



Exact Simulation of Poisson-Dirichlet Distribution and Generalised Gamma Process

Angelos Dassios¹ · Junyi Zhang²

Received: 13 February 2022 / Revised: 1 May 2023 / Accepted: 4 May 2023 /
Published online: 10 June 2023
© The Author(s) 2023

Abstract

Let $J_1 > J_2 > \dots$ be the ranked jumps of a gamma process τ_α on the time interval $[0, \alpha]$, such that $\tau_\alpha = \sum_{k=1}^{\infty} J_k$. In this paper, we design an algorithm that samples from the random vector $(J_1, \dots, J_N, \sum_{k=N+1}^{\infty} J_k)$. Our algorithm provides an analog to the well-established inverse Lévy measure (ILM) algorithm by replacing the numerical inversion of exponential integral with an acceptance-rejection step. This research is motivated by the construction of Dirichlet process prior in Bayesian nonparametric statistics. The prior assigns weight to each atom according to a GEM distribution, and the simulation algorithm enables us to sample from the N largest random weights of the prior. Then we extend the simulation algorithm to a generalised gamma process. The simulation problem of inhomogeneous processes will also be considered. Numerical implementations are provided to illustrate the effectiveness of our algorithms.

Keywords Exact simulation · Gamma process · Generalised gamma process · Lévy process · Poisson-Dirichlet distribution

MSC Classification 62F15 · 62G05 · 60J25

1 Introduction

Consider a subordinator $\tau_{\alpha,\sigma}$ with the Lévy-Khintchine representation

Angelos Dassios and Junyi Zhang has contributed equally to this work

✉ Junyi Zhang
JunyiZhang@polyu.edu.hk
Angelos Dassios
A.Dassios@lse.ac.uk

¹ Department of Statistics, London School of Economics, Houghton Street, London WC2A 2AE, United Kingdom

² Department of Applied Mathematics, The Hong Kong Polytechnic University, Kowloon, Hong Kong

$$\mathbb{E}(\exp(-\beta\tau_{\alpha,\sigma})) = \exp\left(-\alpha \int_0^\infty (1 - e^{-\beta w})w^{-\sigma-1}e^{-w}dw\right).$$

When $\alpha > 0$ and $\sigma = 0$, the subordinator $\tau_\alpha := \tau_{\alpha,0}$ is a gamma process; while for $\alpha > 0$ and $0 < \sigma < 1$, $\tau_{\alpha,\sigma}$ is a generalised gamma process. From the basic properties of Poisson random measure (see, for example, Section 2.2 of Kyprianou 2006) we know both processes are pure-jump with infinite activity. Thus, they can be written as $\tau_{\alpha,\sigma} = \sum_{k=1}^\infty J_k$, where $J_1 > J_2 > \dots > 0$ are the ranked jumps of $\tau_{\alpha,\sigma}$ on the time interval $[0, \alpha]$. In this paper, we design an acceptance-rejection algorithm that samples from the N largest jumps, J_1, \dots, J_N , and the sum of the smaller jumps, $\sum_{k=N+1}^\infty J_k$, of the subordinator simultaneously.

This research is motivated by the simulation problem of a Poisson-Dirichlet distribution, which is a random probability measure on the set of decreasing positive sequences with sum 1. The definition is given as follows.

Definition 1 (Poisson-Dirichlet distribution) For $\alpha > 0$, suppose that independent random variables V_i are such that V_i has $Beta(1, \alpha)$ distribution. Let

$$\tilde{\pi}_1 = V_1, \quad \tilde{\pi}_k = V_k \prod_{i=1}^{k-1} (1 - V_i), \quad \text{for } k = 2, 3, \dots$$

Define the Griffiths-Engen-McCloskey distribution with parameter α , abbreviated $GEM(\alpha)$ to be the resultant distribution $(\tilde{\pi}_1, \tilde{\pi}_2, \dots)$. Moreover, define (π_1, π_2, \dots) to be their sorted values such that $\pi_1 > \pi_2 > \dots$. Then (π_1, π_2, \dots) follows the Poisson-Dirichlet distribution with parameter α , abbreviated $PD(\alpha)$.

Results of Perman et al. (1992) show that the sequence $\{\tilde{\pi}_k\}_{k=1}^\infty$ is a size-biased permutation of $\{\pi_k\}_{k=1}^\infty$, i.e., the same sequence presented in a random order $(\pi_{\sigma_1}, \pi_{\sigma_2}, \dots)$, where $\mathbb{P}(\sigma_1 = i) = \pi_i$, and for k distinct i_1, \dots, i_k ,

$$\mathbb{P}(\sigma_k = i_k \mid \sigma_1 = i_1, \dots, \sigma_{k-1} = i_{k-1}) = \frac{\pi_{i_k}}{1 - (\pi_{i_1} + \dots + \pi_{i_{k-1}})}.$$

An index i with bigger ‘size’ π_i tends to appear earlier in the permutation, hence the name size-biased. On the other hand, we revert to the $PD(\alpha)$ distribution by sorting the random weights of $GEM(\alpha)$ in descending order.

The application of $PD(\alpha)$ distribution appears in many fields, for example, to study the asymptotic distribution of the prime factorization of a randomly drawn integer in number theory, see Donnelly and Grimmett (1993), Tenenbaum (1995), Billingsley (1972) and Vershik (1986); to model the asymptotic distribution of the ranked relative cycle lengths in a random permutation in combinatorics, see Arratia et al. (2003), Aldous (1985), Hansen (1994) and Schmidt and Vershik (1977); to describe the limiting distribution of several sequences of discrete stochastic models of the infinitely-many-neutral-alleles type in population genetics, see Ewens (1990), Watterson (1976) and Kingman (1980); and in particular, to construct the Dirichlet process prior in Bayesian nonparametric statistics, see Ferguson (1973). As we will see in Sect. 2, the construction of Bayesian nonparametric priors also leads us to the simulation problem of a generalised gamma process.

Despite its huge variety of applications, the exact simulation algorithm for $PD(\alpha)$ is less frequently examined. Wolpert and Ickstadt (1998b) introduced the inverse Lévy measure (ILM) algorithm that samples from the largest jumps of a gamma process and use them to approximate the Ferguson-Klass representation (Ferguson and Klass 1972) of a Poisson-Dirichlet distribution. Zarepour and Labadi (2012) suggested to approximate the distribution of the largest jumps by a gamma density. Due to the difficulty in simulation, these methods

ignore the infinite number of smaller jumps of the gamma process in the Ferguson-Klass representation. However, a recent research Dassios et al. (2019) proposed an exact simulation algorithm for the truncated gamma process. The algorithm can be used to sample from the sum of the smaller jumps. In this paper, we will combine these methods together and design an exact simulation algorithm for the N largest components of a Poisson-Dirichlet distribution.

In the meanwhile, a trivial simulation algorithm for $PD(\alpha)$ is immediately obtained from Definition 1.

Theorem 1 (Trivial algorithm) *This algorithm generates an approximation for the N largest components of a $PD(\alpha)$ distribution.*

1. Initialise α and N , select a positive integer $m \gg N$ (for example $m = 10N$).
2. For each $i = 1, \dots, m$, generate a beta random number $V_i \leftarrow \text{Beta}(1, \alpha)$.
3. For each $i = 1, \dots, m$, set $\tilde{\pi}_i \leftarrow (1 - V_1) \dots (1 - V_{i-1})V_i$.
4. Sort $\{\tilde{\pi}_i\}_{i=1, \dots, m}$ in descending order and let $\pi_1 > \pi_2 > \dots > \pi_m$ be the ranked values of $\{\tilde{\pi}_i\}_{i=1, \dots, m}$.
5. Truncate the sequence $\{\pi_i\}_{i=1, \dots, m}$ at the first N components, output $(\pi_1, \pi_2, \dots, \pi_N)$.

In Step 2 and 3 of Algorithm 1, we follow the stick-breaking construction of the GEM distribution and sample from its first m components. As $m \rightarrow \infty$, these components become an exact sample of the GEM distribution, and Step 4 returns an exact sample of the Poisson-Dirichlet distribution. However, it is impossible to take m to be infinity in practice, so Algorithm 1 is only an approximation for the $PD(\alpha)$ distribution.

The rest of the paper is organised as follows. In Sect. 2, we provide some preliminary results for our simulation algorithms and explain their potential applications in Bayesian nonparametric statistics. In Sect. 3, we derive the joint density of the N largest jumps of the gamma and generalised gamma processes, then develop the acceptance-rejection algorithm that samples from these jumps. Section 4 presents the numerical implementations of these algorithms. Section 5 gives some concluding remarks.

2 Preliminary Results

In this section, we illustrate the connection between the (generalised) gamma process and Bayesian nonparametric statistics, and provide some preliminary results that will be used later. The concept of Dirichlet process was first introduced by Ferguson (1973); since then it turns out to be a most notable prior in Bayesian nonparametric statistics. From Theorem 2 of Ferguson (1973), we know that a Dirichlet process can be written as

$$G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k},$$

where $\{\pi_k\}_{k=1}^{\infty}$ follows the $PD(\alpha)$ distribution, $\{\phi_k\}_{k=1}^{\infty}$ is a sequence of i.i.d. random variables with the common distribution G_0 , and δ_{ϕ_k} is a point mass at ϕ_k . In Bayesian nonparametric statistics, the latent parameter ϕ_k is randomly chosen with probability π_k , i.e., according to the $PD(\alpha)$ distribution.

On the other hand, let $\{J_k\}_{k=1}^{\infty}$ be the ranked jumps of a gamma process τ_α on the time interval $[0, \alpha]$. From Section 4 of Ferguson (1973), we know that the normalised jumps of τ_α has the same law as the Poisson-Dirichlet distribution, i.e.,

$$(J_1/\tau_\alpha, J_2/\tau_\alpha, J_3/\tau_\alpha, \dots) \stackrel{law}{=} (\pi_1, \pi_2, \pi_3, \dots).$$

Then we deduce that to sample from the N largest components of the $PD(\alpha)$ distribution, it is sufficient to simulate the random vector $(J_1, \dots, J_N, \tau_\alpha)$, and that is exactly the target of our algorithm.

To simplify the procedure, we define a partial sum $\tau_\alpha^{N-} := \sum_{k=N+1}^\infty J_k$, then our task is equivalent to sampling from $(J_1, \dots, J_N, \tau_\alpha^{N-})$ because we can revert to τ_α easily by calculating $\sum_{k=1}^N J_k + \tau_\alpha^{N-}$. But the partial sum is extremely useful because once we have sampled from (J_1, \dots, J_N) , the conditional Lévy-Khintchine representation of τ_α^{N-} is

$$\mathbb{E} \left(\exp(-\beta \tau_\alpha^{N-}) \mid J_1, \dots, J_N \right) = \exp \left(-\alpha \int_0^{J_N} (1 - e^{-\beta w}) w^{-1} e^{-w} dw \right).$$

It follows that τ_α^{N-} is a truncated gamma process, and its exact simulation algorithm has been developed in Dassios et al. (2019). Finally, we can derive the N largest components and the sum of smaller components of $PD(\alpha)$ from

$$\pi_i \stackrel{law}{=} \frac{J_i}{\sum_{k=1}^N J_k + \tau_\alpha^{N-}}, i = 1, \dots, N, \text{ and } \sum_{k=N+1}^\infty \pi_k \stackrel{law}{=} \frac{\tau_\alpha^{N-}}{\sum_{k=1}^N J_k + \tau_\alpha^{N-}}. \quad (1)$$

Another popular prior in Bayesian nonparametric statistics is the generalised Dirichlet process prior. It was first introduced by Hougaard (1986) and Brix (1999), and further studied in Epifani et al. (2003), James et al. (2005), Lijoi et al. (2007) and Ayed et al. (2019). Let $\{J_k\}_{k=1}^\infty$ be the ranked jumps of a generalised gamma process $\tau_{\alpha,\sigma}$ on the time interval $[0, \alpha]$, then we can express the normalised jump sizes of $\tau_{\alpha,\sigma}$ as $(J_1/\tau_{\alpha,\sigma}, J_2/\tau_{\alpha,\sigma}, J_3/\tau_{\alpha,\sigma}, \dots)$. This is again a random probability measure on the set of decreasing positive sequences with sum 1. We denote by $p_k := J_k/\tau_{\alpha,\sigma}$ the k -th component of this distribution, then the generalised Dirichlet process prior can be written as $G = \sum_{k=1}^\infty p_k \delta_{\phi_k}$.

Our target is to sample from the N largest random weights, (p_1, \dots, p_N) , of the generalised Dirichlet process prior. To this end, we first sample from the N largest jumps (J_1, \dots, J_N) of the generalised gamma process, then simulate the partial sum $\tau_{\alpha,\sigma}^{N-} := \sum_{k=N+1}^\infty J_k$, whose conditional Lévy-Khintchine representation is

$$\mathbb{E} \left(\exp(-\beta \tau_{\alpha,\sigma}^{N-}) \mid J_1, \dots, J_N \right) = \exp \left(-\alpha \int_0^{J_N} (1 - e^{-\beta w}) w^{-\sigma-1} e^{-w} dw \right).$$

This turns out to be the Lévy-Khintchine representation of a truncated tempered stable process, and the exact simulation algorithm can be found in Dassios et al. (2020). Finally, we can use the decomposition (1) to derive the N largest components and the sum of smaller components of the generalised Dirichlet process.

3 Simulation Algorithms

In this section, we review the inverse Lévy measure (ILM) algorithm for the gamma and generalised processes, then introduce the acceptance-rejection (AR) algorithm for these processes. We will also extend these algorithms to inhomogeneous processes.

3.1 ILM Algorithm

The ILM algorithm is a straightforward method that samples from the ranked jumps of a Lévy process. We briefly explain the key ideas here and refer the readers to Wolpert and Ickstadt (1998a) and Wolpert and Ickstadt (1998b) for more details. Consider the largest jump J_1 of a generalised gamma process $\tau_{\alpha,\sigma}$. From the basic properties of Poisson random measure, we know

$$\begin{aligned} \mathbb{P}(J_1 \leq x_1) &= \mathbb{P}(\text{Pois}(\alpha v(x_1, \infty)) = 0) \\ &= \exp(-\alpha v(x_1, \infty)) = \exp\left(-\alpha \int_{x_1}^{\infty} w^{-\sigma-1} e^{-w} dw\right), \end{aligned} \tag{2}$$

where $v(\cdot)$ denotes the Lévy measure. Then we can sample from J_1 via the inverse transform method. Let $U_1 \sim U(0, 1)$ be an uniform random number, then the solution to the equation $U_1 = \exp(-\alpha \int_{x_1}^{\infty} w^{-\sigma-1} e^{-w} dw)$ provides a sample of J_1 . Conditioning on J_{k-1} , we can sample from J_k iteratively by setting $U_k \sim U(0, 1)$ and solving the equation $U_k = \exp(-\alpha \int_{x_k}^{J_{k-1}} w^{-\sigma-1} e^{-w} dw)$, for $x_k \in (0, J_{k-1})$. This method is easy to implement, but since the explicit inversion of incomplete gamma integral is unknown, a numerical inversion is required in every iteration. Note that by taking $\sigma = 0$, we get the ILM algorithm for a gamma process, and the numerical inversion of an exponential integral is required.

3.2 AR Algorithm for Gamma Process

Next, we derive the joint density of the N largest jumps of a gamma process and develop an acceptance-rejection algorithm to sample from these jumps. Since the size of the $(k + 1)$ -th largest jump depends on the size of the k -th largest jump, i.e., $J_{k+1} \in (0, J_k)$, it is more convenient to work on the ratio between two consecutive jumps. To this end, we define $R_k := J_{k+1}/J_k, k = 1, 2, \dots$, and study the joint density of $(J_1, R_1, \dots, R_{N-1})$.

Theorem 2 *Denote by J_1, \dots, J_N the N largest jumps of a gamma process τ_α . Let $R_k := J_{k+1}/J_k$ be the ratio between the $(k + 1)$ -th and k -th largest jumps, then the joint density of $(J_1, R_1, \dots, R_{N-1})$ is*

$$\begin{aligned} &\mathbb{P}(J_1 \in dx_1, R_1 \in dr_1, \dots, R_{N-1} \in dr_{N-1}) \\ &= \exp\left(-\alpha \int_{x_1 r_1 \dots r_{N-1}}^{\infty} w^{-1} e^{-w} dw\right) \alpha^N (x_1 r_1 \dots r_{N-1})^{-1} \\ &\quad \times e^{-x_1(1+r_1+\dots+r_1 r_2 \dots r_{N-1})} dr_{N-1} \dots dr_1 dx_1, \end{aligned}$$

where $x_1 \in (0, \infty)$ and $r_k \in (0, 1)$, for $k = 1, \dots, N - 1$.

Proof It follows from Eq. (2) that J_1 has the density

$$\mathbb{P}(J_1 \in dx_1) = \exp\left(-\alpha \int_{x_1}^{\infty} w^{-1} e^{-w} dw\right) \alpha x_1^{-1} e^{-x_1} dx_1,$$

where $x_1 \in (0, \infty)$. The same argument gives that

$$\mathbb{P}(J_k \in dx_k \mid J_{k-1} = x_{k-1}) = \exp\left(-\alpha \int_{x_k}^{x_{k-1}} w^{-1} e^{-w} dw\right) \alpha x_k^{-1} e^{-x_k} dx_k,$$

where $x_k \in (0, x_{k-1})$. Then we obtain the joint density of (J_1, \dots, J_N) in terms of the product of the density functions above,

$$\begin{aligned} &\mathbb{P}(J_1 \in dx_1, \dots, J_N \in dx_N) \\ &= \exp\left(-\alpha \int_{x_N}^{\infty} w^{-1} e^{-w} dw\right) \alpha^N (x_1 \dots x_N)^{-1} e^{-(x_1 + \dots + x_N)} dx_N \dots dx_1, \end{aligned}$$

where $x_1 > x_2 > \dots > x_N > 0$. Finally, we make the change of variables $x_k = x_1 \prod_{i=1}^{k-1} r_i$, for $k = 2, \dots, N$, and the theorem is proved. \square

Next, we design an acceptance-rejection algorithm to sample from the random vector $(J_1, \dots, J_N, \tau_\alpha^{N-})$. It is Algorithm 3.2 of Dassios et al. (2019) which we refer to as *AlgorithmTG*(\cdot, \cdot).

Theorem 3 *The acceptance-rejection algorithm for $(J_1, \dots, J_N, \tau_\alpha^{N-})$.*

1. Numerically maximising

$$\frac{\exp(-\alpha \int_{x_1 r_1 \dots r_{N-1}}^{\infty} w^{-1} e^{-w} dw) e^{-x_1(1+r_1+\dots+r_1 r_2 \dots r_{N-1})}}{\frac{1}{\Gamma(\alpha)0.99^\alpha} x_1^\alpha e^{-(x_1/0.99)} \alpha^{-1} (r_1 \dots r_{N-1})^\alpha}$$

with respect to $x_1 \in (0, \infty)$ and $r_k \in (0, 1)$, for $k = 1, \dots, N - 1$. Record the maximum value $C_{\alpha, N}$.

- 2. Sample from a gamma distribution $X_1 \leftarrow \text{Gamma}(\alpha, 0.99)$, and for each $k = 1, \dots, N - 1$, sample from an uniform distribution $U_k \leftarrow U(0, 1)$, then set $R_k \leftarrow U_k^{1/\alpha}$.
- 3. Sample from an uniform distribution $U \leftarrow U(0, 1)$, if

$$U \leq \frac{\exp(-\alpha \int_{X_1 R_1 \dots R_{N-1}}^{\infty} w^{-1} e^{-w} dw) e^{-X_1(1+R_1+\dots+R_1 R_2 \dots R_{N-1})}}{C \frac{1}{\Gamma(\alpha)0.99^\alpha} X_1^\alpha e^{-(X_1/0.99)} \alpha^{-1} (R_1 \dots R_{N-1})^\alpha},$$

accept the candidates $(X_1, R_1, \dots, R_{N-1})$ and go to Step 4; otherwise, go to Step 2.

- 4. Set $J_1 \leftarrow X_1$, and for each $k = 2, \dots, N$, set $J_k \leftarrow J_1 \prod_{i=1}^{k-1} R_i$.
- 5. Set $\tau_\alpha^{N-} \leftarrow J_N \times \text{AlgorithmTG}(\alpha, J_N)$.
- 6. Output $(J_1, \dots, J_N, \tau_\alpha^{N-})$.

Proof We apply the acceptance-rejection method to sample from the joint density of $(J_1, R_1, \dots, R_{N-1})$ in Theorem 2 with the envelope

$$\begin{aligned} &\frac{1}{\Gamma(\alpha)b^\alpha} x_1^{\alpha-1} e^{-x_1/b} \exp\left(-\alpha \int_{x_1 r_1 \dots r_{N-1}}^{x_1} w^{-1} dw\right) \alpha^{N-1} (r_1 \dots r_{N-1})^{-1} \\ &= \frac{1}{\Gamma(\alpha)b^\alpha} x_1^{\alpha-1} e^{-x_1/b} \alpha^{N-1} (r_1 \dots r_{N-1})^{\alpha-1}. \end{aligned}$$

The first part of the envelope is a gamma density, whose curve should be similar to the density function of J_1 . To this end, we set its shape parameter to be α and scale parameter as close as possible to 1 from below, for example $b = 0.99$. The second part can be viewed as the product of $N - 1$ number of Beta($\alpha, 1$) density functions $\alpha r_k^{\alpha-1} \mathbb{1}_{\{0 < r_k < 1\}}$. Then we can sample from R_k using the inverse transform. Finally, the remaining term τ_α^{N-} has the conditional Laplace transform

$$\begin{aligned} \mathbb{E}\left(\exp(-\beta\tau_\alpha^{N-}) \mid J_1, \dots, J_N\right) &= \exp\left(-\alpha \int_0^{J_N} (1 - e^{-\beta y})w^{-1}e^{-w}dw\right) \\ &= \exp\left(-\alpha \int_0^1 (1 - e^{-\beta J_N w})w^{-1}e^{-J_N w}dw\right), \end{aligned}$$

hence $\tau_\alpha^{N-} \stackrel{d}{=} J_N Z_{J_N}(\alpha)$, where $Z_\mu(t)$ is a truncated gamma process with Lévy measure $w^{-1}e^{-\mu w} \mathbb{1}_{\{0 < w < 1\}}dw$ at time t , and the exact simulation algorithm can be found in Dassios et al. (2019). □

When α is large, the maximum of the ratio between the target density and envelope in Step 1 could be large, making the AR algorithm less efficient or even unfeasible. Next, we design an alternative algorithm that works better for large α .

Theorem 4 *Alternative acceptance-rejection algorithm for $(J_1, \dots, J_N, \tau_\alpha^{N-})$.*

1. Numerically maximising

$$\frac{\exp(-\int_{x_1}^\infty w^{-1}e^{-w}dw)x_1^{-1}e^{-x_1}}{(1/b)e^{-x_1/b}}$$

and

$$\frac{\exp(-\alpha \int_{x_1}^\infty w^{-1}e^{-w}dw)\alpha x_1^{-1}e^{-x_1}}{\exp(-\int_{x_1}^\infty w^{-1}e^{-w}dw)x_1^{-1}e^{-x_1}}$$

with respect to $x_1 \in (0, \infty)$, record the maximum values as $C_{1,1}$ and $C_{1,2}$ respectively.

2. Sample from an exponential distribution $X_1 \leftarrow \text{Exp}(b)$.
3. Sample from uniform distributions $V_{1,1} \leftarrow U[0, 1]$ and $V_{1,2} \leftarrow U[0, 1]$, if both

$$V_{1,1} \leq \frac{\exp(-\int_{X_1}^\infty w^{-1}e^{-w}dw)X_1^{-1}e^{-X_1}}{C_{1,1}(1/b)e^{-X_1/b}}$$

and

$$V_{1,2} \leq \frac{\alpha}{C_{1,2}} \exp\left((1 - \alpha) \int_{X_1}^\infty w^{-1}e^{-w}dw\right)$$

are true, accept X_1 and continue; otherwise, go to Step 2.

4. For $k = 2, \dots, N$, do

(a) Numerically maximising

$$\frac{\exp(-\int_{x_k}^{X_{k-1}} w^{-1}e^{-w}dw)x_k^{-1}e^{-x_k}}{1/X_{k-1}}$$

and

$$\frac{\exp(-\alpha \int_{x_k}^{X_{k-1}} w^{-1} e^{-w} dw) \alpha x_k^{-1} e^{-x_k}}{\exp(-\int_{x_k}^{X_{k-1}} w^{-1} e^{-w} dw) x_k^{-1} e^{-x_k}}$$

with respect to $x_k \in (0, X_{k-1})$, record the maximum values as $C_{k,1}$ and $C_{k,2}$ respectively.

(b) Sample from an uniform distribution $X_k \leftarrow U(0, X_{k-1})$.

(c) Sample from uniform distributions $V_{k,1} \leftarrow U(0, 1)$ and $V_{k,2} \leftarrow U(0, 1)$, if both

$$V_{k,1} \leq \frac{\exp(-\int_{X_k}^{X_{k-1}} w^{-1} e^{-w} dw) X_k^{-1} e^{-X_k}}{C_{k,1}(1/X_{k-1})}$$

and

$$V_{k,2} \leq \frac{\alpha}{C_{k,2}} \exp\left((1 - \alpha) \int_{X_k}^{X_{k-1}} w^{-1} e^{-w} dw\right)$$

are true, accept X_k and continue; otherwise, go to Step 4(b).

5. Denote by $\lfloor \alpha \rfloor$ the largest integer smaller than α . For each $i = 1, \dots, \lfloor \alpha \rfloor$, sample from a truncated gamma process with time parameter 1,

$$Z_i \leftarrow X_N \times \text{AlgorithmTG}(1, X_N),$$

then sample from a truncated gamma process with time parameter $\alpha - \lfloor \alpha \rfloor$,

$$Z_{\lfloor \alpha \rfloor + 1} \leftarrow X_N \times \text{AlgorithmTG}(\alpha - \lfloor \alpha \rfloor, X_N),$$

and set $\tau_\alpha^{N-} \leftarrow Z_1 + \dots + Z_{\lfloor \alpha \rfloor} + Z_{\lfloor \alpha \rfloor + 1}$.

6. Set $J_1 \leftarrow X_1, \dots, J_N \leftarrow X_N$, then output $(J_1, \dots, J_N, \tau_\alpha^{N-})$.

Proof To sample from the density of the largest jump J_1 , we use a two-step acceptance-rejection method. The first step is to maximise

$$\frac{\exp(-\int_{x_1}^\infty w^{-1} e^{-w} dw) x_1^{-1} e^{-x_1}}{(1/b)e^{-x_1/b}}$$

for $x_1 \in (0, \infty)$, then we can sample from the density in the numerator with an exponential envelope. The second step is to maximise

$$\frac{\exp(-\alpha \int_{x_1}^\infty w^{-1} e^{-w} dw) \alpha x_1^{-1} e^{-x_1}}{\exp(-\int_{x_1}^\infty w^{-1} e^{-w} dw) x_1^{-1} e^{-x_1}}$$

for $x_1 \in (0, \infty)$. When $\alpha > 1$, the maximum is approximately α . We can then sample from the density in the denominator via the first step, and use it as the envelope to sample from the target density in the numerator, which is the density of J_1 . The same method applies to the k -th largest jump J_k conditional on $J_{k-1} = x_{k-1}$.

When α is large, the exact simulation algorithm for truncated gamma process could be slow, but we can improve its performance by rewriting the Lévy-Khintchine representation of τ_α^{N-} as

$$\begin{aligned} & \mathbb{E} \left(\exp(-\beta \tau_{\alpha}^{N-}) \mid J_1, \dots, J_N \right) \\ &= \exp \left(-([\alpha] + (\alpha - [\alpha])) \int_0^1 (1 - e^{-\beta J_N w}) w^{-1} e^{-J_N w} dw \right). \end{aligned}$$

Then we can simulate $[\alpha]$ number of independent truncated gamma processes, each at time 1; and a truncated gamma process at time $\alpha - [\alpha]$. Their summation is a sample of the truncated gamma process at time α . □

3.3 AR Algorithm for Generalised Gamma Process

This section contains the similar results as before but for a generalised gamma process.

Theorem 5 Denote by (J_1, \dots, J_N) the N largest jumps of a generalised gamma process $\tau_{\alpha, \sigma}$. Let $R_k := J_{k+1}/J_k$ be the ratio between the $(k + 1)$ -th and k -th largest jumps, then the joint density of $(J_1, R_1, \dots, R_{N-1})$ is

$$\begin{aligned} & \mathbb{P}(J_1 \in dx_1, R_1 \in dr_1, \dots, R_{N-1} \in dr_{N-1}) \\ &= \exp \left(-\alpha \int_{x_1 r_1 \dots r_{N-1}}^{\infty} w^{-\sigma-1} e^{-w} dw \right) \alpha^N x_1^{-N\sigma-1} r_1^{-(N-1)\sigma-1} \dots r_{N-1}^{-\sigma-1} \\ & \quad \times e^{-x_1(1+r_1+r_1 r_2+\dots+r_1 r_2 \dots r_{N-1})} dr_{N-1} \dots dr_1 dx_1, \end{aligned}$$

where $x_1 \in (0, \infty)$ and $r_k \in (0, 1)$, for $k = 1, \dots, N - 1$.

Proof This is a repeat of the proof of Theorem 2 with a different Lévy measure, we omit the details. □

Next, we design an acceptance-rejection algorithm to sample from $(J_1, \dots, J_N, \tau_{\alpha, \sigma}^{N-})$. It is Algorithm 4.4 of Dassios et al. (2020) which we refer to as *Algorithm TTS*(\cdot, \cdot, \cdot).

Theorem 6 The acceptance-rejection algorithm for $(J_1, \dots, J_N, \tau_{\alpha, \sigma}^{N-})$.

1. Numerically maximising

$$\frac{\exp(-\alpha \int_{x_1 r_1 \dots r_{N-1}}^{\infty} w^{-\sigma-1} e^{-w} dw) e^{-x_1(1+r_1+\dots+r_1 r_2 \dots r_{N-1})}}{\exp(-\alpha (x_1 r_1 \dots r_{N-1})^{-\sigma} / \sigma)}$$

with respect to $x_1 \in (0, \infty)$ and $r_k \in (0, 1)$, for $k = 1, \dots, N - 1$. Record the maximum value $C_{\alpha, \sigma, N}$.

2. Sample from an uniform distribution $U \leftarrow U(0, 1)$, then set

$$X_1 \leftarrow (-\sigma \ln(U)/\alpha)^{-1/\sigma},$$

and for each $k = 1, \dots, N - 1$, sample from an uniform distribution $U_k \leftarrow U(0, 1)$, then set

$$R_k \leftarrow (1 - \sigma \ln(U_k)(X_1 R_1 \dots R_{k-1})^\sigma / \alpha)^{-1/\sigma}.$$

3. Sample from an uniform distribution $V \leftarrow U(0, 1)$, if

$$V \leq \frac{\exp(-\alpha \int_{X_1 R_1 \dots R_{N-1}}^{\infty} w^{-\sigma-1} e^{-w} dw) e^{-X_1(1+R_1+\dots+R_1 R_2 \dots R_{N-1})}}{C_{\alpha, \sigma, N} \exp(-\alpha (X_1 R_1 \dots R_{N-1})^{-\sigma} / \sigma)},$$

- accept the candidates $(X_1, R_1, \dots, R_{N-1})$ and go to Step 4; otherwise, go to Step 2.
4. Set $J_1 \leftarrow X_1$, and for $k = 2, \dots, N$, set $J_k \leftarrow J_1 \prod_{i=1}^{k-1} R_i$.
 5. Set $\tau_{\alpha, \sigma}^{N-} \leftarrow J_N \times \text{AlgorithmTTS}(\sigma, J_N, \alpha J_N^{-\sigma})$.
 6. Output $(J_1, \dots, J_N, \tau_{\alpha, \sigma}^{N-})$.

Proof We apply the acceptance-rejection method to sample from the joint density of $(J_1, R_1, \dots, R_{N-1})$ in Theorem 5 with the envelope

$$\begin{aligned} & \exp\left(-\alpha \int_{x_1 r_1 \dots r_{N-1}}^{\infty} w^{-\sigma-1} dw\right) \alpha^N x_1^{-N\sigma-1} r_1^{-(N-1)\sigma-1} \dots r_{N-1}^{-\sigma-1} \\ &= \exp(-\alpha x_1^{-\sigma} / \sigma) \alpha x_1^{-\sigma-1} \\ & \quad \times \prod_{k=1}^{N-1} \exp(\alpha(x_1 r_1 \dots r_{k-1})^{-\sigma} (1 - r_k^{-\sigma}) / \sigma) \alpha(x_1 r_1 \dots r_{k-1})^{-\sigma} r_k^{-\sigma-1}. \end{aligned}$$

The envelope can be divided into several independent components and each of them can be sampled from using the inverse transform. For the first part, we set

$$U = \exp(-\alpha x_1^{-\sigma} / \sigma),$$

and solve for

$$X_1 = (-\sigma \ln(U) / \alpha)^{-1/\sigma}.$$

While for each component in the product term, we set

$$U = \exp(\alpha(x_1 r_1 \dots r_{k-1})^{-\sigma} (1 - r_k^{-\sigma}) / \sigma),$$

and solve for

$$R_k = (1 - \sigma \ln(U) (X_1 R_1 \dots R_{k-1})^\sigma / \alpha)^{-1/\sigma}.$$

Finally, we know the remaining term $\tau_{\alpha, \sigma}^{N-}$ has the conditional Laplace transform

$$\begin{aligned} \mathbb{E}\left(\exp(-\beta \tau_{\alpha, \sigma}^{N-}) \mid J_1, \dots, J_N\right) &= \exp\left(-\alpha \int_0^{J_N} (1 - e^{-\beta w}) w^{-\sigma-1} e^{-w} dw\right) \\ &= \exp\left(-\alpha J_N^{-\sigma} \int_0^1 (1 - e^{-\beta J_N w}) w^{-\sigma-1} e^{-J_N w} dw\right), \end{aligned}$$

hence $\tau_{\alpha, \sigma}^{N-} \stackrel{d}{=} J_N Z_{\sigma, J_N}(\alpha J_N^{-\sigma})$, where $Z_{\sigma, \mu}(t)$ is a truncated tempered stable process with the Lévy measure $w^{-\sigma-1} e^{-\mu w} \mathbb{1}_{\{0 < w < 1\}} dw$ at time t , and the exact simulation algorithm can be found in Dassios et al. (2020). □

For the same reason as before, we provide an alternative algorithm for the generalised gamma process.

Theorem 7 *Alternative AR algorithm for $(J_1, \dots, J_N, \tau_{\alpha, \sigma}^{N-})$.*

1. Numerically maximising

$$\frac{\exp(-\int_{x_1}^{\infty} w^{-\sigma-1} e^{-w} dw) x_1^{-\sigma-1} e^{-x_1}}{(1/b) e^{-x_1/b}}$$

and

$$\frac{\exp(-\alpha \int_{x_1}^{\infty} w^{-\sigma-1} e^{-w} dw) \alpha x_1^{-\sigma-1} e^{-x_1}}{\exp(-\int_{x_1}^{\infty} w^{-\sigma-1} e^{-w} dw) x_1^{-\sigma-1} e^{-x_1}}$$

with respect to $x_1 \in (0, \infty)$, record the maximum values as $C_{1,1}$ and $C_{1,2}$ respectively.

2. Sample from an exponential distribution $X_1 \leftarrow \text{Exp}(b)$.
3. Sample from uniform distributions $V_{1,1} \leftarrow U(0, 1)$ and $V_{1,2} \leftarrow U(0, 1)$, if both

$$V_{1,1} \leq \frac{\exp(-\int_{X_1}^{\infty} w^{-\sigma-1} e^{-w} dw) X_1^{-\sigma-1} e^{-X_1}}{C_{1,1} (1/b) e^{-X_1/b}}$$

and

$$V_{1,2} \leq \frac{\exp(-\alpha \int_{x_1}^{\infty} w^{-\sigma-1} e^{-w} dw) \alpha x_1^{-\sigma-1} e^{-x_1}}{C_{1,2} \exp(-\int_{x_1}^{\infty} w^{-\sigma-1} e^{-w} dw) x_1^{-\sigma-1} e^{-x_1}}$$

are true, accept X_1 and continue; otherwise, go to Step 2.

4. For $k = 2, \dots, N$, do

(a) Numerically maximising

$$\frac{\exp(-\int_{x_k}^{X_{k-1}} w^{-\sigma-1} e^{-w} dw) x_k^{-\sigma-1} e^{-x_k}}{1/X_{k-1}}$$

and

$$\frac{\exp(-\alpha \int_{x_k}^{X_{k-1}} w^{-\sigma-1} e^{-w} dw) \alpha x_k^{-\sigma-1} e^{-x_k}}{\exp(-\int_{x_k}^{X_{k-1}} w^{-\sigma-1} e^{-w} dw) x_k^{-\sigma-1} e^{-x_k}}$$

with respect to $x_k \in (0, X_{k-1})$, record the maximum values as $C_{k,1}$ and $C_{k,2}$ respectively.

- (b) Sample from an uniform distribution $X_k \leftarrow U(0, X_{k-1})$.
- (c) Sample from uniform distributions $V_{k,1} \leftarrow U(0, 1)$ and $V_{k,2} \leftarrow U(0, 1)$, if both

$$V_{k,1} \leq \frac{\exp(-\int_{X_k}^{X_{k-1}} w^{-\sigma-1} e^{-w} dw) X_k^{-\sigma-1} e^{-X_k}}{C_{k,1} (1/X_{k-1})}$$

and

$$V_{k,2} \leq \frac{\exp(-\alpha \int_{X_k}^{X_{k-1}} w^{-\sigma-1} e^{-w} dw) \alpha X_k^{-\sigma-1} e^{-X_k}}{C_{k,2} \exp(-\int_{X_k}^{X_{k-1}} w^{-\sigma-1} e^{-w} dw) X_k^{-\sigma-1} e^{-X_k}}$$

are true, accept X_k and continue; otherwise, go to Step 4(b).

5. For each $i = 1, \dots, \lfloor \alpha X_N^{-\sigma} \rfloor$, sample from a truncated tempered stable process with time parameter 1,

$$Z_i \leftarrow X_N \times \text{AlgorithmTTS}(\sigma, X_N, 1),$$

then sample from a truncated tempered stable process with time parameter $\alpha X_N^{-\sigma} - \lfloor \alpha X_N^{-\sigma} \rfloor$,

$$Z_{\lfloor \alpha X_N^{-\sigma} \rfloor + 1} \leftarrow X_N \times \text{AlgorithmTTS}(\sigma, X_N, \alpha X_N^{-\sigma} - \lfloor \alpha X_N^{-\sigma} \rfloor),$$

and set $\tau_{\alpha,\sigma}^{N-} \leftarrow Z_1 + \dots + Z_{\lfloor \alpha X_N^{-\sigma} \rfloor} + Z_{\lfloor \alpha X_N^{-\sigma} \rfloor + 1}$.
 6. Set $J_1 \leftarrow X_1, \dots, J_N \leftarrow X_N$, then output $(J_1, \dots, J_N, \tau_{\alpha,\sigma}^{N-})$.

Proof This is similar to the proof of Algorithm 4, we omit the details. □

3.4 Inhomogeneous Processes

In Corollary 1 of Wolpert and Ickstadt (1998b), the ILM algorithm was extended to inhomogeneous processes. Consider an inhomogeneous generalised gamma process $\tau_{\alpha,\sigma,\Pi}$ with the Lévy-Khintchine representation

$$\mathbb{E}(\exp(-\beta\tau_{\alpha,\sigma,\Pi})) = \exp\left(-\int_{\mathcal{S}} \alpha(s) \int_0^\infty (1 - e^{-\beta w}) w^{-\sigma(s)-1} e^{-w} dw \Pi(ds)\right),$$

where $\alpha(s) \geq 0$ and $\sigma(s) \geq 0$ are measurable functions on a space \mathcal{S} , and $\Pi(ds)$ is a probability measure on \mathcal{S} . Then we can sample from the jumps of $\tau_{\alpha,\sigma,\Pi}$ using the following algorithm. For $k = 1, \dots, N$, we generate independent samples s_k from $\Pi(ds)$ and construct the event times of a Poisson process by setting $T_k := \sum_{i=1}^k e_i$, where e_i are i.i.d. exponential random variables with mean 1. Then we solve the equation $T_k = \int_{J_k}^\infty \alpha(s_k) w^{-\sigma(s_k)-1} e^{-w} dw$ to find out J_k , for $k = 1, \dots, N$. The inhomogeneous generalised gamma process is approximated by $\tau_{\alpha,\sigma,\Pi} \leftarrow \sum_{k=1}^N J_k$.

To recast the algorithm in terms of the acceptance-rejection method, we notice that Corollary 2 of Wolpert and Ickstadt (1998a) provided the joint density of the jump sizes and locations $\{(J_k, s_k)\}_{k=1}^N$ as follows,

$$\exp\left(-\alpha(s_N) \int_{J_N}^\infty w^{-\sigma(s_N)-1} e^{-w} dw\right) \prod_{k=1}^N \left(\alpha(s_k) J_k^{-\sigma(s_k)-1} e^{-J_k}\right), \tag{3}$$

with respect to the product measure $\prod_{k=1}^N dJ_k \Pi(ds_k)$ and the constraint

$$\alpha(s_k) \int_{J_k}^\infty w^{-\sigma(s_k)-1} e^{-w} dw < \alpha(s_{k+1}) \int_{J_{k+1}}^\infty w^{-\sigma(s_{k+1})-1} e^{-w} dw, \tag{4}$$

for every $k < N$.

If the hyperparameters are fixed, i.e., $\alpha(\cdot) \equiv \alpha$ and $\sigma(\cdot) \equiv \sigma$, Eqs. (3) and (4) reduce to the joint density of the N largest jumps of a generalised gamma process, see Theorem 5. However, with the hyperparameters depending on the jump locations, it is hard to develop an acceptance-rejection method directly from (3), especially when the ranges of variables in (4) are not explicit.

To sample from the jumps of the inhomogeneous generalised gamma process via the AR algorithm, we attempt the following procedure. For every $k = 1, \dots, N$, we draw a jump location s_k from $\Pi(ds)$, then use the AR algorithm in Sect. 3.3 to sample from the k -th largest jump of a homogeneous generalised gamma process with the Lévy-Khintchine representation

$$\exp\left(-\alpha(s_k) \int_0^\infty (1 - e^{-\beta w}) w^{-\sigma(s_k)-1} e^{-w} dw\right),$$

and take it as a jump of the inhomogeneous generalised gamma process. The numerical illustration of this algorithm is given in the next section.

4 Numerical Implementation

In this section, we present some numerical results for the algorithms introduced in Sect. 3. The experiments are conducted on an Intel Core i9–10900 CPU @ 2.80GHz processor, 64.0 GB RAM, Windows 10, 64-bit system and performed in Matlab R2023a. Parallel computing is not used throughout the section.

4.1 Jump Mean

We first use the AR algorithms to sample from the five largest jumps of a (generalised) gamma process and compare their sample averages to the true means. To this end, we calculate the mean of these jumps as follows.

Proposition 1 (*Mean*) *The mean of the k -th largest jump of a gamma process is*

$$\mathbb{E}(J_k) = \int_0^\infty \int_0^1 \dots \int_0^1 \exp\left(-\alpha \int_{x_1 r_1 \dots r_{k-1}}^\infty w^{-1} e^{-w} dw\right) \times \alpha^k e^{-x_1(1+r_1+\dots+r_{k-1})} dr_{k-1} \dots dr_1 dx_1,$$

and that of a generalised gamma process is

$$\mathbb{E}(J_k) = \int_0^\infty \int_0^1 \dots \int_0^1 \exp\left(-\alpha \int_{x_1 r_1 \dots r_{k-1}}^\infty w^{-\sigma-1} e^{-w} dw\right) \times \alpha^k x_1^{-k\sigma} r_1^{-(k-1)\sigma} \dots r_{k-1}^{-\sigma} e^{-x_1(1+r_1+\dots+r_{k-1})} dr_{k-1} \dots dr_1 dx_1.$$

Proof These are direct consequences of Theorems 2 and 5. □

Both expectations can be estimated by Monte Carlo method with an exponential variate for x_1 and an uniform variate for each $r_i, i = 1, \dots, k - 1$. We record the numerical results in Table 1 and 2. From the tables we can see that the AR algorithms can sample from the jumps exactly. Moreover, when the acceptance rate is high, the AR algorithm would be more efficient than the ILM algorithm. But in general, the ILM algorithm is faster.

Moreover, using the connection between the gamma process and Poisson-Dirichlet distribution in Eq. (1), we can sample from the N largest components of a $PD(\alpha)$ distribution exactly. We compare the results to the approximation obtained from Algorithm 1. The numerical results are recorded in Table 3.

4.2 Jump Covariance

Next, we study the true and empirical covariance between the jumps. The following proposition implies a Monte Carlo method that estimates the true covariance. For simplicity, we assume that $m \geq n$ and focus on the product term $J_m J_n$ only.

Proposition 2 (*Covariance*) *Assume that $m \geq n \geq 1$, then for a gamma process, the expectation of the product $J_m J_n$ is*

$$\mathbb{E}(J_m J_n) = \int_0^\infty \int_0^1 \dots \int_0^1 (x_1 r_1 \dots r_{n-1}) \exp\left(-\alpha \int_{x_1 r_1 \dots r_{m-1}}^\infty w^{-1} e^{-w} dw\right) \times \alpha^m e^{-x_1(1+r_1+\dots+r_{m-1})} dr_{m-1} \dots dr_1 dx_1,$$

and that of a generalised gamma process is

Table 1 Jump sizes of the gamma process, $N = 5$, sample size is 10^4 . CPU time is rounded to the nearest integer. AR1(GP) and AR2(GP) stand for Theorems 3 and 4 respectively

α	Algorithm	J_1	J_2	J_3	J_4	J_5	τ_α^{N-}	Total (τ_α)	Time (secs)
$\alpha = 0.5$	true mean	0.3784	0.0855	0.0245	0.0076	0.0024	0.0012	0.5	N.A
	IML	0.3728	0.0867	0.0243	0.0077	0.0025	0.0012	0.4953	9
$\alpha = 1.0$	AR1(GP)	0.3810	0.0856	0.0242	0.0074	0.0025	0.0012	0.5019	< 1
	true mean	0.6241	0.2093	0.0883	0.0404	0.0190	0.0183	1.0	N.A
$\alpha = 4.0$	IML	0.6210	0.2105	0.0884	0.0405	0.0192	0.0184	0.9981	8
	AR1(GP)	0.6289	0.2065	0.0872	0.0397	0.0193	0.0190	1.0007	< 1
$\alpha = 5.0$	true mean	1.3454	0.7349	0.4830	0.3434	0.2457	0.8403	4.0	N.A
	IML	1.3413	0.7328	0.4810	0.3404	0.2495	0.8474	3.9925	6
$\alpha = 5.0$	AR1(GP)	1.3727	0.7464	0.4886	0.3446	0.2516	0.8502	4.0541	42
	AR2(GP)	1.3678	0.7415	0.4846	0.3382	0.2476	0.8378	4.0176	32
$\alpha = 5.0$	true mean	1.4857	0.8447	0.5869	0.4262	0.3288	1.3173	5.0	N.A
	IML	1.4763	0.8429	0.5792	0.4248	0.3232	1.3217	4.9680	7
$\alpha = 5.0$	AR1(GP)	-	-	-	-	-	-	-	∞
	AR2(GP)	1.4766	0.8505	0.5827	0.4266	0.3246	1.3314	4.9925	34

Table 2 Jump sizes of the generalised gamma process, $N = 5$, sample size is 10^4 . CPU time is rounded to the nearest integer. AR1(GGP) and AR2(GGP) stand for Theorems 6 and 7 respectively

	Algorithm	J_1	J_2	J_3	J_4	J_5	$\tau_{\alpha, \sigma}^{N-}$	Total ($\tau_{\alpha, \sigma}$)	Time (secs)
$\sigma = 0.5 \alpha = 1$	true mean	0.6135	0.2739	0.1634	0.1121	0.0825	0.5306	1.7725	N.A
	IML	0.6089	0.2757	0.1655	0.1118	0.0808	0.5290	1.7717	9
	AR1(GGP)	0.6173	0.2725	0.1650	0.1115	0.0811	0.5267	1.7742	8
$\sigma = 0.25 \alpha = 2$	true mean	0.9078	0.4414	0.2695	0.1823	0.1320	0.5128	2.4508	N.A
	IML	0.9090	0.4428	0.2701	0.1829	0.1294	0.5139	2.4481	22
	AR1(GGP)	0.9056	0.4384	0.2695	0.1821	0.1302	0.5178	2.4435	651
$\sigma = 0.75 \alpha = 3$	AR2(GGP)	0.9082	0.4435	0.2716	0.1823	0.1306	0.5178	2.4540	29
	true mean	1.0130	0.5943	0.4335	0.3477	0.2813	8.2433	10.8768	N.A
	IML	1.0157	0.5949	0.4346	0.3450	0.2866	8.2081	10.8848	17
$\sigma = 0.5 \alpha = 5$	AR1(GGP)	-	-	-	-	-	-	-	∞
	AR2(GGP)	1.0094	0.5921	0.4296	0.3414	0.2838	8.2038	10.8601	25
	true mean	1.3020	0.7881	0.5901	0.4695	0.3809	5.3801	8.8623	N.A
$\sigma = 0.5 \alpha = 5$	IML	1.3014	0.7935	0.5843	0.4645	0.3828	5.3643	8.8908	18
	AR1(GGP)	-	-	-	-	-	-	-	∞
	AR2(GGP)	1.3044	0.7944	0.5824	0.4611	0.3802	5.3349	8.8574	27

Table 3 Sample averages of the first 5 components of a Poisson-Dirichlet distribution, $m = 50$ for Algorithm 1, sample size is 10^4

	Algorithm	$\pi_1 = J_1/\tau_\alpha$	$\pi_2 = J_2/\tau_\alpha$	$\pi_3 = J_3/\tau_\alpha$	$\pi_4 = J_4/\tau_\alpha$	$\pi_5 = J_5/\tau_\alpha$	Summation
$\alpha = 0.5$	Algorithm 1	0.7592	0.1698	0.0484	0.0153	0.0049	0.9977
	Algorithm 3	0.7637	0.1667	0.0480	0.0150	0.0046	0.9980
	Algorithm 1	0.6250	0.2085	0.0885	0.0404	0.0192	0.9815
$\alpha = 1.0$	Algorithm 3	0.6250	0.2089	0.0883	0.0407	0.0189	0.9818
	Algorithm 1	0.3366	0.1836	0.1205	0.0854	0.0625	0.7886
	Algorithm 4	0.3356	0.1837	0.1212	0.0851	0.0622	0.7878
$\alpha = 5.0$	Algorithm 1	0.2965	0.1701	0.1166	0.0854	0.0645	0.7331
	Algorithm 4	0.2969	0.1698	0.1160	0.0856	0.0653	0.7336

Table 4 Covariance between the jumps of the gamma process, $\alpha = 1$, sample size is 10^4

	J_1	J_2	J_3	J_4	J_5
J_1	(0.8529, 0.8727)	(0.2293, 0.2335)	(0.0901, 0.0917)	(0.0399, 0.0409)	(0.0186, 0.0190)
J_2		(0.1130, 0.1145)	(0.0423, 0.0429)	(0.0183, 0.0187)	(0.0085, 0.0086)
J_3			(0.0246, 0.0250)	(0.0104, 0.0107)	(0.0048, 0.0048)
J_4				(0.0064, 0.0067)	(0.0029, 0.0030)
J_5					(0.0018, 0.0019)

$$\mathbb{E}(J_m J_n) = \int_0^\infty \int_0^1 \dots \int_0^1 (x_1 r_1 \dots r_{n-1}) \exp\left(-\alpha \int_{x_1 r_1 \dots r_{m-1}}^\infty w^{-\sigma-1} e^{-w} dw\right) \times \alpha^m x_1^{-m\alpha} r_1^{-(m-1)\alpha} \dots r_{m-1}^{-\alpha} e^{-x_1(1+r_1+\dots+r_1 \dots r_{m-1})} dr_{m-1} \dots dr_1 dx_1 \dots$$

Proof These are direct consequences of Theorems 2 and 5. □

Then we run the AR algorithms and record the empirical covariance in Tables 4 and 5. The data are presented in the format (a, b) , where a represents the true covariance obtained from Proposition 9 and b represents the sample average of $J_m J_n$.

4.3 Functional Mean

We can also compare the true and empirical Laplace transforms of the gamma and generalised gamma processes. From the Lévy-Khintchine representations, we derive their Laplace transforms explicitly to be $\mathbb{E}(\exp(-\beta\tau_\alpha)) = (\beta + 1)^{-\alpha}$ and $\mathbb{E}(\exp(-\beta\tau_{\alpha,\sigma})) = \exp(-\alpha\Gamma(1 - \sigma)((\beta + 1)^\sigma - 1)/\sigma)$ respectively. On the other hand, we sample from τ_α and $\tau_{\alpha,\sigma}$ using the AR algorithms and calculate the sample average of $\exp(-\beta\tau_\alpha)$ and $\exp(-\beta\tau_{\alpha,\sigma})$ with some specific β .

Table 5 Covariance between the jumps of the generalised gamma process, $\alpha = 1, \sigma = 0.5$, sample size is 10^4

	J_1	J_2	J_3	J_4	J_5
J_1	(0.6780, 0.6774)	(0.2343, 0.2344)	(0.1282, 0.1282)	(0.0825, 0.0824)	(0.0582, 0.0580)
J_2		(0.1227, 0.1229)	(0.0641, 0.0644)	(0.0406, 0.0405)	(0.0282, 0.0281)
J_3			(0.0416, 0.0418)	(0.0258, 0.0258)	(0.0178, 0.0177)
J_4				(0.0183, 0.0183)	(0.0125, 0.0124)
J_5					(0.0094, 0.0093)

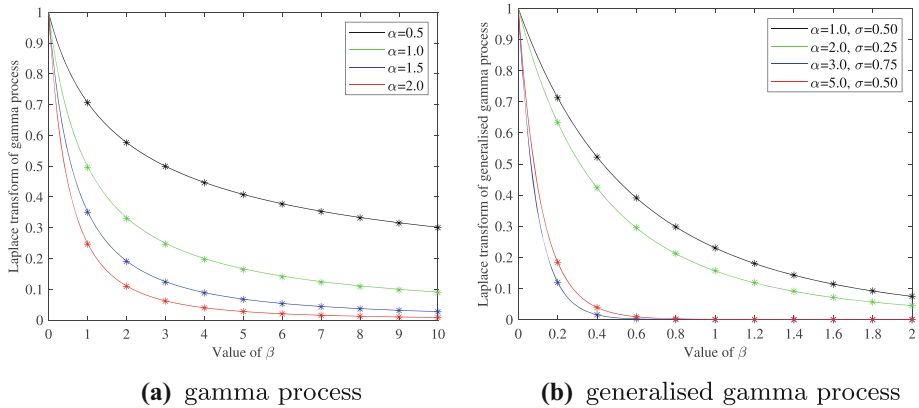


Fig. 1 True and empirical Laplace transforms, sample size is 10^4

In Fig. 1a and b, we plot the true Laplace transform curve in solid line and mark the sample average by a star. The figures show that the AR algorithms are accurate in estimating the Laplace transforms of the gamma and generalised gamma processes.

4.4 Posterior Distribution of the Functional Mean

To illustrate the performance of the AR algorithms in Bayesian nonparametric statistics, we provide another example based on the posterior distribution of the functional mean of a Dirichlet process. For ease of comparison, we consider the same model settings as in Example 1 of Muliere and Tardella (1998), where the target was to study the performance of the ϵ -Dirichlet process.

Assume that we have observed two samples $x_1 = 0.1$ and $x_2 = 0.05$ from a Dirichlet process G with concentration parameter α and reference distribution $U(0, 1)$, it is well-known (see, for example, Ferguson 1973) that the posterior distribution of G given x_1 and x_2 is another Dirichlet process with concentration parameter $\alpha + 2$ and reference distribution $(\alpha U(0, 1) + \delta_{x_1} + \delta_{x_2})/(\alpha + 2)$. Since the posterior is built based on a Dirichlet process, we can sample from its largest components via the representations in Eq. (1).

In Table 6, we use the ILM and AR algorithms to sample from the functional mean of the posterior. We also present the existing results in Muliere and Tardella (1998). From the table we can see that both the ILM and AR algorithms work well in estimating the posterior functional mean.

Table 6 Posterior distribution of the functional mean of a Dirichlet process, sample size is 10^4

Total mass	Truth	ILM $N = 100$	AR2 $N = 100$	ϵ -Dirichlet $\epsilon = 0.1$	ϵ -Dirichlet $\epsilon = 10^{-6}$	MS(1996)	M. Chain
1	0.2166	0.2156	0.2157	0.2203	0.2173	0.2156	0.2152
2	0.2875	0.2851	0.2879	0.2825	0.2876	0.2888	0.2862
100	0.4916	0.4913	0.4910	0.4916	0.4914	0.4918	0.4891

Table 7 Inhomogeneous generalised gamma process with $\alpha(s) = -0.5s + 0.6$, $\sigma(s) = 0.2 \sin(\pi s) + 0.4$ and $\Pi(s) = 0.4s + 0.8$, $0 < s < 1$. Jump sizes of the generalised gamma process, $N = 100$, sample size is 10^4

Algorithm	J_1	J_2	J_3	J_4	J_5	...	J_N	Total ($\tau_{\sigma, \alpha, \Pi}$)
IML	0.3064	0.1035	0.0527	0.0321	0.0219	...	0.0001	0.6264
AR2(GGP)	0.3066	0.1045	0.0529	0.0319	0.0216	...	0.0001	0.6242

4.5 Inhomogeneous Generalised Gamma Process

Finally, we illustrate the numerical results for the inhomogeneous generalised gamma process. We set $\alpha(s) = -0.5s + 0.6$, $\sigma(s) = 0.2 \sin(\pi s) + 0.4$ and $\Pi(s) = 0.4s + 0.8$, for $0 < s < 1$, and sample 100 jumps from the inhomogeneous generalised gamma process. Due to the size of the data, we only present the first 5 jumps in Table 7. We also use the sum of the jumps to approximate the inhomogeneous generalised gamma process; they are compared to the true expectation given by $\mathbb{E}(\tau_{\sigma, \alpha, \Pi}) = \int_0^1 \alpha(s)\Gamma(1 - \sigma(s))\Pi(ds) = 0.6393$.

5 Concluding Remarks

In this research, we design an acceptance-rejection algorithm that samples from the N largest jumps of gamma and generalised gamma processes. The effectiveness of our algorithms is demonstrated by the empirical mean, covariance and functional mean of the jumps. Moreover, we use the normalised jumps of a gamma process to construct the N largest components of a Poisson-Dirichlet distribution and provide an application in estimating the posterior function mean of a Dirichlet process. We also extend the simulation algorithm to inhomogeneous generalised gamma process. In general, the AR algorithm does not overperform the ILM algorithm in terms of efficiency, but it provides another simulation approach where the numerical inversion of exponential or gamma integral is replaced by an acceptance-rejection step. Specific applications are still to be found in which the AR algorithm is of benefit.

From the Bayesian nonparametrics’ point of view, our algorithms provide a truncation method for the Dirichlet and generalised Dirichlet process priors. Moreover, since our construction is based on a decreasing sequence of random weights, it has the lowest truncation error comparing to other methods, see Campbell et al. (2019) for more details. Apart from that, our algorithms sample from the actual jump sizes J_k , not only their weights $\pi_k = J_k/\tau_\alpha$. The actual jump sizes are essential in some statistical models. For example, in the Caron-Fox network model (see Caron and Fox 2017), the jump sizes are used to construct an atomic random measure $\sum_{k=1}^\infty J_k \delta_{\phi_k}$, which describes the sociability parameters of the nodes in a network. The actual jump sizes are important because it was assumed that the total number of links in the network follows a Poisson distribution with mean $(\sum_{k=1}^\infty J_k)^2$, which clearly can not be obtained from the relative weights of the jumps. It would be interesting to investigate the properties of the network model when its nodes are sorted in descending order according to their sociability parameters; we will study this topic in a future work.

Acknowledgements The authors are grateful to the editor and referee for giving us extremely helpful comments and pointing out important references.

Funding The authors did not receive support from any organization for the submitted work.

Data Availability The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aldous DJ (1985) Exchangeability and related topics. In: *École d'été de probabilités de Saint-Flour, Lecture Notes in Math*, vol 1117. Springer, Berlin
- Arratia R, Barbour AD, Tavaré S (2003) *Logarithmic combinatorial structures: a probabilistic approach*. EMS Monographs in Mathematics, European Mathematical Society (EMS), Zürich
- Ayed F, Lee J, Caron F (2019) Beyond the Chinese restaurant and Pitman-Yor processes: Statistical models with double power-law behavior. In: *International Conference on Machine Learning*. PMLR, pp 395–404
- Billingsley P (1972) On the distribution of large prime divisors. *Period Math Hungar* 2:283–289
- Brix A (1999) Generalized Gamma measures and shot-noise Cox processes. *Adv in Appl Probab* 31(4):929–953
- Campbell T, Huggins JH, How JP et al (2019) Truncated random measures. *Bernoulli* 25(2):1256–1288
- Caron F, Fox EB (2017) Sparse graphs using exchangeable random measures. *J R Stat Soc Ser B Stat Methodol* 79(5):1295–1366
- Dassios A, Qu Y, Lim JW (2019) Exact simulation of generalised Vervaat perpetuities. *J Appl Probab* 56(1):57–75
- Dassios A, Lim JW, Qu Y (2020) Exact simulation of a truncated Lévy subordinator. *ACM Trans Model Comput Simul* 30(3):Art. 17, 26
- Donnelly P, Grimmett G (1993) On the asymptotic distribution of large prime factors. *J London Math Soc* (2) 47(3):395–404
- Epifani I, Lijoi A, Prünster I (2003) Exponential functionals and means of neutral-to-the-right priors. *Biometrika* 90(4):791–808
- Ewens WJ (1990) Population genetics theory—the past and the future. In: *Mathematical and statistical developments of evolutionary theory* (Montreal, PQ, 1987), NATO Adv. Sci Inst Ser C Math Phys Sci vol 299. Kluwer Acad. Publ., Dordrecht, p 177–227
- Ferguson TS (1973) A Bayesian analysis of some nonparametric problems. *Ann Statist* 1:209–230
- Ferguson TS, Klass MJ (1972) A representation of independent increment processes without Gaussian components. *Ann Math Statist* 43:1634–1643
- Hansen JC (1994) Order statistics for decomposable combinatorial structures. *Random Structures Algorithms* 5(4):517–533
- Hougaard P (1986) Survival models for heterogeneous populations derived from stable distributions. *Biometrika* 73(2):387–396
- James L, Lijoi A, Prünster I (2005) Bayesian inference for classes of normalized random measures. Available at <http://arxiv.org/abs/math.ST/0503394>
- Kingman JFC (1980) *Mathematics of genetic diversity*, CBMS-NSF Regional Conference Series in Applied Mathematics, vol 34. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pa
- Kyprianou AE (2006) *Introductory lectures on fluctuations of Lévy processes with applications*. Universitext, Springer-Verlag, Berlin
- Lijoi A, Mena RH, Prünster I (2007) Controlling the reinforcement in Bayesian non-parametric mixture models. *J R Stat Soc Ser B Stat Methodol* 69(4):715–740

- Muliere P, Tardella L (1998) Approximating distributions of random functionals of Ferguson-Dirichlet priors. *Canad J Statist* 26(2):283–297
- Perman M, Pitman J, Yor M (1992) Size-biased sampling of Poisson point processes and excursions. *Probab Theory Related Fields* 92(1):21–39
- Schmidt AA, Vershik AM (1977) Limit measures that arise in the asymptotic theory of symmetric groups. I. *Teor Verojatnost i Primenen* 22(1):72–88
- Tenenbaum G (1995) Introduction to analytic and probabilistic number theory, Cambridge Studies in Advanced Mathematics, vol 46. Cambridge University Press, Cambridge
- Vershik AM (1986) Asymptotic distribution of decompositions of natural numbers into prime divisors. *Dokl Akad Nauk SSSR* 289(2):269–272
- Watterson GA (1976) The stationary distribution of the infinitely-many neutral alleles diffusion model. *J Appl Probability* 13(4):639–651
- Wolpert RL, Ickstadt K (1998a) Poisson/gamma random field models for spatial statistics. *Biometrika* 85(2):251–267
- Wolpert RL, Ickstadt K (1998b) Simulation of Lévy random fields. In: Practical nonparametric and semiparametric Bayesian statistics, Lect Notes Stat vol 133. Springer, New York, p 227–242
- Zarepour M, Labadi LA (2012) On a rapid simulation of the Dirichlet process. *Statist Probab Lett* 82(5):916–924

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.