Efficient Simulation of Tail Probabilities of Sums of Lognormal Random Variables with Gaussian Copula

Leonardo Rojas-Nandayapa
joint work with José Blanchet and Sandeep Juneja

A Conference on the Occasion of R.Y. Rubinstein’s 70th Birthday,
Outline

Introduction
  Basic Concepts
  Simulation of Tail Probabilities of Sums of Random Variables.

Main Results
  The Problem
  The Algorithms
  Examples
Outline

Introduction
   Basic Concepts
   Simulation of Tail Probabilities of Sums of Random Variables.

Main Results
   The Problem
   The Algorithms
   Examples
Outline

Introduction
  Basic Concepts
    Simulation of Tail Probabilities of Sums of Random Variables.

Main Results
  The Problem
  The Algorithms
  Examples
Estimation of rare event probabilities

Rare Event Simulation Algorithm
Let \( \{ A_x \} \) be an indexed set of events such that

\[
\lim_{x \to x_0} \mathbb{P}(A_x) \to 0.
\]

An algorithm for estimating \( \mathbb{P}(A_x) \) is a set of r.v. variables \( \{ Z_x \} \)

\[
\mathbb{E} Z_x = \mathbb{P}(A_x) \forall x.
\]

Efficient algorithms

\[
\limsup_{x \to x_0} \frac{\text{Var } Z_x}{\mathbb{P}^2 - \epsilon(A_x)} < \infty.
\]

Either for \( \epsilon = 0 \) (Bounded Relative Error) or for all \( \epsilon > 0 \) (Logarithmic Efficient).
Estimation of rare event probabilities

Rare Event Simulation Algorithm
Let \( \{A_x\} \) be an indexed set of events such that

\[
\lim_{x \to x_0} P(A_x) \to 0.
\]

An algorithm for estimating \( P(A_x) \) is a set of r.v. variables \( \{Z_x\} \)

\[
E Z_x = P(A_x) \forall x.
\]

Efficient algorithms

\[
\limsup_{x \to x_0} \frac{\text{Var} Z_x}{P^2 - \epsilon(A_x)} < \infty.
\]

Either for \( \epsilon = 0 \) (Bounded Relative Error) or for all \( \epsilon > 0 \) (Logarithmic Efficient).
Outline

Introduction
Basic Concepts
Simulation of Tail Probabilities of Sums of Random Variables.

Main Results
The Problem
The Algorithms
Examples
Rare event simulation involving sums of r.v.’s.

Rare event probabilities of the type

$$\mathbb{P}(X_1 + \ldots + X_N > u) \quad u \to \infty.$$  

$N$ possibly random.

Light tails
Most established tool is Importance Sampling.

Heavy Tails
Rare event simulation involving sums of r.v.’s.

Rare event probabilities of the type

\[ \mathbb{P}(X_1 + \ldots + X_N > u) \quad u \to \infty. \]

\( N \) possibly random.

**Light tails**
Most established tool is Importance Sampling.

**Heavy Tails**
Rare event simulation involving sums of r.v.’s.

Rare event probabilities of the type

\[ \mathbb{P}(X_1 + \ldots + X_N > u) \quad u \to \infty. \]

\( N \) possibly random.

**Light tails**

Most established tool is Importance Sampling.

**Heavy Tails**

Rare event simulation involving i.i.d. r.v.

State Independent

- Asmussen and Binswanger (1997) Logarithmic Efficiency

State-Dependent

- Dupuis et. al. (2007) IS for Regularly Varying.
Subexponential Distributions

In the independent case

$$P(X_1 + \ldots + X_n > u) \sim \sum_{i=1}^{n} P(X_i > u)$$

Intuitive Idea

$S_n$ becomes large as a consequence of single large jump.

Dependent Case

Ideas from the Independent Case might not work.
Simulation with Heavy Tailed Random Variables

Subexponential Distributions

In the independent case

\[ \mathbb{P}(X_1 + \ldots + X_n > u) \sim \sum_{i=1}^{n} \mathbb{P}(X_i > u) \]

Intuitive Idea

\( S_n \) becomes large as a consequence of single large jump.

Dependent Case

Ideas from the Independent Case might not work.
Simulation with Heavy Tailed Random Variables

Subexponential Distributions
In the independent case

\[ \mathbb{P}(X_1 + \ldots + X_n > u) \sim \sum_{i=1}^{n} \mathbb{P}(X_i > u) \]

Intuitive Idea
\( S_n \) becomes large as a consequence of single large jump.

Dependent Case
Ideas from the Independent Case might not work.
Outline

Introduction
   Basic Concepts
   Simulation of Tail Probabilities of Sums of Random Variables.

Main Results
   The Problem
   The Algorithms
   Examples
Definition
Let \((Y_1, \ldots, Y_n)\) be a multivariate Gaussian random vector. Take \(X_k = e^{Y_k}\). The vector \((X_1, \ldots, X_n)\) is a lognormal random vector with gaussian copula.

Lognormal Random Variables as Heavy Tailed
- Light among Subexponential Distributions.
- All moments exist, however it does not have mgf.
Lognormal marginals with Gaussian Copula

Definition
Let \((Y_1, \ldots, Y_n)\) be a multivariate Gaussian random vector. Take \(X_k = e^{Y_k}\). The vector \((X_1, \ldots, X_n)\) is a lognormal random vector with gaussian copula.

Lognormal Random Variables as Heavy Tailed

- Light among Subexponential Distributions.
- All moments exist, however it does not have mgf.
Lognormal marginals with Gaussian Copula

Definition
Let \((Y_1, \ldots, Y_n)\) be a multivariate Gaussian random vector. Take \(X_k = e^{Y_k}\). The vector \((X_1, \ldots, X_n)\) is a lognormal random vector with gaussian copula.

Lognormal Random Variables as Heavy Tailed

- Light among Subexponential Distributions.
- All moments exist, however it does not have mgf.
Lognormal marginals with Gaussian Copula

Definition
Let \((Y_1, \ldots, Y_n)\) be a multivariate Gaussian random vector. Take \(X_k = e^{Y_k}\). The vector \((X_1, \ldots, X_n)\) is a lognormal random vector with gaussian copula.

Lognormal Random Variables as Heavy Tailed

- Light among Subexponential Distributions.
- All moments exist, however it does not have mgf.
Objective

Construct algorithms to estimate

$$\mathbb{P}(X_1 + \ldots + X_n > u)$$

where \((X_1, \ldots, X_n) \sim \text{LN}(\mu, \Sigma)\).
Main Contributions

Algorithm A
✓ Importance Sampling.
✓ Logarithmic efficient.

Algorithm B
✓ Conditional Monte Carlo.
✓ Logarithmic efficient.

Algorithm C
✓ Algorithm A or B plus IS.
✓ Bounded Relative Error.
Main Contributions

Algorithm A
✓ Importance Sampling.
✓ Logarithmic efficient.

Algorithm B
✓ Conditional Monte Carlo.
✓ Logarithmic efficient.

Algorithm C
✓ Algorithm A or B plus IS.
✓ Bounded Relative Error.
Main Contributions

Algorithm A
✓ Importance Sampling.
✓ Logarithmic efficient.

Algorithm B
✓ Conditional Monte Carlo.
✓ Logarithmic efficient.

Algorithm C
✓ Algorithm A or B plus IS.
✓ Bounded Relative Error.
Outline

Introduction
  Basic Concepts
  Simulation of Tail Probabilities of Sums of Random Variables.

Main Results
  The Problem
  The Algorithms
  Examples
Algorithm A

IS for Estimating the Probability of an Event $A$
Increases the probability of the event $A$ while resembling the original distribution.

How to Build a Proposal
Remember that if $\tilde{X} \sim LN(\mu, \theta^2 \sigma^2)$ then

$$\mathbb{E}(X) = e^{\mu + \theta^2 \sigma^2 / 2} \quad \text{Var}(X_i) = e^{2\mu + 2\theta^2 \sigma^2} - e^{2\mu + \theta^2 \sigma^2}$$

Hence, it seems reasonable to propose as IS distribution

$$LN_n(\mu, \theta^2(u)\Sigma)$$
Algorithm A

IS for Estimating the Probability of an Event $A$

Increases the probability of the event $A$ while resembling the original distribution.

How to Build a Proposal

Remember that if $\tilde{X} \sim LN(\mu, \theta^2 \sigma^2)$ then

$$\mathbb{E}(X) = e^{\mu + \theta^2 \sigma^2 / 2}$$

$$\text{Var}(X_i) = e^{2\mu + 2\theta^2 \sigma^2} - e^{2\mu + \theta^2 \sigma^2}$$

Hence, it seems reasonable to propose as IS distribution

$$LN_n(\bar{\mu}, \theta^2(u)\Sigma)$$
Algorithm A

IS for Estimating the Probability of an Event $A$
Increases the probability of the event $A$ while resembling the original distribution.

How to Build a Proposal
Remember that if $\tilde{X} \sim LN(\mu, \theta^2 \sigma^2)$ then

$$\mathbb{E}(X) = e^{\mu + \theta^2 \sigma^2 / 2} \quad \text{Var}(X_i) = e^{2\mu + 2\theta^2 \sigma^2} - e^{2\mu + \theta^2 \sigma^2}$$

Hence, it seems reasonable to propose as IS distribution

$$LN_n(\mu, \theta^2(u) \Sigma)$$
Algorithm A

IS for Estimating the Probability of an Event $A$

Increases the probability of the event $A$ while resembling the original distribution.

How to Build a Proposal

Remember that if $\tilde{X} \sim \text{LN}(\mu, \theta^2 \sigma^2)$ then

$$\mathbb{E}(X) = e^{\mu + \theta^2 \sigma^2 / 2} \quad \text{Var}(X_i) = e^{2\mu + 2\theta^2 \sigma^2} - e^{2\mu + \theta^2 \sigma^2}$$

Hence, it seems reasonable to propose as IS distribution

$$\text{LN}_n(\bar{\mu}, \theta^2(u)\Sigma)$$
Algorithm A

Intuitive Idea
Let $\theta(u)$ grow moderately to as $u \to \infty$.

Formal Statement
Algorithm A is logarithmic efficient if and only if

$$\log \theta(u) = o(\log^2 u)$$

How to choose it?
A convenient way to choose $\theta(u)$ is as the solution of

$$e^{\mu_1 + \theta^2 \sigma^2/2} + \ldots + e^{\mu_n + \theta^2 \sigma^2/2} = u.$$
Algorithm A

Intuitive Idea
Let $\theta(u)$ grow moderately to as $u \to \infty$.

Formal Statement
Algorithm A is logarithmic efficient if and only if

$$\log \theta(u) = o\left( \log^2 u \right)$$

How to choose it?
A convenient way to choose $\theta(u)$ is as the solution of

$$e^{\mu_1 + \theta^2 \sigma^2 / 2} + \ldots + e^{\mu_n + \theta^2 \sigma^2 / 2} = u.$$
Algorithm A

Intuitive Idea
Let $\theta(u)$ grow moderately to as $u \to \infty$.

Formal Statement
Algorithm A is logarithmic efficient if and only if

$$\log \theta(u) = o(\log^2 u)$$

How to choose it?
A convenient way to choose $\theta(u)$ is as the solution of

$$e^{\mu_1 + \theta^2 \sigma^2/2} + \ldots + e^{\mu_n + \theta^2 \sigma^2/2} = u.$$
Algorithm B

Conditional Monte Carlo
Use all known information. Simulate less.

Key Ideas
Let $Z_1, \ldots, Z_n$ i.i.d. $N(0, 1)$ r.v.’s and define

$$\tilde{Z}_i := \frac{Z_i}{\sqrt{Z_1^2 + \ldots + Z_n^2}}$$

We know $Z_1^2 + \ldots + Z_n^2 \sim \chi_n^2$ and how to simulate $\tilde{Z}_i$’s.
Algorithm B

Conditional Monte Carlo
Use all known information. Simulate less.

Key Ideas
Let $Z_1, \ldots, Z_n$ i.i.d. $N(0, 1)$ r.v.'s and define

$$\hat{Z}_i := \frac{Z_i}{\sqrt{Z_1^2 + \ldots + Z_n^2}}$$

We know $Z_1^2 + \ldots + Z_n^2 \sim \chi_n^2$ and how to simulate $\hat{Z}_i$'s.
Algorithm B

Conditional Monte Carlo
Use all known information. Simulate less.

Key Ideas
Let $Z_1, \ldots, Z_n$ i.i.d. $N(0, 1)$ r.v.’s and define

$$\tilde{Z}_i := \frac{Z_i}{\sqrt{Z_1^2 + \ldots + Z_n^2}}$$

We know $Z_1^2 + \ldots + Z_n^2 \sim \chi_n^2$ and how to simulate $\tilde{Z}_i$’s.
Algorithm B

Key Ideas
Take $R^2 \sim \chi^2_n$ and a decomposition $\Sigma = CC^*$ such that $C$ is square.

$$(Y_1, \ldots, Y_n) := R C (\tilde{Z}_1, \ldots, \tilde{Z}_n)^t + \bar{\mu} \sim N(\bar{\mu}, \Sigma)$$

CMC Algorithm

$$\mathbb{P}(S_n > u) = \mathbb{P}(e^{Y_1} + \ldots + e^{Y_n} > u)$$

$$= \mathbb{E}\left[ \mathbb{P}(e^{R(C_{11}\tilde{Z}_1 + \ldots + C_{1n}\tilde{Z}_n) + \mu_1} + \ldots + e^{R(C_{n1}\tilde{Z}_1 + \ldots + C_{nn}\tilde{Z}_n) + \mu_n} > u | \tilde{Z}_1, \ldots, \tilde{Z}_n) \right]$$
Algorithm B

Key Ideas

Take $R^2 \sim \chi^2_n$ and a decomposition $\Sigma = CC^*$ such that $C$ is square.

$$(Y_1, \ldots, Y_n) := R C (\tilde{Z}_1, \ldots, \tilde{Z}_n)^t + \mu \sim N(\mu, \Sigma)$$

CMC Algorithm

$$\mathbb{P}(S_n > u) = \mathbb{P}(e^{Y_1} + \ldots + e^{Y_n} > u)$$
$$= \mathbb{E} \left[ \mathbb{P} \left( e^{R(C_{11}\tilde{Z}_1 + \ldots + C_{1n}\tilde{Z}_n) + \mu_1} + \ldots ight. \right.$$  
$$\left. + e^{R(C_{n1}\tilde{Z}_1 + \ldots + C_{nn}\tilde{Z}_n) + \mu_n} > u | \tilde{Z}_1, \ldots, \tilde{Z}_n \right) \right]$$
Algorithm B

CMC Algorithm
Simulate $\tilde{Z} = (\tilde{Z}_1, \ldots, \tilde{Z}_n)$ and return

$$P(R < \psi_1(u, \tilde{Z})) + P(R > \psi_2(u, \tilde{Z}))$$

Efficiency
The algorithm B has logarithmic efficiency.
Algorithm B

CMC Algorithm
Simulate $\tilde{Z} = (\tilde{Z}_1, \ldots, \tilde{Z}_n)$ and return

$$\mathbb{P}(R < \psi_1(u, \tilde{Z})) + \mathbb{P}(R > \psi_2(u, \tilde{Z}))$$

Efficiency
The algorithm B has logarithmic efficiency.
Preliminaries of Algorithm C

Asymptotic Result
A consequence of Asmussen and Rojas-Nandayapa (2008)

\[ P(S_n > u) \sim P(\max\{X_i : i = 1, \ldots, n\} > u) \sim \sum_{i=1}^{n} P(X_i > u) \]

in the Dependent Case.

Intuitive Ideas
- Asymptotically \( P(M_n > u) \) accounts for most of the total probability \( P(S_n > u) \).
- In the event \( \{M_n > u\} \) the random variables \( X_1, \ldots, X_n \) behave as independent random variables.
Preliminaries of Algorithm C

Asymptotic Result
A consequence of Asmussen and Rojas-Nandayapa (2008)

\[ \mathbb{P}(S_n > u) \sim \mathbb{P}(\max\{X_i : i = 1, \ldots, n\} > u) \sim \sum_{i=1}^{n} \mathbb{P}(X_i > u) \]

in the Dependent Case.

Intuitive Ideas

- Asymptotically \( \mathbb{P}(M_n > u) \) accounts for most of the total probability \( \mathbb{P}(S_n > u) \).
- In the event \( \{M_n > u\} \) the random variables \( X_1, \ldots, X_n \) behave as independent random variables.
Asymptotic Result

A consequence of Asmussen and Rojas-Nandayapa (2008)

\[ P(S_n > u) \sim P(\max\{X_i : i = 1, \ldots, n\} > u) \sim \sum_{i=1}^{n} P(X_i > u) \]

in the Dependent Case.

Intuitive Ideas

▶ Asymptotically \( P(M_n > u) \) accounts for most of the total probability \( P(S_n > u) \).

▶ In the event \( \{M_n > u\} \) the random variables \( X_1, \ldots, X_n \) behave as independent random variables.
Asymptotic Result
A consequence of Asmussen and Rojas-Nandayapa (2008)

\[ \mathbb{P}(S_n > u) \sim \mathbb{P}(\max\{X_i : i = 1, \ldots, n\} > u) \sim \sum_{i=1}^{n} \mathbb{P}(X_i > u) \]

in the Dependent Case.

Intuitive Ideas
- Asymptotically \( \mathbb{P}(M_n > u) \) accounts for most of the total probability \( \mathbb{P}(S_n > u) \).
- In the event \( \{M_n > u\} \) the random variables \( X_1, \ldots, X_n \) behave as independent random variables.
Preliminaries of Algorithm C

Key Idea
First proposed in Juneja (2008)

\[ P(S_n > u) = P(S_n > u, M_n < u) + P(M_n > u) \]

Estimation of \( P(S_n > u, M_n < u) \)
The same as in Algorithms A and B. Smaller variance.

Estimation of \( P(M_n > u) \)
Design a new method for the Gaussian Copula (IS).
Preliminaries of Algorithm C

Key Idea
First proposed in Juneja (2008)

\[ P(S_n > u) = P(S_n > u, M_n < u) + P(M_n > u) \]

Estimation of \( P(S_n > u, M_n < u) \)
The same as in Algorithms A and B. Smaller variance.

Estimation of \( P(M_n > u) \)
Design a new method for the Gaussian Copula (IS).
Preliminaries of Algorithm C

Key Idea
First proposed in Juneja (2008)

\[ P(S_n > u) = P(S_n > u, M_n < u) + P(M_n > u) \]

Estimation of \( P(S_n > u, M_n < u) \)
The same as in Algorithms A and B. Smaller variance.

Estimation of \( P(M_n > u) \)
Design a new method for the Gaussian Copula (IS).
Estimation of $\mathbb{P}(M_n > u)$

Importance Sampling Distribution
Take $K$ supported over $\{1, \ldots, n\}$. Consider the distribution of

$$(X_1, \ldots, X_n|X_K > u)$$

Main Features
✓ We know how to simulate it.
✓ We know its density.
✓ It is supported exactly over $\{M_n > u\}$.

Distribution of $K$
Our proposal

$$
P(K = k) = \frac{\mathbb{P}(X_k > u)}{\sum_{\ell=1}^{n} \mathbb{P}(X_{\ell} > u)}.
$$
Estimation of $\mathbb{P}(M_n > u)$

Importance Sampling Distribution
Take $K$ supported over $\{1, \ldots, n\}$. Consider the distribution of

$$(X_1, \ldots, X_n|X_K > u)$$

Main Features
✓ We know how to simulate it.
✓ We know its density.
✓ It is supported exactly over $\{M_n > u\}$.

Distribution of $K$
Our proposal

$$
\mathbb{P}(K = k) = \frac{\mathbb{P}(X_k > u)}{\sum_{\ell=1}^{n} \mathbb{P}(X_\ell > u)}.
$$
Estimation of $\mathbb{P}(M_n > u)$

Importance Sampling Distribution
Take $K$ supported over $\{1, \ldots, n\}$. Consider the distribution of

$$(X_1, \ldots, X_n | X_K > u)$$

Main Features
✓ We know how to simulate it.
✓ We know its density.
✓ It is supported exactly over $\{M_n > u\}$.

Distribution of $K$
Our proposal

$$\mathbb{P}(K = k) = \frac{\mathbb{P}(X_k > u)}{\sum_{\ell=1}^{n} \mathbb{P}(X_{\ell} > u)}.$$
Algorithm C

Efficiency
The following algorithms has Bounded Relative Error

- The IS algorithm for $\mathbb{P}(M_n > u)$.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm A.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm B.
Algorithm C

Efficiency

The following algorithms has Bounded Relative Error

- The IS algorithm for $\mathbb{P}(M_n > u)$.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm A.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm B.
Algorithm C

Efficiency

The following algorithms have Bounded Relative Error

- The IS algorithm for $\mathbb{P}(M_n > u)$.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm A.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm B.
Algorithm C

Efficiency
The following algorithms has Bounded Relative Error

- The IS algorithm for $\mathbb{P}(M_n > u)$.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm A.
- Algorithm C for $\mathbb{P}(S_n > u)$ based on Algorithm B.
Outline

Introduction
Basic Concepts
Simulation of Tail Probabilities of Sums of Random Variables.

Main Results
The Problem
The Algorithms
Examples
Example 1

Example

10 lognormal r.v. with Gaussian Copula

- $\mu_i = i - 10$
- $\sigma_i^2 = i$
- $\sigma_{ij} = 0.4\sigma_i\sigma_j$
- $R = 10000$
  (Estimator)
- $R = 1000000$
Example 1

Comparison of Variances

- AK estimator
- Algorithm A
- Algorithm C

Comparison of Relative Errors

- AK estimator
- Algorithm A
- Algorithm C