes

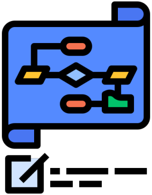
Introduction

Rare-event simulation for Lévy processes

Risk analysis



Option pricing

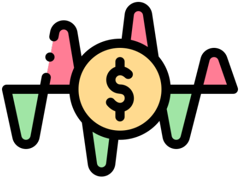


Queuing networks

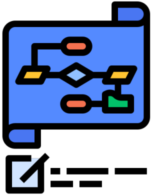
Introduction

Rare-event simulation for Lévy processes

Risk analysis



Option pricing



Queuing networks

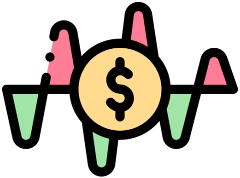
Task at hand: Through simulation, estimate

$$\mathbb{P}(X \in A)$$

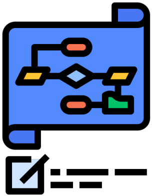
Introduction

Rare-event simulation for Lévy processes

Risk analysis



Option pricing



Queuing networks

Task at hand: Through simulation, estimate

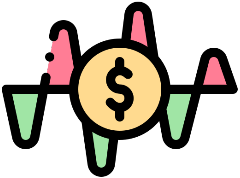
$$\mathbb{P}(X \in A)$$

Lévy process $\{X(t) : t \in [0,1]\}$ ↗

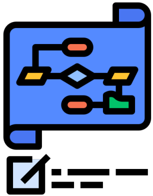
Introduction

Rare-event simulation for Lévy processes

Risk analysis



Option pricing



Queuing networks

Task at hand: Through simulation, estimate

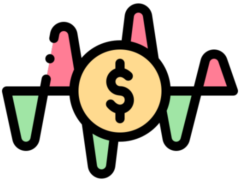
$$\mathbb{P}(X \in A)$$

Lévy process $\{X(t) : t \in [0,1]\}$ ↗ ↖ $\{X \in A\}$ is rare

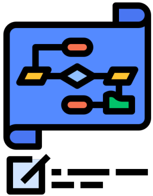
Introduction

Rare-event simulation for Lévy processes

Risk analysis



Option pricing



Queuing networks

Task at hand: Through simulation, estimate

$$\mathbb{P}(X \in A)$$

Lévy process $\{X(t) : t \in [0, 1]\}$ $\nearrow$ $\nwarrow$ $\{X \in A\}$ is rare

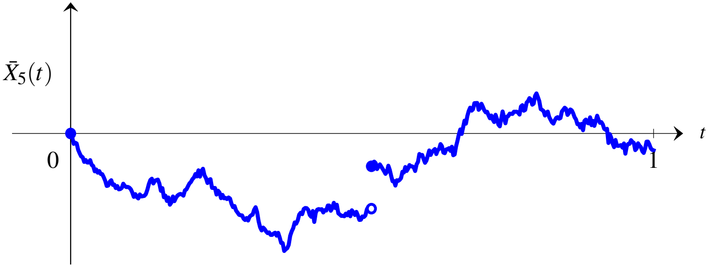
- **Why is this difficult?**

Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$

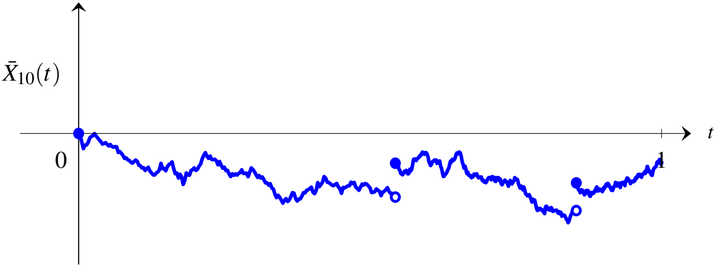
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



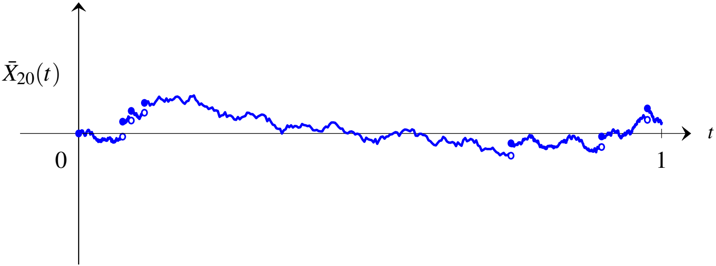
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



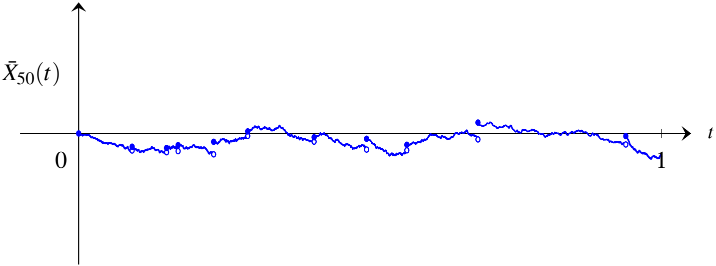
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



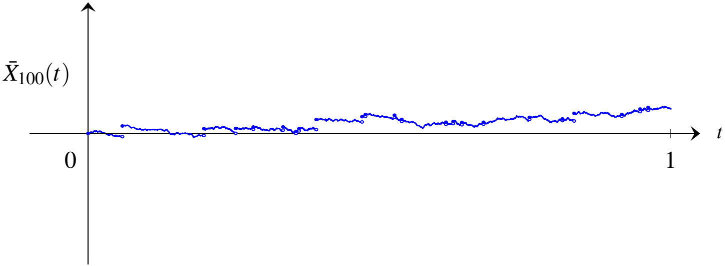
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$

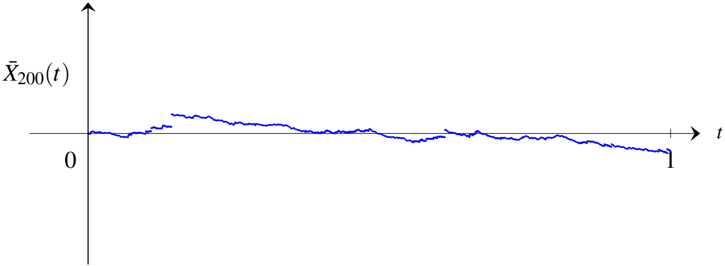


Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$

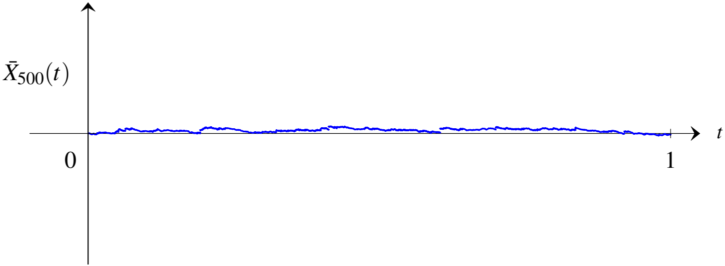


Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



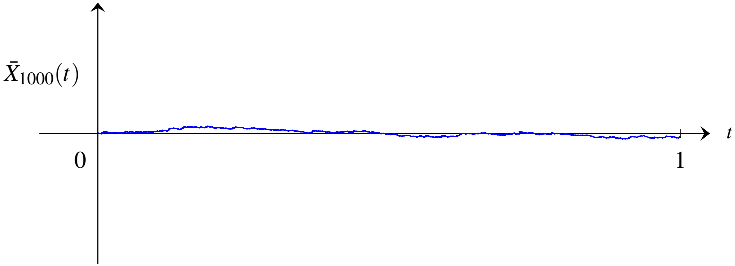
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



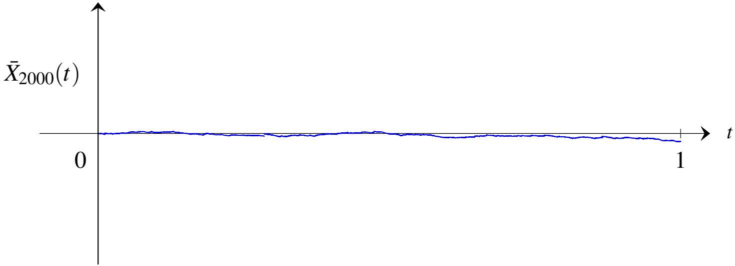
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



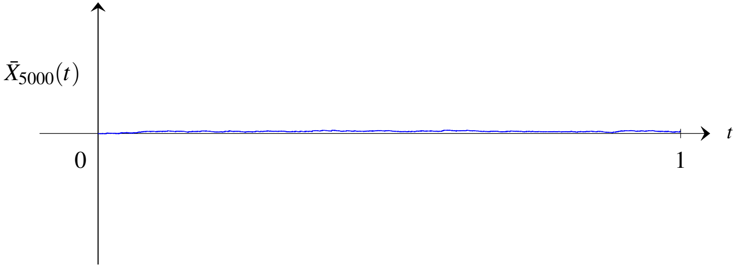
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



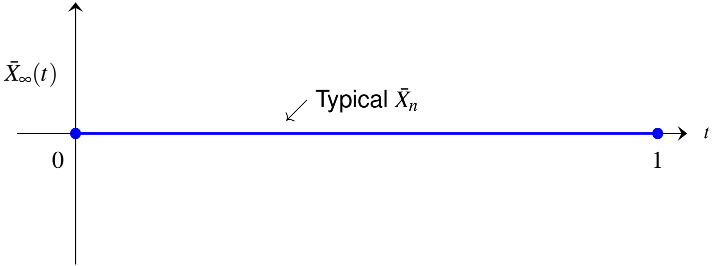
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



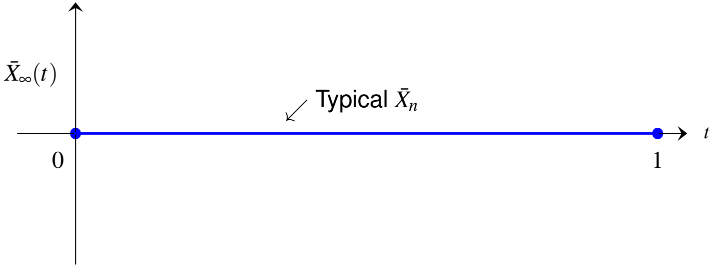
Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



Difficulty 1: Inefficiency

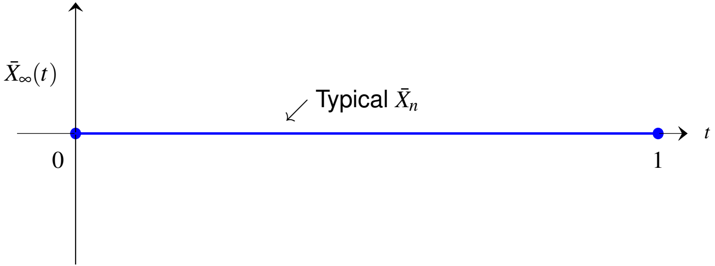
Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



- $\mathbf{0} \notin A$

Difficulty 1: Inefficiency

Scaled processes: $\bar{X}_n \triangleq \{X(nt)/n : t \in [0,1]\}$



- $\mathbf{0} \notin A \Rightarrow \mathbb{P}(\bar{X}_n \in A) \to 0.$

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$

- **Required # of samples for 10% relative error**:

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$

- **Required # of samples for 10% relative error**:

	n = 100	n = 1000	n = 10000
Crude Monte-Carlo method ($\alpha = 3$)	$\approx 10^{15}$	$\approx 10^{19}$	$\approx 10^{23}$
Crude Monte-Carlo method ($\alpha = 5$)	$\approx 10^{20}$	$\approx 10^{28}$	$\approx 10^{36}$

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$

- **Required # of samples for 10% relative error**:

↙Polynomial rate

	n = 100	n = 1000	n = 10000
Crude Monte-Carlo method ($\alpha = 3$)	$\approx 10^{15}$	$\approx 10^{19}$	$\approx 10^{23}$
Crude Monte-Carlo method ($\alpha = 5$)	$\approx 10^{20}$	$\approx 10^{28}$	$\approx 10^{36}$

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$
- **Strong Efficiency:** estimators $(L_n)_{n\geq 1}$ for $\mathbb{P}(\bar{X}_n \in A)$

- **Required # of samples for 10% relative error**:

Polynomial rate

	n = 100	n = 1000	n = 10000
Crude Monte-Carlo method ($\alpha = 3$)	$\approx 10^{15}$	$\approx 10^{19}$	$\approx 10^{23}$
Crude Monte-Carlo method ($\alpha = 5$)	$\approx 10^{20}$	$\approx 10^{28}$	$\approx 10^{36}$

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$
- **Strong Efficiency:** estimators $(L_n)_{n\geq 1}$ for $\mathbb{P}(\bar{X}_n \in A)$

$$\limsup_n \mathbb{E}L_n^2/(\mathbb{E}L_n)^2 < \infty.$$

- **Required # of samples for 10% relative error**:

↙Polynomial rate

	n = 100	n = 1000	n = 10000
Crude Monte-Carlo method ($\alpha = 3$)	$\approx 10^{15}$	$\approx 10^{19}$	$\approx 10^{23}$
Crude Monte-Carlo method ($\alpha = 5$)	$\approx 10^{20}$	$\approx 10^{28}$	$\approx 10^{36}$

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$
- **Strong Efficiency:** estimators $(L_n)_{n\geq 1}$ for $\mathbb{P}(\bar{X}_n \in A)$

$$\limsup_n \mathbb{E}L_n^2/(\mathbb{E}L_n)^2 < \infty.$$

- **Required # of samples for 10% relative error**:

↙Polynomial rate

	n = 100	n = 1000	n = 10000
Crude Monte-Carlo method ($\alpha = 3$)	$\approx 10^{15}$	$\approx 10^{19}$	$\approx 10^{23}$
Crude Monte-Carlo method ($\alpha = 5$)	$\approx 10^{20}$	$\approx 10^{28}$	$\approx 10^{36}$
Strongly efficient estimator ($\alpha = 3$)	$\approx 10^{7}$	$\approx 10^{7}$	$\approx 10^{7}$
Strongly efficient estimator ($\alpha = 5$)	$\approx 10^{10}$	$\approx 10^{10}$	$\approx 10^{10}$

Difficulty 1: Inefficiency

- $\mathbb{P}(\bar{X}_n \in A) \to 0$
- **Strong Efficiency:** estimators $(L_n)_{n\geq 1}$ for $\mathbb{P}(\bar{X}_n \in A)$

$$\limsup_n \mathbb{E}L_n^2/(\mathbb{E}L_n)^2 < \infty.$$

- **Required # of samples for 10% relative error**:

↙Polynomial rate

	n = 100	n = 1000	n = 10000
Crude Monte-Carlo method ($\alpha = 3$)	$\approx 10^{15}$	$\approx 10^{19}$	$\approx 10^{23}$
Crude Monte-Carlo method ($\alpha = 5$)	$\approx 10^{20}$	$\approx 10^{28}$	$\approx 10^{36}$
Strongly efficient estimator ($\alpha = 3$)	$\approx 10^{7}$	$\approx 10^{7}$	$\approx 10^{7}$
Strongly efficient estimator ($\alpha = 5$)	$\approx 10^{10}$	$\approx 10^{10}$	$\approx 10^{10}$

↖Uniform bound for any n

Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"

Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process X

Difficulty 2: Simulation of X with Infinite Activity

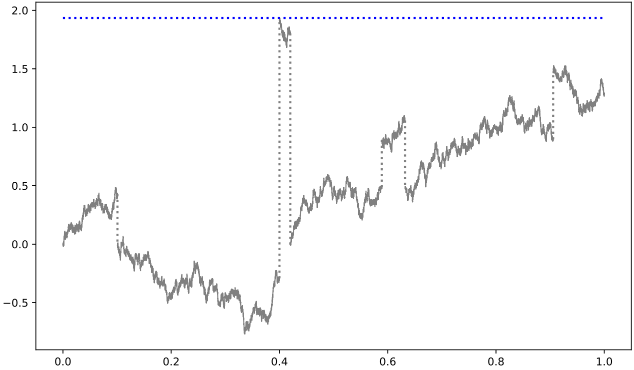
- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path

Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path
- Hard to simulate **extrema** of X

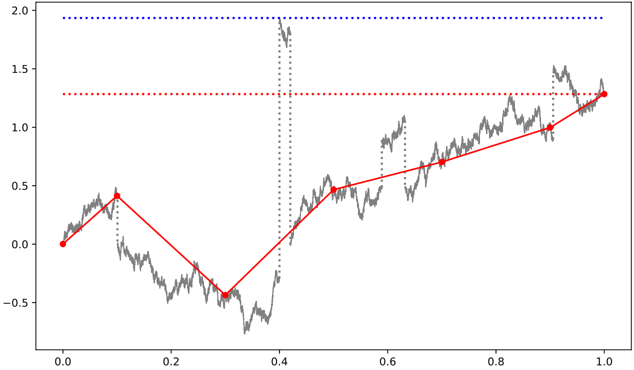
Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path
- Hard to simulate **extrema** of X



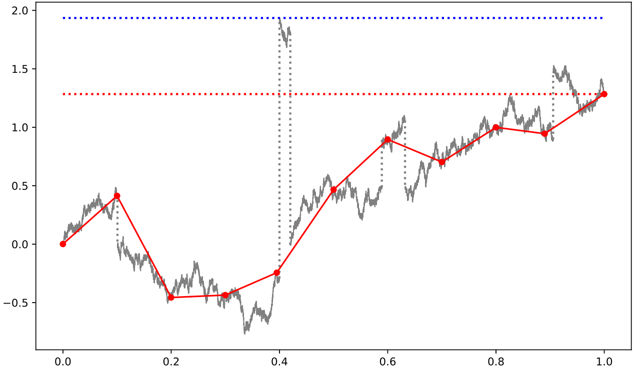
Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path
- Hard to simulate **extrema** of X



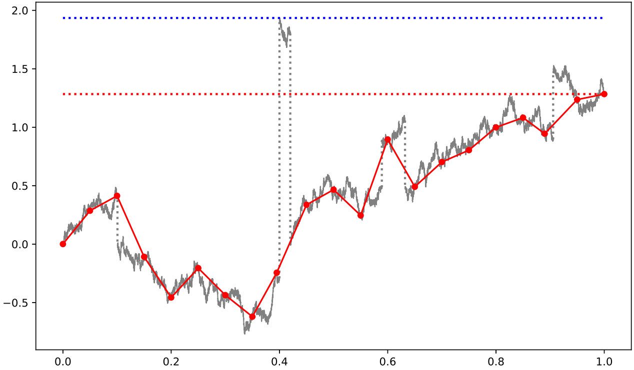
Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path
- Hard to simulate **extrema** of X



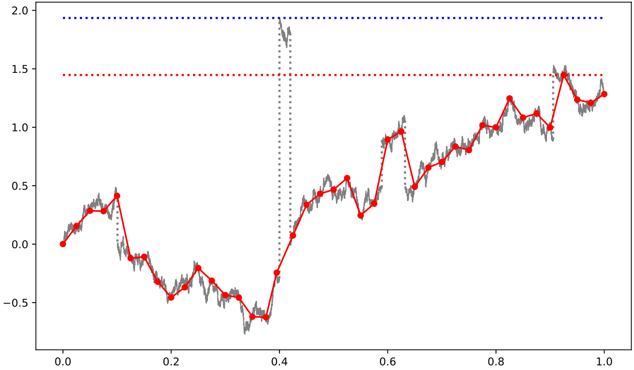
Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path
- Hard to simulate **extrema** of X



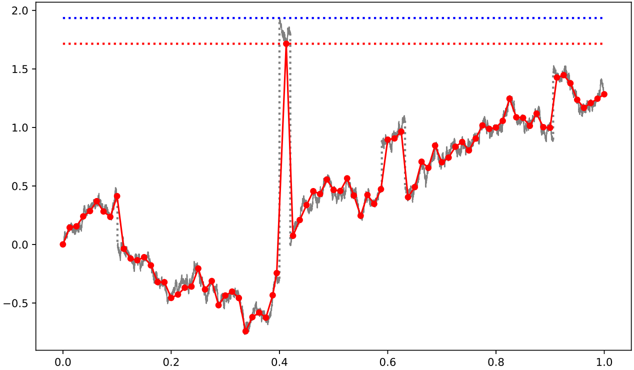
Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path
- Hard to simulate **extrema** of X



Difficulty 2: Simulation of X with Infinite Activity

- "Simulate X, check if $X \in A$"
- **Infinite activities** in process $X \Rightarrow$ cannot simulate the entire path
- Hard to simulate **extrema** of X



Algorithm

Problem Setting

Lévy Process X

Problem Setting

Lévy Process X

- Infinite activities (not compound Poisson), $\mathbb{E}X(t) = 0\ \forall t$

Problem Setting

Lévy Process X

- Infinite activities (not compound Poisson), $\mathbb{E}X(t) = 0 \ \forall t$
- Heavy-tailed: $X(1) \in RV_{-\alpha}$ with $\alpha > 1$

Problem Setting

Lévy Process X

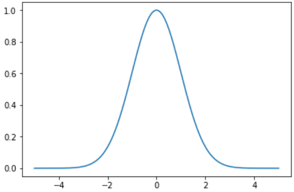
- Infinite activities (not compound Poisson), $\mathbb{E}X(t) = 0\ \forall t$
- Heavy-tailed: $X(1) \in RV_{-\alpha}$ with $\alpha > 1$

  $\Rightarrow$ Tail of the distribution for $X(t) - X(s)$ resembles $x^{-\alpha}$.

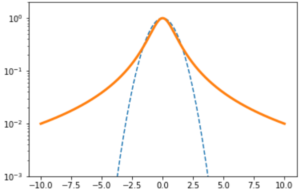
Lévy Process X

- Infinite activities (not compound Poisson), $\mathbb{E}X(t) = 0 \; \forall t$
- Heavy-tailed: $X(1) \in RV_{-\alpha}$ with $\alpha > 1$

  $\Rightarrow$ Tail of the distribution for $X(t) - X(s)$ resembles $x^{-\alpha}$.



Gaussian



Heavy-tailed

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A

Running example (reinsurance): "all claim $< b$, still go bankrupt (total loss $> a$)"

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A

Running example (reinsurance): "all claim $< b$, still go bankrupt (total loss $> a$)"

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A

Running example (reinsurance): "all claim $< b$, still go bankrupt (total loss $> a$)"

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

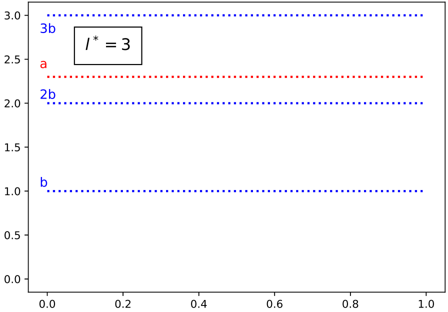
- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A

Running example (reinsurance): "all claim $< b$, still go bankrupt (total loss $> a$)"

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A

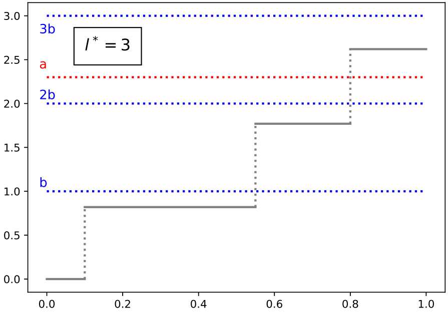


← Set A (reinsurance)

Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

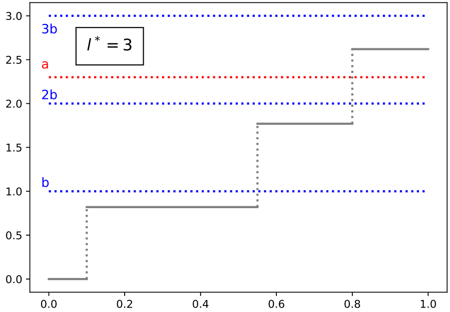
- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A



← Set A (reinsurance)

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A

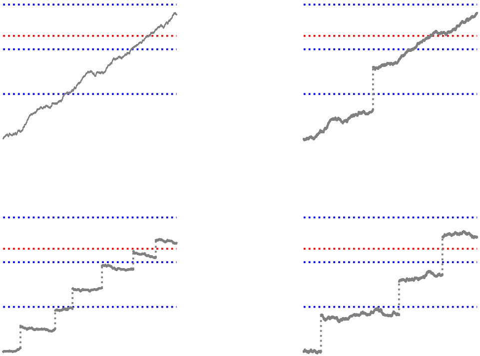


← Set A (reinsurance)

Large Deviation Principle for Heavy-tailed Lévy Processes

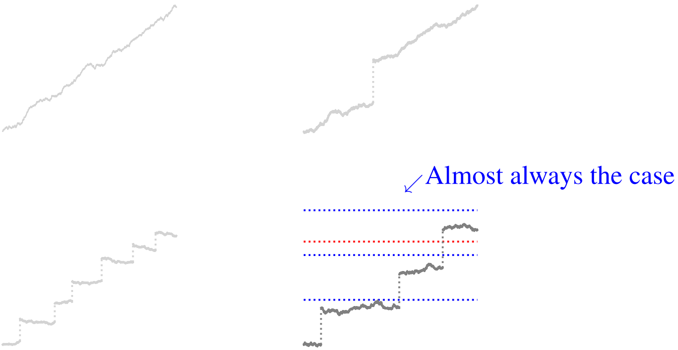
Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A



Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

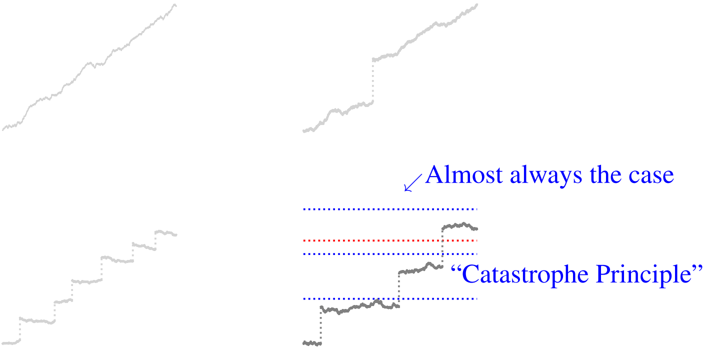
- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A



Large Deviation Principle for Heavy-tailed Lévy Processes

Rhee, Blanchet & Zwart (2019): For $A_n = \{\bar{X}_n \in A\}$, we have

- How rare the events are: $\mathbb{P}(A_n) \sim C(A) \cdot n^{-l^*(\alpha-1)}$
- How the rare events occur: $\bar{X}_n$ will most likely look like l^*-jump step function
- l^*: Minimal # of jumps for a **step function** to be in A



Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

$$\mathbb{E}_{\mathbb{P}} \mathbb{1}\{X \in A\} = \mathbb{E}_{\mathbb{Q}} \frac{d\mathbb{P}}{d\mathbb{Q}} \mathbb{1}\{X \in A\}$$

Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

$$\mathbb{E}_{\mathbb{P}} \mathbb{1}\{X \in A\} = \mathbb{E}_{\mathbb{Q}} \frac{d\mathbb{P}}{d\mathbb{Q}} \mathbb{1}\{X \in A\}$$

- Extend the framework in **Chen et al. (2019)**

$$\mathbb{Q}(\cdot) = w\mathbb{P}(\cdot) + (1-w)\mathbb{P}(\cdot|\bar{X}_n \in B).$$

Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

$$\mathbb{E}_{\mathbb{P}} \mathbb{1}\{X \in A\} = \mathbb{E}_{\mathbb{Q}} \frac{d\mathbb{P}}{d\mathbb{Q}} \mathbb{1}\{X \in A\}$$

- Extend the framework in **Chen et al. (2019)**

$$\mathbb{Q}(\cdot) = w\mathbb{P}(\cdot) + (1-w)\mathbb{P}(\cdot|\bar{X}_n \in B).$$

- $d\mathbb{P}/d\mathbb{Q} < \infty$.

Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

$$\mathbb{E}_{\mathbb{P}} \mathbb{1}\{X \in A\} = \mathbb{E}_{\mathbb{Q}} \frac{d\mathbb{P}}{d\mathbb{Q}} \mathbb{1}\{X \in A\}$$

- Extend the framework in **Chen et al. (2019)**

$$\mathbb{Q}(\cdot) = w\mathbb{P}(\cdot) + (1-w)\mathbb{P}(\cdot|\bar{X}_n \in B).$$

- $d\mathbb{P}/d\mathbb{Q} < \infty$.
- $\mathbb{Q}(\cdot)$ approximates $\mathbb{P}(\cdot|\bar{X}_n \in A)$.

Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

$$\mathbb{E}_{\mathbb{P}} \mathbb{1}\{X \in A\} = \mathbb{E}_{\mathbb{Q}} \frac{d\mathbb{P}}{d\mathbb{Q}} \mathbb{1}\{X \in A\}$$

- Extend the framework in **Chen et al. (2019)**

$$\mathbb{Q}(\cdot) = w\mathbb{P}(\cdot) + (1-w)\mathbb{P}(\cdot|\bar{X}_n \in B).$$

- $d\mathbb{P}/d\mathbb{Q} < \infty$.
- $\mathbb{Q}(\cdot)$ approximates $\mathbb{P}(\cdot|\bar{X}_n \in A)$. $\leftarrow$most likely has l^* large jumps

Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

$$\mathbb{E}_{\mathbb{P}} \mathbb{1}\{X \in A\} = \mathbb{E}_{\mathbb{Q}} \frac{d\mathbb{P}}{d\mathbb{Q}} \mathbb{1}\{X \in A\}$$

- Extend the framework in **Chen et al. (2019)**

$$\mathbb{Q}(\cdot) = w\mathbb{P}(\cdot) + (1-w)\mathbb{P}(\cdot|\bar{X}_n \in B).$$

- $d\mathbb{P}/d\mathbb{Q} < \infty$.
- $\mathbb{Q}(\cdot)$ approximates $\mathbb{P}(\cdot|\bar{X}_n \in A)$. ←most likely has l^* large jumps
-

$$f \in B \Leftrightarrow f \text{ has at least } l^* \text{ jumps that are larger than } \gamma$$

Strongly Efficient Importance Sampling Algorithm

- **Importance sampling:** Generate $X \sim \mathbb{Q}$ instead of $\mathbb{P}$,

$$\mathbb{E}_{\mathbb{P}} \mathbb{1}\{X \in A\} = \mathbb{E}_{\mathbb{Q}} \frac{d\mathbb{P}}{d\mathbb{Q}} \mathbb{1}\{X \in A\}$$

- Extend the framework in **Chen et al. (2019)**

$$\mathbb{Q}(\cdot) = w\mathbb{P}(\cdot) + (1-w)\mathbb{P}(\cdot|\bar{X}_n \in B).$$

We proposed algorithm to sample from $\mathbb{P}(\cdot|\bar{X}_n \in B)$

- $d\mathbb{P}/d\mathbb{Q} < \infty$.
- $\mathbb{Q}(\cdot)$ approximates $\mathbb{P}(\cdot|\bar{X}_n \in A)$. ←most likely has l^* large jumps
-

$$f \in B \Leftrightarrow f \text{ has at least } l^* \text{ jumps that are larger than } \gamma$$

How to Debias Discrete Approximation Error

- Want to know $\mathbb{E}Y$; Cannot simulate Y

How to Debias Discrete Approximation Error

- Want to know $\mathbb{E}Y$; Cannot simulate Y
- Approximations: $\mathbb{E}Y_m \to \mathbb{E}Y$

How to Debias Discrete Approximation Error

- Want to know $\mathbb{E}Y$; Cannot simulate Y
- Approximations: $\mathbb{E}Y_m \to \mathbb{E}Y$
- **Rhee and Glynn (2015):** Randomized Monte-Carlo Estimator

$$Z = \sum_{m=1}^{\tau} \frac{Y_m - Y_{m-1}}{\mathbb{P}(\tau \geq m)}$$

How to Debias Discrete Approximation Error

- Want to know $\mathbb{E}Y$; Cannot simulate Y
- Approximations: $\mathbb{E}Y_m \to \mathbb{E}Y$
- **Rhee and Glynn (2015):** Randomized Monte-Carlo Estimator

$$Z = \sum_{m=1}^{\tau} \frac{Y_m - Y_{m-1}}{\mathbb{P}(\tau \geq m)}$$

- Finite computational effort: Only need $Y_1, \cdots, Y_\tau$.

How to Debias Discrete Approximation Error

- Want to know $\mathbb{E}Y$; Cannot simulate Y
- Approximations: $\mathbb{E}Y_m \to \mathbb{E}Y$
- **Rhee and Glynn (2015):** Randomized Monte-Carlo Estimator

$$Z = \sum_{m=1}^{\tau} \frac{Y_m - Y_{m-1}}{\mathbb{P}(\tau \geq m)}$$

- Finite computational effort: Only need $Y_1, \cdots, Y_\tau$.
- Unbiasedness: For appropriate τ, $\mathbb{E}Z = \mathbb{E}Y$.

How to Debias Discrete Approximation Error

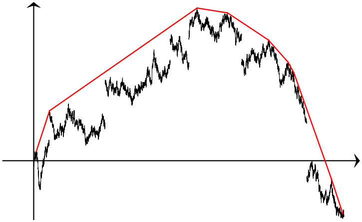
- Want to know $\mathbb{E}Y$; Cannot simulate Y
- Approximations: $\mathbb{E}Y_m \to \mathbb{E}Y$
- **Rhee and Glynn (2015):** Randomized Monte-Carlo Estimator

$$Z = \sum_{m=1}^{\tau} \frac{Y_m - Y_{m-1}}{\mathbb{P}(\tau \geq m)}$$

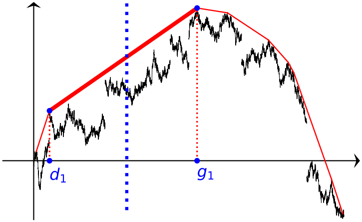
- Finite computational effort: Only need $Y_1, \cdots, Y_\tau$.
- Unbiasedness: For appropriate τ, $\mathbb{E}Z = \mathbb{E}Y$.
- **Question:** What approximation $(Y_m)_{m\geq 1}$ should we use?

Efficient Approximation via Concave Majorization

Concave majorant of X

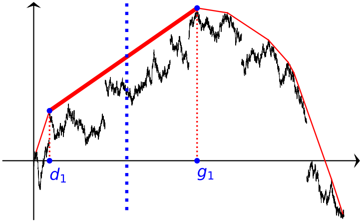


Concave majorant of X



Efficient Approximation via Concave Majorization

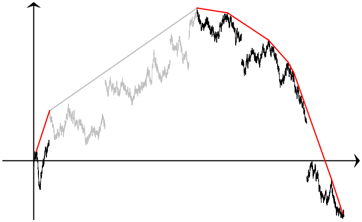
Concave majorant of X



$l_1 = g_1 - d_1, \ \ s_1 = X(g_1) - X(d_1)$

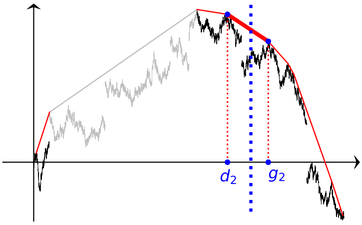
Efficient Approximation via Concave Majorization

Concave majorant of X



Efficient Approximation via Concave Majorization

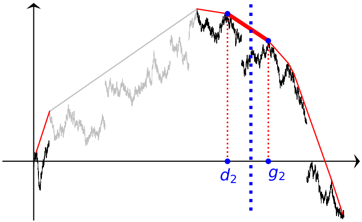
Concave majorant of X



$l_2 = g_2 - d_2, \;\; s_2 = X(g_2) - X(d_2)$

Efficient Approximation via Concave Majorization

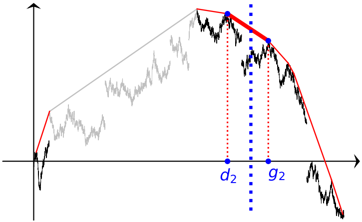
Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

Efficient Approximation via Concave Majorization

Concave majorant of X

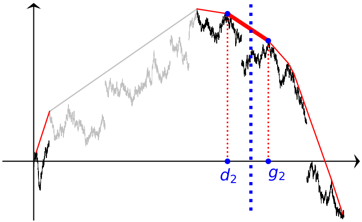


$\left(l_n, s_n\right)_{n\geq 1}$

Stick breaking procedure

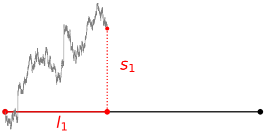
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

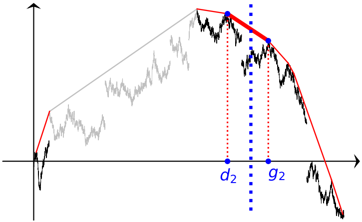
Stick breaking procedure



$l_1 \sim \text{Unif}(0,1)$

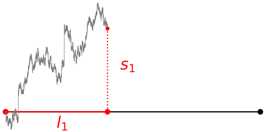
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

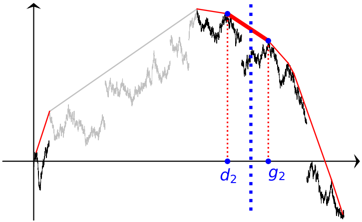
Stick breaking procedure



$l_1 \sim \text{Unif}(0,1),\ s_1 \stackrel{d}{=} X(l_1)$

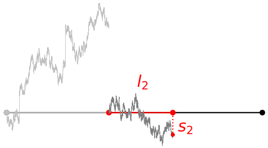
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

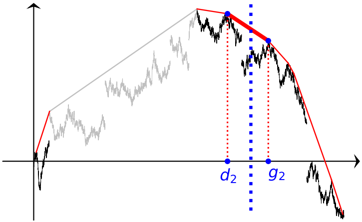
Stick breaking procedure



$l_2 \sim \text{Unif}(0, 1 - l_1)$

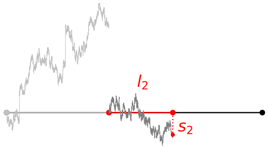
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

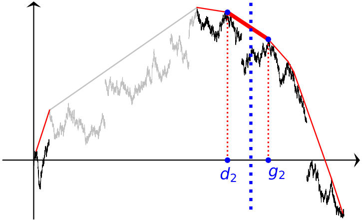
Stick breaking procedure



$l_2 \sim \mathrm{Unif}(0, 1-l_1),\ s_2 \overset{d}{=} X(l_2)$

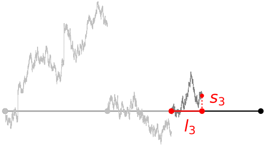
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

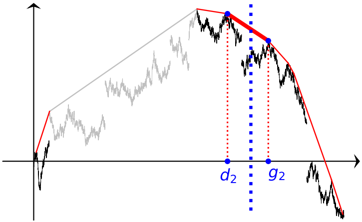
Stick breaking procedure



$l_3 \sim \text{Unif}(0, 1 - l_1 - l_2)$

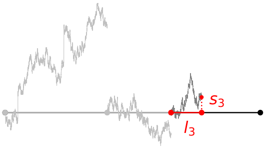
Efficient Approximation via Concave Majorization

Concave majorant of X



$\big(l_n, s_n\big)_{n\geq 1}$

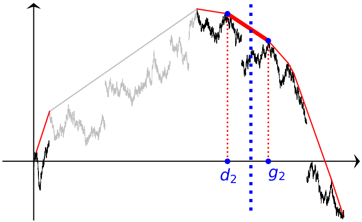
Stick breaking procedure



$l_3 \sim \mathrm{Unif}(0, 1 - l_1 - l_2), s_3 \stackrel{d}{=} X(l_3)$

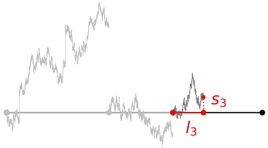
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

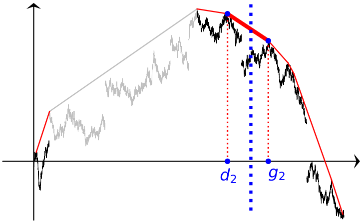
Stick breaking procedure



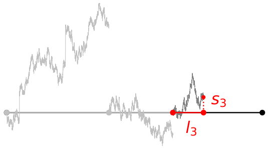
$\left(l_n, s_n\right)_{n\geq 1}$

Efficient Approximation via Concave Majorization

Concave majorant of X



Stick breaking procedure

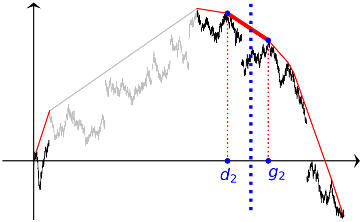


$$(l_n, s_n)_{n \geq 1} \overset{d}{=} (l_n, s_n)_{n \geq 1}$$

Pitman and Bravo (2012)

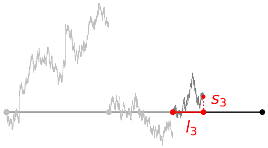
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

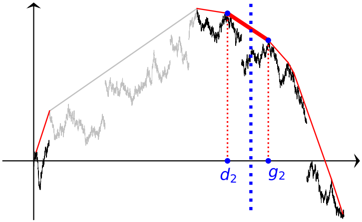
Stick breaking **approximation**



$\left(l_n, s_n\right)_{n=1}^{M}$

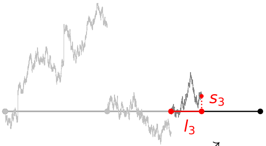
Efficient Approximation via Concave Majorization

Concave majorant of X



$\left(l_n, s_n\right)_{n\geq 1}$

Stick breaking **approximation**



Geometric convergence rate↗

$\left(l_n, s_n\right)_{n=1}^{M}$

Theoretical Analysis of the Algorithm

Strong Efficiency

Algorithm 1 Efficient Estimation of $\mathbb{P}(A_n)$

Require: $w \in (0,1), \gamma > 0, \rho \in (0,1)$

if $\text{Unif}(0,1) < w$ **then** ▷ Sample J_n from $\mathbb{Q}$
  Sample $J_n = \sum_{i=1}^{k} z_i \mathbb{1}_{[u_i,n]}$ from $\mathbb{P}$
else
  Sample $J_n = \sum_{i=1}^{k} z_i \mathbb{1}_{[u_i,n]}$ from $\mathbb{P}(\cdot|B_n^\gamma)$ using Algorithm 2
end if
Let $u_0 = 0, u_{k+1} = n$.
Sample $\tau \sim \text{Geom}(\rho)$ ▷ Decide Truncation Index τ
for $i = 0,1,\cdots,k$ **do** ▷ Generate Stick Lengths, and Decide Increments
  Sample $U_1^{(i)} \sim \text{Unif}(0,1)$. Let $l_1^{(i)} = U_1^{(i)}(u_{i+1} - u_i)$
  Sample $\xi_{i,1} \sim F_{\widetilde{X}}(\cdot, l_1^{(i)})$
  for $j = 2,3,\cdots,\lceil \log_2(n^2) \rceil + \tau$ **do**
    Sample $U_j^{(i)} \sim \text{Unif}(0,1)$. Let $l_j^{(i)} = U_j^{(i)}(u_{i+1} - u_i - l_1^{(i)} - l_2^{(i)} - \cdots - l_{j-1}^{(i)})$
    Sample $\xi_{i,j} \sim F_{\widetilde{X}}(\cdot, l_j^{(i)})$
  end for
  Let $l^{(i)}_{\lceil \log(n^2) \rceil + \tau + 1} = u_{i+1} - u_i - l_1^{(i)} - l_2^{(i)} - \cdots - l^{(i)}_{\lceil \log(n^2) \rceil + \tau}$
  Sample $\xi_{i,\lceil \log_2(n^2) \rceil + \tau + 1} \sim F_{\widetilde{X}}(\cdot, l^{(i)}_{\lceil \log_2(n^2) \rceil + \tau + 1})$
end for
for $m = 0,1,\cdots,\tau$ **do** ▷ Evaluate $Y_{n,m}$
  for $i = 0,1,2,\cdots,k$ **do**
    Let $\widetilde{M}_m^{(i)} = \sum_{l=0}^{i-1} \sum_{j=1}^{\lceil \log_2(n^2) \rceil + \tau + 1} \xi_{l,j}^m + \sum_{j=1}^{\lceil \log_2(n^2) \rceil + \tau} (\xi_{i,j}^m)^+$
  end for
  Let $Y_{n,m} = \mathbb{1}\{ \max_{i=0,1,\cdots,k} \widetilde{M}_m^{(i)} + J_n(u_i) \geq na \}$
end for
Let $Z_n = Y_{n,0} + \sum_{m=1}^{\tau} (Y_{n,m} - Y_{n,m-1}) / \rho^{m-1}$ ▷ Return the Estimator L_n
if $\max_{i=1,\cdots,k} z_i > b$ **then**
  Return $L_n = 0$.
else
  Let $\lambda_n = n\nu[n\gamma, \infty)$, $p_n = 1 - \sum_{l=0}^{l^*-1} e^{-\lambda_n} \frac{\lambda_n^l}{l!}$, $I_n = \mathbb{1}\{J_n \in B_n^\gamma\}$
  Return $L_n = Z_n / (w + \frac{1-w}{p_n} I_n)$
end if

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption: For any $z_0 > 0$, there exist $C > 0, \beta > 0, \theta \in (0,1]$ such that for any $t > 0, z \geq z_0, x \in \mathbb{R}, \delta \in [0,1]$, we have

$$\mathbb{P}(X^{<z}(t) \in [x, x+\delta]) \leq \frac{C}{t^\beta \wedge 1}\delta^\theta; \tag{A1}$$

where $X^{<z}$ is the Lévy process with the generating triplet $(c_X, \sigma^2, \nu|_{(-\infty,z)})$.

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Two sufficient conditions for (A1):

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Two sufficient conditions for (A1):

- Either there is Brownian motion in X;

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Two sufficient conditions for (A1):

- Either there is Brownian motion in X;
- Or for Lévy measure ν

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Two sufficient conditions for (A1):

- Either there is Brownian motion in X;
- Or for Lévy measure ν, $\nu(\mathbb{R}) = \infty$ and $\nu(\mathbb{R} \backslash [-x, x]) > x^{-\beta}$ when x close to 0.

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Two sufficient conditions for (A1):

- Either there is Brownian motion in X; $\leftarrow$ X with infinite activities
- Or for Lévy measure ν, $\nu(\mathbb{R}) = \infty$ and $\nu(\mathbb{R} \backslash [-x, x]) > x^{-\beta}$ when x close to 0.

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Two sufficient conditions for (A1):

- Either there is Brownian motion in X; $\leftarrow X$ with infinite activities
- Or for Lévy measure ν, $\nu(\mathbb{R}) = \infty$ and $\nu(\mathbb{R} \setminus [-x, x]) > x^{-\beta}$ when x close to 0.

  X with infinite activities$\nearrow$

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

Two sufficient conditions for (A1):

- Either there is Brownian motion in X; $\leftarrow X$ with infinite activities
- Or for Lévy measure ν, $\nu(\mathbb{R}) = \infty$ and $\nu(\mathbb{R}\backslash[-x,x]) > x^{-\beta}$ when x close to 0.

X with infinite activities↗ ↖Only requirement: not too slow

Strong Efficiency

Theorem (W. and Rhee, 2020)

If **assumption (A1)** holds, then our importance sampling algorithm is unbiased and strongly efficient.

Assumption:

$$\text{Distribution of } X(t) \text{ does not concentrate at any } x. \tag{A1}$$

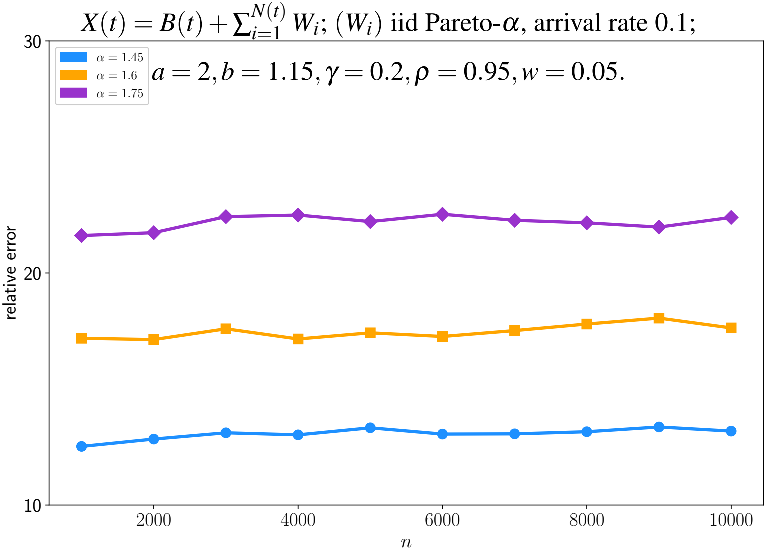
Two sufficient conditions for (A1):

- Either there is Brownian motion in X;
- Or for Lévy measure ν, $\nu(\mathbb{R}) = \infty$ and $\nu(\mathbb{R} \backslash [-x, x]) > x^{-\beta}$ when x close to 0.

The algorithm is applicable to a broad class of heavy-tailed Lévy processes.

Numerical Experiments

Experiment Results: Reinsurance Case

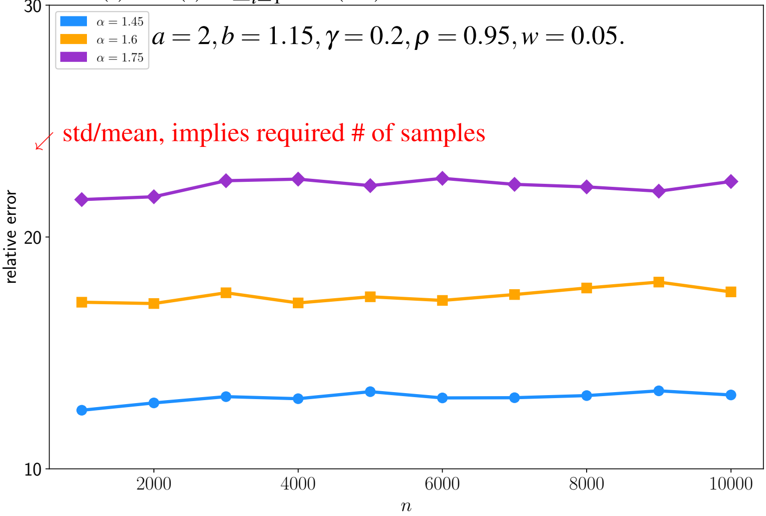


Experiment Results: Reinsurance Case

$X(t) = B(t) + \sum_{i=1}^{N(t)} W_i$; (W_i) iid Pareto-α, arrival rate 0.1;

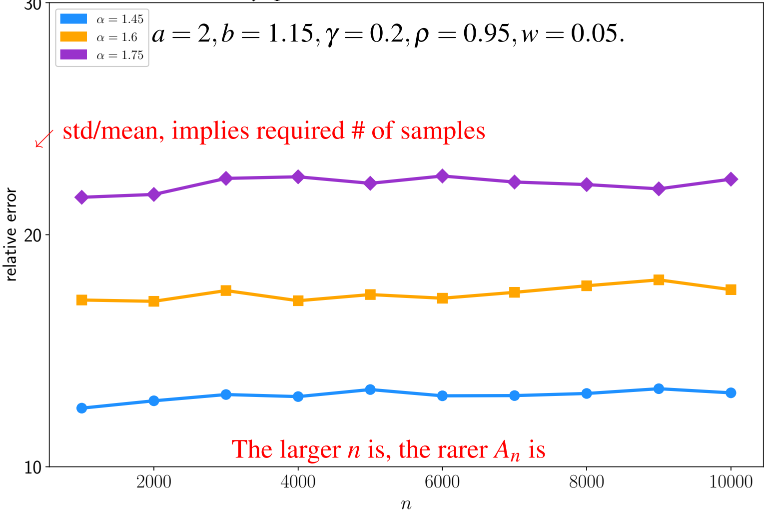
$a = 2, b = 1.15, \gamma = 0.2, \rho = 0.95, w = 0.05.$

std/mean, implies required # of samples

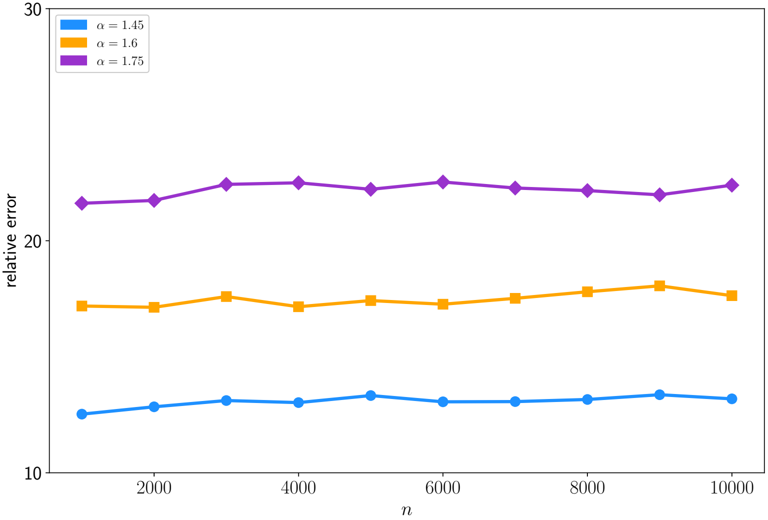


Experiment Results: Reinsurance Case

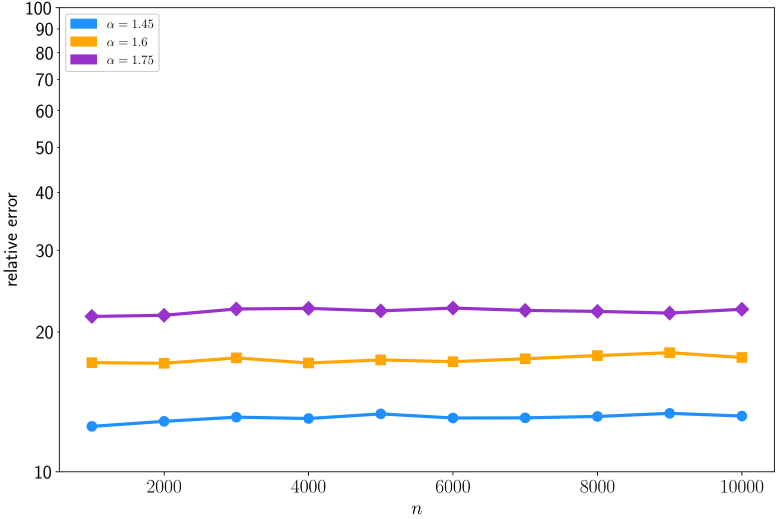
$X(t) = B(t) + \sum_{i=1}^{N(t)} W_i$; (W_i) iid Pareto-α, arrival rate 0.1;
$a = 2, b = 1.15, \gamma = 0.2, \rho = 0.95, w = 0.05.$



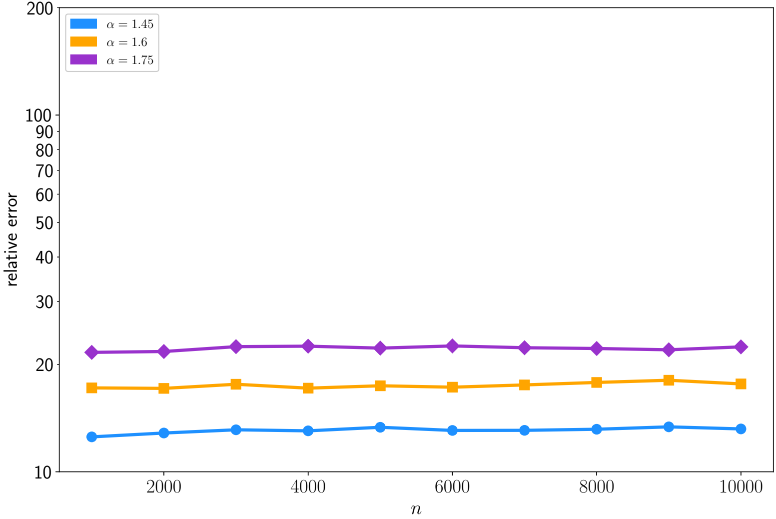
Experiment Results: Reinsurance Case



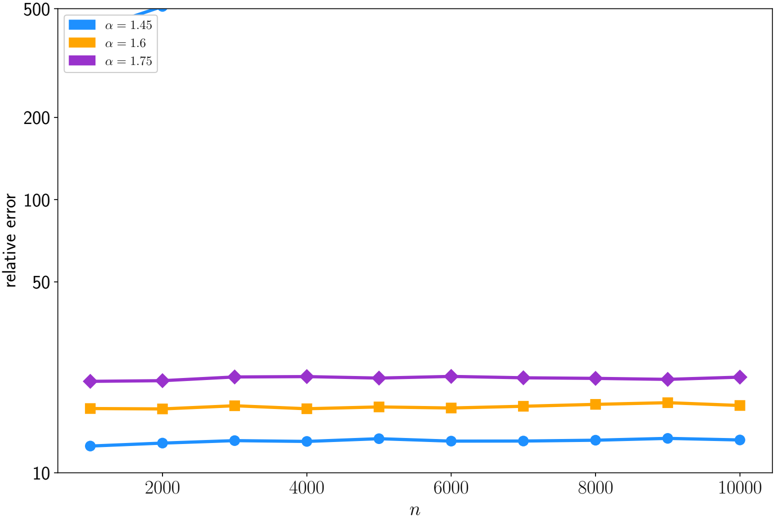
Experiment Results: Reinsurance Case



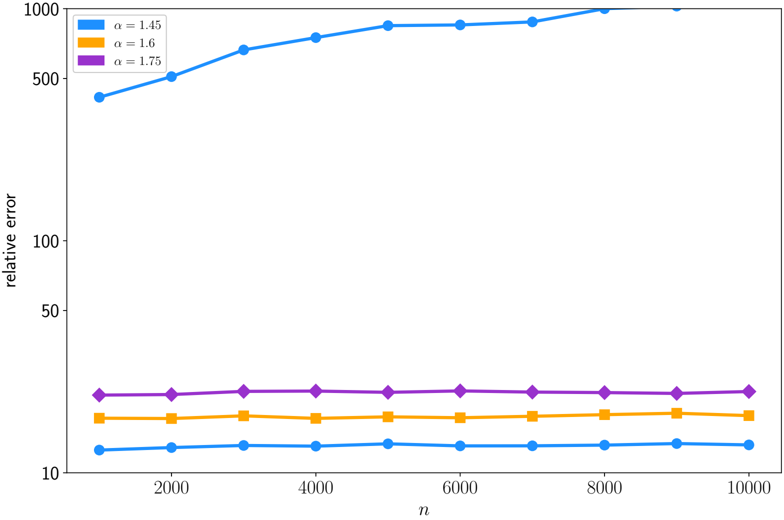
Experiment Results: Reinsurance Case



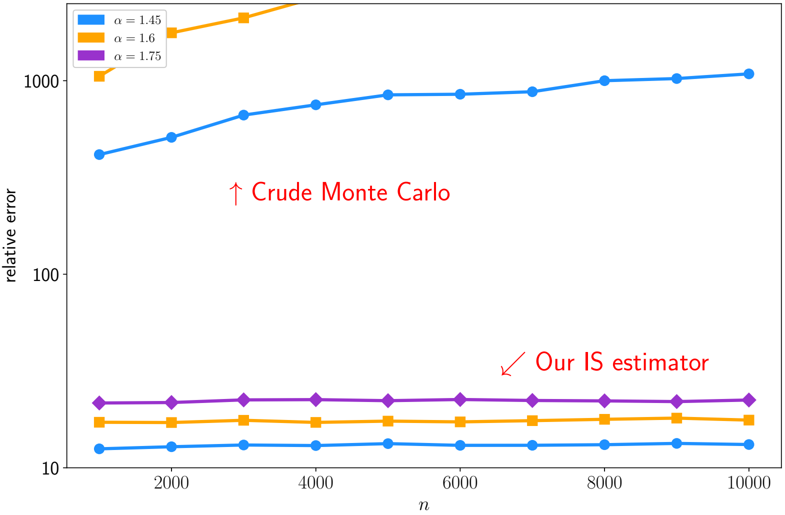
Experiment Results: Reinsurance Case



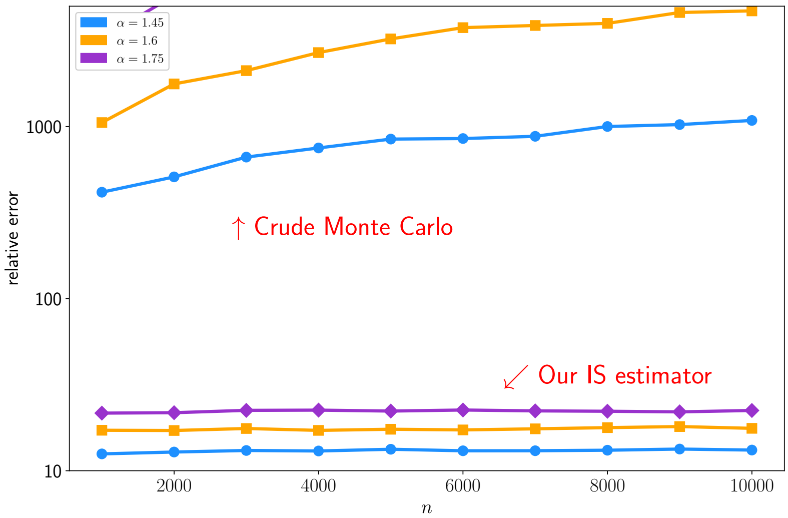
Experiment Results: Reinsurance Case



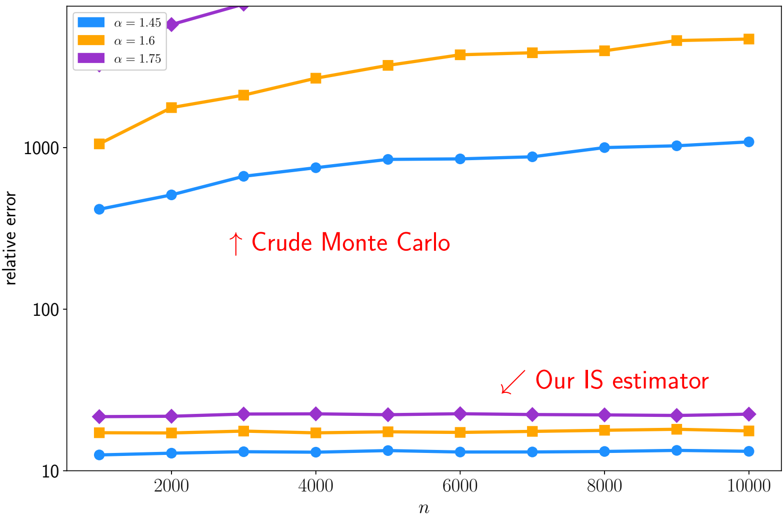
Experiment Results: Reinsurance Case



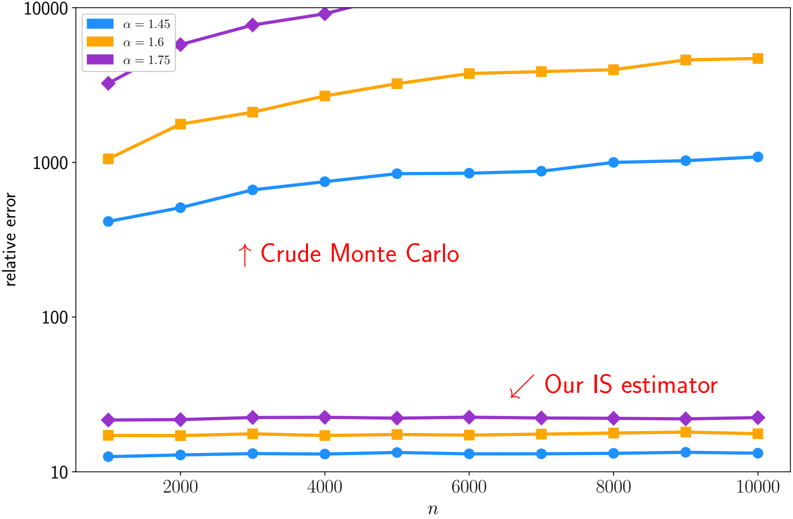
Experiment Results: Reinsurance Case



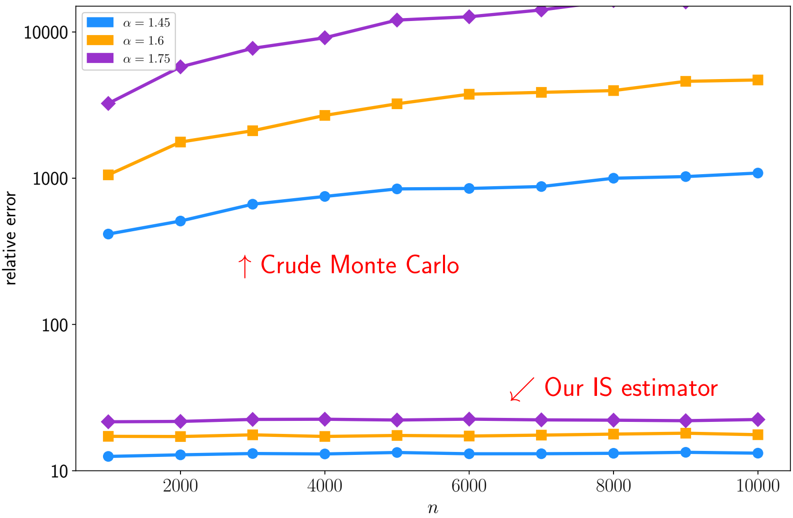
Experiment Results: Reinsurance Case



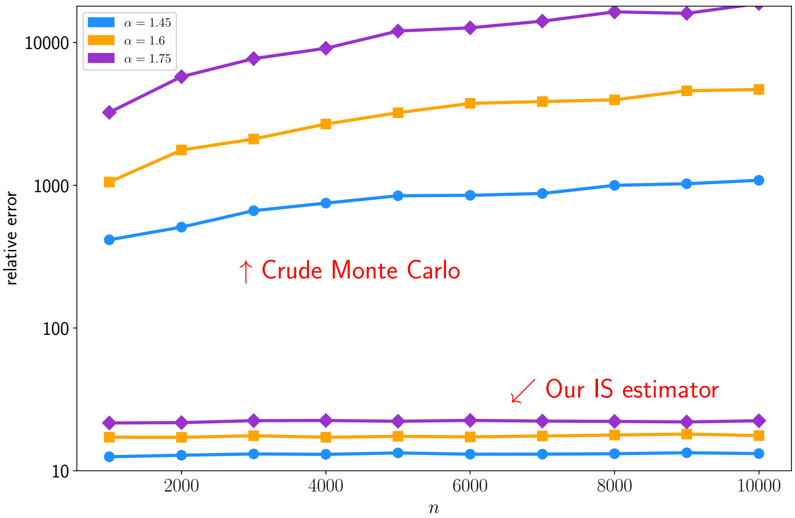
Experiment Results: Reinsurance Case



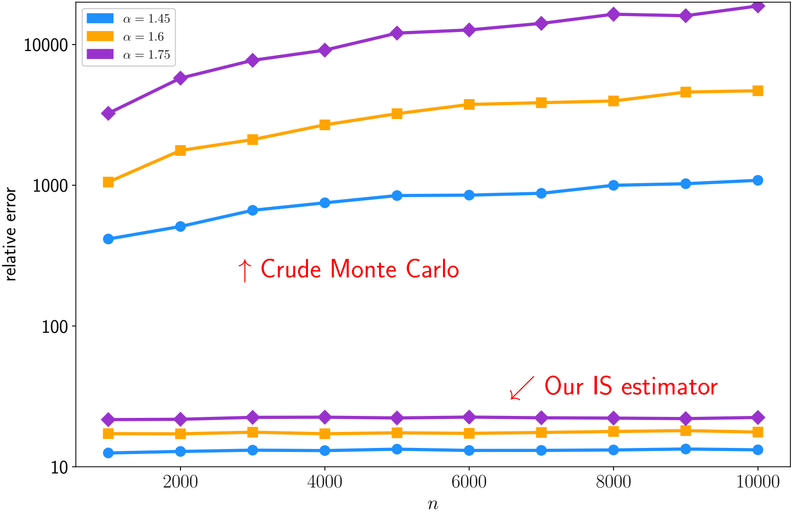
Experiment Results: Reinsurance Case



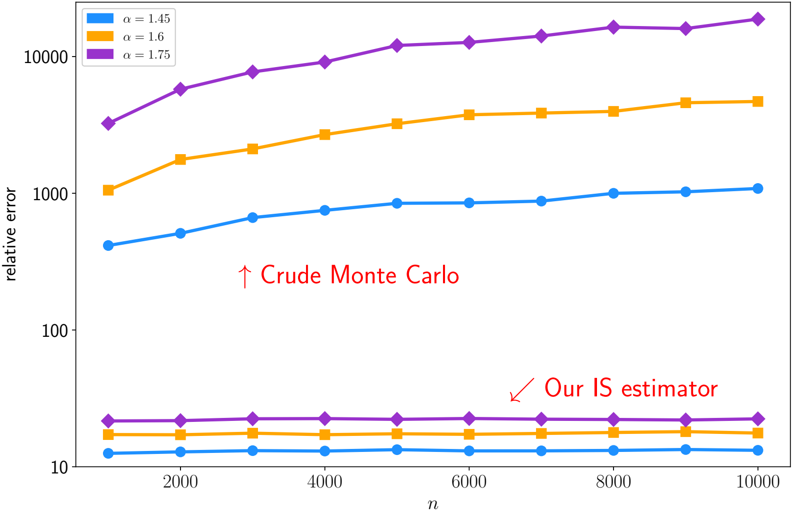
Experiment Results: Reinsurance Case



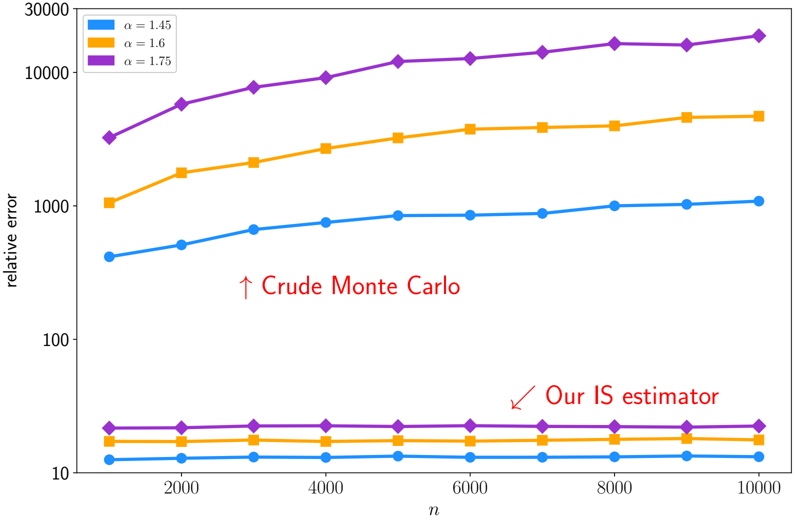
Experiment Results: Reinsurance Case



Experiment Results: Reinsurance Case



Experiment Results: Reinsurance Case



Conclusion

Proposed importance sampling algorithms

- for heavy-tailed Lévy processes with **infinite activities**
- with guarantee of **strong efficiency**
- **Significant improvements** illustrated in numerical experiments
- extension to cases where $X(t)$ is not simulatable is also available