Research Article

Henrik Sloot*

Implementing Markovian models for extendible Marshall–Olkin distributions

https://doi.org/10.1515/demo-2022-0151
received May 20, 2022; accepted November 2, 2022

Abstract: We derive a novel stochastic representation of exchangeable Marshall–Olkin distributions based on their death-counting processes. We show that these processes are Markov. Furthermore, we provide a numerically stable approximation of their infinitesimal generator matrices in the extendible case. This approach uses integral representations of Bernstein functions to calculate the generator's first row, and then uses a recursion to calculate the remaining rows. Combining the Markov representation with the numerically stable approximation of corresponding generators allows us to sample extendible Marshall–Olkin distributions with a flexible simulation algorithm derived from known Markov sampling strategies. Finally, we benchmark an implementation of this Markov-based simulation algorithm against alternative simulation algorithms based on the Lévy frailty model, the Arnold model, and the exogenous shock model.

Keywords: Marshall–Olkin distribution, sampling algorithm, Markov processes, exchangeability, lack of memory, multivariate survival analysis

MSC 2020: 60G09, 60J28, 62-08, 62H05

1 Introduction

Simulating a multivariate distribution can be complex: even if a stochastic model is known, implementing a model into a simulation algorithm requires considering and overcoming numerical problems, methodological challenges, and efficiency obstacles. A typical numerical problem is aggregation operations over large or infinite sets, requiring approximations and introducing a bias. A methodological challenge is choosing between general algorithms covering a large family of distributions or specialized algorithms targeting smaller subfamilies. General algorithms can be valuable for a general-purpose application or cover cases for which a specialized simulation algorithm does not exist. However, specialized simulation algorithms can be less complex, faster, or have a smaller memory footprint. Examples of this are the canonical implementations of the natural models for independent and comonotonic multivariate distributions. An efficiency obstacle often arises from implementation details, e.g., storage requirements and limitations or algorithm choices for subtasks. Thus, developing, selecting, and implementing general or specialized simulation algorithms for a multivariate distribution is an intriguing and challenging research area.

The Marshall–Olkin distribution is a multivariate exponential distribution. It was identified in [19] as the unique class of multivariate distributions whose margins possess the following multivariate generalization of the lack of memory property: given joint survival up to a particular time, the excess-time distribution equals the original distribution. A familiar characterizing property was derived in [5]: the Marshall–Olkin distribution is the unique class of multivariate distributions whose margins' death-set processes have the (homogeneous) Markov property.

* **Corresponding author: Henrik Sloot,** Technical University of Munich, Parkring 11, 85748 Garching-Hochbrück, Germany, e-mail: henrik.sloot@tum.de

The Markov representation allows using simulation algorithms for Markov processes, for example, as described in [4, p. 493], to simulate Marshall–Olkin distributions, given the corresponding Markov generator. However, general models for the entire family of Marshall–Olkin distribution share the *curse of dimensionality*, meaning that the number of parameters grows exponentially with the dimension. Even for low-parametric subclasses, the first row of the Markov generator has $2^d - 1$ off-diagonal parameters, and a simple calculation shows that the generator matrix has up to $3^d - 1$ nonzero parameters. Considering that the maximum index value for implementations of statically typed programming languages is often one of the large unsigned integer values $2^{16} - 1$, $2^{32} - 1$, or $2^{64} - 1$, the difficulties of implementing such a model in high dimensions become evident.

A large number of parameters also poses a methodological issue, reducing model interpretability and risking over-parametrization. A specialized simulation algorithm for exchangeable Marshall–Olkin distributions, introduced in [15], addresses this problem by reducing the number of parameters to the dimension. Another specialized simulation algorithm for certain low-parametric subclasses of exchangeable Marshall–Olkin distributions was suggested in [13]. The focus on exchangeable distributions is supported by the hierarchical models presented in [14, Sec. 5.2] and [16, Sec. 4.2], which use the exchangeable subclass as building blocks. However, [15] and [13] leave significant practical issues unaddressed. The first algorithm requires the on-the-fly calculation of submodel parameters, having numerical issues in higher dimensions similar to those discussed in Section 5. The second algorithm requires the simulation of Lévy processes; this can only be done without bias for the compound Poisson case and requires specialized sampling algorithms for the corresponding jump distribution. In particular, both algorithms have limited applicability to extendible Marshall–Olkin distributions in higher dimensions without further research or specialized implementations of the Lévy processes' jump distributions.

We asked ourselves: *Can we find low-parametric Markov-based models for extendible Marshall–Olkin distributions with a numerically stable implementation?*

This article is structured as follows: Section 2 recalls prerequisites about the Marshall–Olkin, exchangeable Marshall–Olkin, and extendible Marshall–Olkin distributions, including Bernstein functions. In particular, we present existing simulation algorithms and provide several extendible parametric subfamilies. Following, we provide an alternative proof that the death-set process of Marshall–Olkin distributed random vectors is Markov in Section 3, which allows identifying the corresponding Markov generator matrix. In Section 4, we derive the first primary result of this article. It proves that the death-counting process of exchangeable Marshall–Olkin distributed random vectors is Markov. Furthermore, it shows that we can construct an exchangeable Marshall–Olkin distributed random vector from a sample of the corresponding death-counting process by applying an independent random shuffling. Section 5 discusses the second primary result, a numerically stable calculation of the death-counting process' Markov generator Matrix of extendible Marshall–Olkin distributions using integral representations. Finally, before concluding this article, Section 6 benchmarks the runtime of various simulation algorithms for extendible Marshall–Olkin distributions against the algorithm using our new Markov death-counting model.

2 Background

This section recalls the required background information on the Marshall–Olkin distribution. In particular, we discuss the exchangeable and extendible subclass and compare existing stochastic models.

We assume the reader is familiar with stochastic vectors and processes, particularly exchangeability and Markov processes. We also presume some familiarity with the Marshall–Olkin distribution, and we will only discuss the material required in later sections for deriving further results in the following. For the interested reader, we refer to [17] for a detailed discussion of random vectors and Marshall–Olkin distributions, [4] for Markov processes, and [1] for exchangeable random variables. For Poisson and Lévy processes, we refer to [25] and [27], respectively. Finally, we refer to [9] for numerical mathematics and numerical integration. Throughout this article, we use Δ to denote the (*finite*) *forward difference* on a sequence or a

function, i.e., $\Delta x_n := x_{n+1} - x_n$ and $\Delta f(x) = f(x+1) - f(x)$, and Δ^i to denote the (finite) forward differences of the order i.

2.1 Marshall–Olkin distributions

The exponential distribution takes a central position among positive, continuous univariate distributions due to its *lack of memory* (*LOM*) *property*: As a consequence of Cauchy's exponential functional equation, the exponential distribution is the unique continuous distribution on the nonnegative half-line for which knowledge of survival up to a given time does not change the law of the *excess time distribution*. In mathematical notation, we write:

$$\mathbb{P}(\tau > t + s | \tau > t) = \mathbb{P}(\tau > s), \quad t, s \geq 0. \tag{1}$$

This property has a fundamental parallel: the *death-indicator process*, defined by $t \in [0, \infty) \mapsto \mathbb{1}_{\{\tau \leq t\}}$, is a (homogeneous) Markov process in the finite state space $\{0, 1\}$. Hence, the LOM property corresponds to the distribution of the time-to-death being independent of the already elapsed time.

The Marshall–Olkin (MO) distribution was introduced in [19] as the unique multivariate exponential distribution whose margins fulfill the following generalization of the univariate LOM property: Given the joint survival until a particular time, the excess time distribution equals the original distribution. In mathematical notation, we write

$$\mathbb{P}(\boldsymbol{\tau}_I > \boldsymbol{s}_I + t | \boldsymbol{\tau}_I > t) = \mathbb{P}(\boldsymbol{\tau}_I > \boldsymbol{s}_I), \quad \boldsymbol{s}_I := (s_i : i \in I), \quad t \geq 0, \quad I \subseteq [d] := \{1, \ldots, d\}. \tag{2}$$

It was shown in [5] that this property also has a parallel: the marginal death-set processes of MO distributed random vectors $\boldsymbol{\tau}$, defined for $\varnothing \neq I \subseteq [d]$ by $t \in [0, \infty) \mapsto \{i \in I : \tau_i \leq t\}$, are (homogeneous) Markov processes.

MO distributions have for $2^d - 1$ nonnegative parameters $\boldsymbol{\lambda} = \{\lambda_I : \varnothing \neq I \subseteq [d]\}$, subsequently called *shock-arrival intensities*, the survival function

$$\mathbb{P}(\boldsymbol{\tau} > \boldsymbol{t}) = \exp\left\{ - \sum_{\varnothing \neq I \subseteq [d]} \lambda_I \cdot \max_{i \in I} t_i \right\}, \quad \boldsymbol{t} \geq 0,$$

where $\sum_{I : I \ni i} \lambda_I > 0 \ \forall i \in [d]$. The latter condition ensures that all components of $\boldsymbol{\tau}$ are almost surely finite and can be relaxed if this is not required.

In the following, we present three stochastic models for MO distributions. Our intention for this is twofold: First, we want to use these representations to derive different results and properties. Second, we want to implement these models for the runtime comparison in Section 6.

The exogenous shock model

A natural stochastic model for MO distributions known as the exogenous shock model (ESM) was proposed in [19]. It assumes independent exponential random variables, called *shock-arrival times*, with respective shock-arrival intensities. Subsequently, it defines the elements of the random vector as the respective minima of component-related shock-arrival times.

Theorem 2.1. [19] *Let $\{E_I : \varnothing \neq I \subseteq [d]\}$ be independent exponential random variables with nonnegative rates $\boldsymbol{\lambda} = \{\lambda_I : \varnothing \neq I \subseteq [d]\}$ and define the vector $\boldsymbol{\tau}$ by*

$$\tau_i := \min\{E_I : I \ni i\}, \quad i \in [d]. \tag{3}$$

Then, $\boldsymbol{\tau}$ has an MO distribution with shock-arrival intensities $\boldsymbol{\lambda}$.

The ESM is a straightforward stochastic model; theoretical aspects of its implementation are discussed in [17, Algo. 3.1]. The aforementioned reference makes two critical findings: First, the index sets $\varnothing \neq I \subseteq [d]$ have the binary representation equal to $(\mathbb{1}_{\{i \in I\}}, i \in [d]) \in \{0, 1\}^d$. Second, the total number of shock-arrival times, equal to the number of parameters $2^d - 1$, grows exponentially and quickly becomes an issue for every implementation.

The Arnold model

Another stochastic model for MO distributions, subsequently called the Arnold model (AM), was derived in [2]. It defines the random vector $\boldsymbol{\tau}$ as first hitting times onto the components of a sequence of iid set-valued random variables with iid exponentially distributed inter-arrival times. The shock-arrival probabilities are proportional to the shock-arrival intensities, and the inter-arrival rate equals the sum over all shock-arrival intensities.

Theorem 2.2. [2, Sec. 4] and [17, Lem. 3.4] *Let $\{W_j : j \in \mathbb{N}\}$ be iid exponential random variables with rate $\lambda > 0$ and, independent thereof, $\{Y_j : j \in \mathbb{N}\}$ be iid discrete random variables with values in the power set of $[d]$ and probability vector $\boldsymbol{p} = \{p_I : \varnothing \neq I \subseteq [d]\}$. Furthermore, define the vector $\boldsymbol{\tau}$ by*

$$\tau_i \coloneqq \sum_{j=1}^{\min\{k \in \mathbb{N} : Y_k \ni i\}} W_j, \quad i \in [d]. \tag{4}$$

Then, $\boldsymbol{\tau}$ has an MO distribution with shock-arrival intensities $\boldsymbol{\lambda} \coloneqq \lambda \cdot \boldsymbol{p}$.

An interesting aspect of this model proven in [2, Sec. 4] is that the ESM is recovered as follows: Assume that $\boldsymbol{\tau}$ has the representation in Eq. (4) and define

$$E_I \coloneqq \sum_{j=1}^{\min\{k \in \mathbb{N} : I = Y_k\}} W_j, \quad \varnothing \neq I \subseteq [d].$$

Then, $\{E_I : \varnothing \neq I \subseteq [d]\}$ are independent with rates $\{\lambda_I : \varnothing \neq I \subseteq [d]\}$ and $\boldsymbol{\tau}$ has the ESM representation from Eq. (3). Thus, the AM samples the shock-arrival times of the ESM in ascending order by sampling their inter-arrival times and corresponding affected components until all components are dead. However, two distinctions between both models are crucial for their comparison: First, the AM allows us to ignore all shocks arriving after the time when all components are dead, requiring potentially considerably less than $2^d - 1$ shocks. Second, the AM can have *looping steps*, whose shocks Y_j contain only already dead elements. The looping steps have no counterpart in the ESM.

Theoretical aspects of this model's implementation are discussed in [17, Algo. 3.3]. A nontrivial choice of any implementation is the sampling method for iid sequences of discrete laws. The aforementioned reference proposes a version of the probability integral transform for discrete random variables with a divide-and-conquer search for the (left) inversion of the probability distribution function. We suggest considering the *alias method* proposed by [31] since, after an initial parameter transformation, it only requires a uniform discrete sample, a uniform real sample, two reference operations, and a comparison per sample; see also Appendix E.

The Markov death-set model

Finally, a last stochastic model for MO distributions is the Markov death-set model (MDSM). For every death-set process $t \in [0, \infty) \mapsto Z_t \coloneqq \{i \in [d] : \tau_i \leq t\}$, we have

$$\tau_i = \inf\{t \geq 0 : i \in Z_t\}, \quad i \in [d].$$

Recall that the death-set process of an MO distributed random vector is Markov. Consequently, given the Markov generator matrix $Q \in \mathbb{R}^{2^d \times 2^d}$ of Z, where the row indices correspond to the binary representation of the states, we can directly sample Z using models for finite-state Markov processes as follows.

Theorem 2.3. [4, p. 493] *Let $Q = (q_{ij})_{i,j\in[n]} \in \mathbb{R}^{n\times n}$ be a Markov generator matrix and let $q_i := -q_{ii}$ for $i \in [n]$. Define the transition matrix $K = (k_{ij})_{i,j\in[n]}$ as follows*:

$$k_{ij} = \begin{cases} \dfrac{q_{ij}}{q_i} & i \neq j, \\ 0 & i = j \quad \text{and} \quad q_i > 0, \\ 1 & \text{else}. \end{cases}$$

Let $\tilde{Z}$ be a discrete Markov chain with transition matrix K and the inter-arrival times of the transitions $\{W_k : k \in \mathbb{N}\}$, conditioned on $\tilde{Z}$, be independent exponentially distributed with rates $\{q_{\tilde{Z}_{k-1}} : k \in \mathbb{N}\}$. Then, Z, defined in the following, is a continuous Markov process with generator matrix Q:

$$Z_t = \tilde{Z}_{\max\left\{k\in\mathbb{N}_0 : \sum_{l=1}^{k} W_l \leq t\right\}}, \quad t \in [0, \infty).$$

We call $\tilde{Z}$ the embedded Markov chain of Z.

Given death-set process Z has the representation from Theorem 2.3 with $n = 2^d$, we conclude that,

$$\tau_i = \inf\{t \geq 0 : i \in Z_t\} = \sum_{j=1}^{\min\{k\in\mathbb{N} : \tilde{Z}_k \ni i\}} W_j, \quad i \in [d].$$

However, to use this algorithm to sample from MO distributions, we require the mapping $\boldsymbol{\lambda} \to Q$, which is not described in the literature to the best of our knowledge.

2.2 Exchangeable Marshall–Olkin distributions

The general MO distribution, as previously discussed, has the problem of an exponentially increasing number of parameters. We can alleviate the issue by restricting ourselves to elements of the exchangeable subclass, whose number of parameters is equal to the dimension. An MO distribution is exchangeable if and only if its shock-arrival intensities are equal among those that represent shocks of the same cardinality, see [14, Lem. 3.1.1], i.e.,

$$|I| = |J| \;\Leftrightarrow\; \lambda_I = \lambda_J \quad \forall \varnothing \neq I,\ J \subseteq [d].$$

For exchangeable Marshall–Olkin (exMO) distributions, we write for their *exchangeable shock-arrival intensities* $\lambda_i \equiv \lambda_I$, $i = |I|$, and define the *exchangeable shock-size-arrival intensities* $\boldsymbol{\eta} = \{\binom{d}{i}\lambda_i : i \in [d]\}$, i.e.,

$$\eta_i = \sum_{\varnothing \neq I \subseteq [d] : |I| = i} \lambda_I, \quad i \in [d],$$

is the rate of $\min\{E_I : |I| = i\}$, the smallest *shock-arrival time* E_I of all shocks I with cardinality i.

2.3 Extendible Marshall–Olkin distributions

A restriction to the exchangeable subclass dramatically reduces the number of parameters from $2^d - 1$ to d. However, the number is still sizeable for higher dimensions. For this reason, many examples of high-dimensional MO distributions are based on low-parametric, extendible subfamilies. Extendible subfamilies of the MO distribution have a stochastic representation as the finite margin of a countably infinite exchangeable sequence with MO-distributed margins. We will not discuss the details of extendible

Marshall–Olkin (extMO) distributions and exchangeable sequences in this article and refer to [14] and [1] for the details on these; however, we require some results and will recall them in the following.

Before discussing extMO distributions, we have to make a short excursus into *Bernstein functions*. A *Bernstein function* is a nonnegative, nondecreasing function $\psi : [0, \infty) \to [0, \infty)$ that has infinitely many derivatives with alternating signs. An extensive monograph on those functions is [28]. We highlight four important properties:

- Every Bernstein function ψ is uniquely linked to a *Lévy–Khintchine triplet* (a, b, ν) such that
$$\psi(x) = a + bx + \int_0^\infty (1 - e^{-ux})\nu(\mathrm{d}u), \quad x \geq 0, \tag{5}$$
where $a, b \geq 0$ and ν is a *Lévy measure* on $(0, \infty)$, i.e., $\int_0^\infty (1 \wedge u)\mathrm{d}u < \infty$, see [28, Thm. 3.2].
- In combination with [3, Prop. 6.12], the previous property implies that a nonnegative function $\psi : [0, \infty) \to [0, \infty)$ is a Bernstein function if and only if it is nondecreasing and the (negative) finite forward differences are of alternating signs, i.e.,
$$(-1)^{i-1}\Delta^i\psi(x) \geq 0 \quad \forall x \geq 0, \quad i \in \mathbb{N}.$$
- A Bernstein function whose *Lévy measure* has a completely monotone density with respect to Lebesgue measure, i.e., the density is nonnegative, nonincreasing with infinitely many derivatives of alternating sign, is called a *complete Bernstein function*. Every complete Bernstein function ψ is uniquely linked to a *Stieltjes triplet* (a, b, σ) such that
$$\psi(x) = a + bx + \int_0^\infty \frac{x}{x + u}\sigma(\mathrm{d}u), \quad x \geq 0, \tag{6}$$
where $a, b \geq 0$ and σ is a *Stieltjes measure* on $(0, \infty)$, i.e., $\int_0^\infty (1 + u)^{-1}\sigma(\mathrm{d}u) < \infty$, see [28, Thm. 6.2].
- Bernstein functions are the *Laplace exponents* of (*killed*) *Lévy subordinators*, see [28, Chp. 5]: For each Bernstein function ψ, we can create a probability space supporting a nonnegative, nondecreasing (killed) Lévy process Λ on $[0, \infty]$ such that
$$\mathbb{E}[e^{-x\Lambda_t}] = e^{-t\psi(x)}, \quad x, t \geq 0. \tag{7}$$
Conversely, if Λ is a nonnegative, nondecreasing Lévy process, we can find a Bernstein function ψ such that Eq. (7) holds. Recall that Lévy subordinators are nondecreasing, stochastically continuous, càdlàg processes that start in zero and have stationary and independent increments. For more details, we refer to [27].

Remark. [27, cf. Chp. 4] and [28, Chp. 5] For a Lévy–Khintchine triplet (a, b, ν), we have the representation
$$\Lambda_t = \infty \cdot \mathbb{1}_{\{\varepsilon \leq a \cdot t\}} + b \cdot t + J_t,$$
where ε is a unit exponential random variable and J is a pure-jump Lévy process, independent of ε, with Lévy measure ν. If ν is finite, J is a compound Poisson process with intensity $\nu((0, \infty))$ and jump distribution $B \mapsto \nu(B)/\nu((0, \infty))$. Hence, we call a the *killing rate*, b the *drift*, and, if ν is finite, $\nu((0, \infty))$ the *jump rate*, and $B \mapsto \nu(B)/\nu((0, \infty))$ the *jump distribution*.

The following stochastic model for extMO distributions, called the Lévy frailty model (LFM), outlines the relationship between extMO distributions and Bernstein functions:

Theorem 2.4. [13, Thm. 3.3] and [14, Thm. 3.4.1] *There exists a bijection between the distributions of extMO sequences and Bernstein functions. Moreover, given a Lévy subordinator Λ with Bernstein function ψ and, independent thereof, unit exponential barrier values $\{E_i : i \in [d]\}$, we can define*
$$\tau_i := \inf\{t \geq 0 : \Lambda_t \geq E_i\}, \quad i \in [d]. \tag{8}$$

Then, $(\tau_1, \ldots, \tau_d)^\top$ *has an extMO distribution with exchangeable shock-arrival intensities*

$$\lambda_i = (-1)^{i-1}\Delta^i\psi(d-i), \quad i \in [d]. \tag{9}$$

The nontrivial mapping from Bernstein functions to exchangeable shock-arrival intensities in Eq. (9) already indicates that the parametrization via Bernstein functions, albeit methodologically convenient, imposes challenges on the implementation.

2.4 Examples of parametrized families

This section introduces selected Bernstein function families that we implemented for our simulation studies. An extensive list of additional Bernstein functions is compiled in [28, Chp. 16]; whenever possible, we provide the corresponding number of Bernstein functions in that list.

- *Armageddon shock*: Let $\alpha, \beta \geq 0$ and consider the Lévy triplet $(\beta, \alpha, 0)$ corresponding to the Bernstein function $\psi(x) = \beta + \alpha x$, $x \geq 0$. The name reflects that the corresponding ESM representation has only individual shocks, arriving with rate α, and a global shock, killing all components, arriving with rate β.
- *Poisson*: Let $\eta > 0$ and consider the Lévy triplet $(0, 0, \delta_\eta)$ corresponding to the Bernstein function $\psi(x) = (1 - e^{-\eta x})$, $x \geq 0$, where $B \mapsto \delta_\eta(B)$ is the Dirac measure. This family corresponds to Lévy subordinators that are compound Poisson processes with intensity 1 and deterministic jump size η.
- *Exponential* ([28, Chp. 16, No. 4]): Let $\eta > 0$ and consider the Lévy triplet $(0, 0, \nu)$ with a completely monotone Lévy density $\nu(\mathrm{d}u) = \eta \cdot e^{-\eta u}\,\mathrm{d}u$ and Stieltjes measure $\sigma = \delta_\eta$ corresponding to the Bernstein function $\psi(x) = \frac{x}{x+\eta}$, $x \geq 0$. This family corresponds to Lévy subordinators that are compound Poisson processes with intensity 1 and exponential jumps with rate η.
- *Pareto*: Let $\alpha \in (0, 1)$, $x_0 > 0$ and consider the Lévy triplet $(0, 0, \nu)$ with Lévy density $\nu(\mathrm{d}u) = \alpha x_0^\alpha \cdot u^{-\alpha-1}\mathbb{1}_{\{u>x_0\}}\,\mathrm{d}u$. This family corresponds to compound Poisson processes with intensity 1 and jumps from a Pareto distribution. The Pareto family is used in [7, Sec. 5.3] to approximate the subsequently defined *Alpha-stable family*.
- *Alpha-stable* ([28, Chp. 16, No. 1]): Let $\alpha \in (0, 1)$ and consider the Lévy triplet $(0, 0, \nu)$ with a completely monotone Lévy density $\nu(\mathrm{d}u) = \alpha[\Gamma(1-\alpha)]^{-1} \cdot u^{-1-\alpha}\mathrm{d}u$ and Stieltjes density $\sigma(\mathrm{d}u) = \pi^{-1}\sin(\alpha\pi) \cdot u^{\alpha-1}\mathrm{d}u$ corresponding to the Bernstein function $\psi(x) = x^\alpha$, $x \geq 0$. This family corresponds to Lévy subordinators with infinite activity.
- *Gamma* ([28, Chp. 16, No. 26]): Let $a > 0$ and consider the Lévy triplet $(0, 0, \nu)$ with a completely monotone Lévy density $\nu(\mathrm{d}u) = e^{-au}u^{-1}\mathrm{d}u$ and Stieltjes density $\sigma(\mathrm{d}u) = u^{-1}\mathbb{1}_{(a,\infty)}(u)\mathrm{d}u$ corresponding to the Bernstein function $\psi(x) = \log(1 + x/a)$, $x \geq 0$. This family corresponds to Lévy subordinators with infinite activity.
- *Inverse Gaussian* ([17, p. 309] and [28, Chp. 16, No. 2]): Let $\eta \geq 0$ and consider the Lévy triplet $(0, 0, \nu)$ with a completely monotone Lévy density $\nu(\mathrm{d}u) = [\sqrt{2\pi u^3}\,]^{-1}e^{-\frac{1}{2}\eta^2 u}\,\mathrm{d}u$ and Stieltjes density $\sigma(\mathrm{d}u) = \sin(\pi/2)[\pi u]^{-1}\sqrt{2u - \eta^2}\,\mathbb{1}_{(\eta^2/2,\infty)}(u)\mathrm{d}u$ corresponding to the Bernstein function $\psi(x) = \sqrt{2x + \eta^2} - \eta$, $x \geq 0$. This family corresponds to Lévy subordinators with infinite activity.

It can be practical to normalize a Bernstein function ψ to ψ^* such that $\psi^*(1) = 1$: we can do that in multiple ways, for example, by

- *adding a constant part* with $\psi^*(x) = [1 - \psi(1)] + \psi(x)$, $x > 0$,
- *adding a linear part* with $\psi^*(x) = [1 - \psi(1)]x + \psi(x)$, $x > 0$, or
- *scaling* with $\psi^*(x) = \psi(1)^{-1} \cdot \psi(x)$, $x > 0$.

Note that the first two normalization methods require $\psi(1) \leq 1$.

2.5 Hierarchical Marshall–Olkin distributions

We want to conclude the background section with a short note on hierarchical Marshall–Olkin (hMO) distributions. There are multiple approaches to define nonexchangeable hMO distributions with exMO distributions as building blocks, e.g., [14, Sec. 5.2] or [16, Sec. 4.2]. An approach similar to the latter has been proposed in [30, Sec. 4.1]. We do not discuss these representations further in this article. Still, we want to highlight a consequence of their existence: *efficient sampling algorithms for hMO distributions can be built using efficient sampling algorithms for extMO distributions as building blocks*. For this reason, we decided to place our primary focus on the development of such an algorithm for the extendible subclass.

3 The Markovian MO death-set model

Similar to [5], the following section proves that the death-set processes Z of MO distributions are Markovian. In addition, we also provide a formula for the infinitesimal generator matrix Q. This explicit formula is used to analyze the slightly more complicated death-counting process of the exchangeable subclass in Section 4.

Theorem 3.1. *Let $d \geq 2$.*

(*a*) *Let Z be the death-set process of a d-dimensional MO-distributed random vector $\boldsymbol{\tau}$ with shock-arrival intensities $\boldsymbol{\lambda}$. Then, Z is a continuous-time, homogeneous Markov process on the power set of $[d]$ with infinitesimal generator matrix $Q = (q_{IJ} : I, J \subseteq [d])$ defined by*

$$q_{IJ} \coloneqq \begin{cases} -\sum\limits_{I \subsetneq K \subseteq [d]} \sum\limits_{L \subseteq I} \lambda_{L \cup (K \setminus I)} & I = J, \\ \sum\limits_{L \subseteq I} \lambda_{L \cup (J \setminus I)} & I \subsetneq J, \\ 0 & \text{else}. \end{cases} \tag{10}$$

(*b*) *Let Z be a continuous-time, homogeneous Markov process on the power set of $[d]$ with infinitesimal generator matrix Q as in Eq. (10) and define the random vector $\boldsymbol{\tau}$ as follows:*

$$\tau_i \coloneqq \inf\{t > 0 : i \in Z_t\}, \quad i \in [d]. \tag{11}$$

Then, $\boldsymbol{\tau}$ is MO distributed with shock-arrival intensities $\boldsymbol{\lambda}$, and Z is the death-set process of $\boldsymbol{\tau}$.

The infinitesimal generator matrix Q in Eq. (10) often takes a simpler form for MO distribution subclasses. Appendix A simplifies the infinitesimal generator Eq. (10) for exMO distributions, extMO distributions, and the Armageddon-shock family.

Nevertheless, the representation in Eq. (10) imposes enormous challenges for possible implementations of the MDSM: the implementation either stores the generator matrix in memory or calculates its entries on-the-fly during the simulation. We argue that both options are hardly feasible in higher dimensions: On the one hand, a straightforward but tedious calculation yields up to $3^d - 1$ nonzero entries of Q. Consequently, storing this matrix is nontrivial for larger d, even using sparse matrix designs. On the other hand, using on-the-fly calculations requires repeatedly populating large vectors and prevents utilizing optimizations for discrete sampling that require nontrivial setup activities.

Before proving the result, we highlight that the MDSM does not translate well into a simulation algorithm in higher dimensions for two reasons: First, implementing the model becomes challenging because of the exponentially growing number of parameters. Second, nontrivial MO distributions are often specified via a Bernstein function, using Eq. (9), often from a low-parametric family. Consequently, higher-order iterated differences and large sums make calculating the generator matrix numerically challenging, compared with Section 5.

The proof of Theorem 3.1 is separated into three steps: First, we identify the shock-arrival times of the AM with a Poisson process. Second, we show that the sequence of cumulative unions of shock sets from the AM is a random walk on the semigroup of the power set of $[d]$ with the union as a conjunction. Third, we conclude that this makes the death-set indicator process a random walk subordinated by a Poisson process; hence, it is also a Markov process. This representation makes it simple to derive the corresponding infinitesimal generator and embedded transition matrix.

Lemma 3.2. [25, Ex. 3.3.7] *Let $W_1, W_2, \dots$ be iid exponentially distributed with rate $\lambda > 0$. Then, the set of their cumulative sums $\Gamma_1, \Gamma_2, \dots$, defined by*

$$\Gamma_j \coloneqq \sum_{k=1}^{j} W_k, \quad j \in \mathbb{N},$$

is a (homogeneous) Poisson point process with intensity λ and defines the (homogeneous) Poisson process N with intensity λ by

$$N : [0, \infty) \to \mathbb{N}_0, \quad t \mapsto \sum_{j=1}^{\infty} \mathbb{1}_{\{\Gamma_j \le t\}}.$$

Lemma 3.3. *Let $d \ge 2$, $\boldsymbol{p} = \{p_I : I \subseteq [d]\}$ be a probability vector for the iid sequence $Y_1, Y_2, \dots$ on the power set of $[d]$, and define the discrete-time process*

$$\tilde{Z} : \mathbb{N}_0 \to \mathcal{P}([d]) \coloneqq \{I : I \subseteq [d]\}, \quad n \mapsto \bigcup_{j=1}^{n} Y_j,$$

where $\bigcup_{j=1}^{0} Y_j \coloneqq \varnothing$. Then, $\tilde{Z}$ is a discrete-time Markov chain with transition matrix $K = (k_{IJ} : I, J \subseteq [d])$ defined by

$$k_{IJ} \coloneqq \begin{cases} \sum_{L \subseteq I} p_{L \cup (J \setminus I)} & I \subseteq J, \\ 0 & \text{else}. \end{cases} \tag{12}$$

Proof. Note that $\tilde{Z}$ is a random walk on the finite semigroup $(\mathcal{P}([d]), \cup)$ and consequently also a Markov chain. Let $k \in \mathbb{N}$ and $I, J \subseteq [d]$. First, since $\tilde{Z}_0, \tilde{Z}_1, \dots$ is a nondecreasing sequence of (random) sets, we have

$$\mathbb{P}(\tilde{Z}_{k+1} = J | \tilde{Z}_k = I) = 0,$$

whenever $I \cap J \neq I$. Second, we have for $I \subseteq J$ that

$$\mathbb{P}(\tilde{Z}_{k+1} = J | \tilde{Z}_k = I) = \mathbb{P}(Y_{k+1} \setminus I = J \setminus I) = \mathbb{P}(Y_1 \setminus I = J \setminus I) = \sum_{L \subseteq [d] : L \setminus I = J \setminus I} p_L = \sum_{L \subseteq I} p_{L \cup (J \setminus I)}. \qquad \square$$

Lemma 3.4. *Let $d \ge 2$, N be a (homogeneous) Poisson process with intensity $\lambda > 0$ and, independent thereof, let $\tilde{Z}$ be the Markov chain from* Lemma 3.3 *for probability vector $\boldsymbol{p}$. Define the continuous-time process*

$$Z : [0, \infty) \to \mathcal{P}([d]), \quad t \mapsto \bigcup_{j=1}^{N_t} Y_j.$$

Then, Z is a continuous-time Markov process with infinitesimal generator matrix $Q = (q_{IJ} : I, J \subseteq [d])$ as in Eq. (10) *with $\boldsymbol{\lambda} = \lambda \cdot \boldsymbol{p}$.*

Proof. Note that, due to Lemma 3.3, $\tilde{Z}$ is a Markov chain with transition matrix $K = (k_{IJ} : I, J \subseteq [d])$ defined in Eq. (12). Hence, Z represents a discrete-time Markov chain with transition matrix K, subordinated by a homogeneous Poisson process with intensity λ. We conclude with [4, Expl. 13.2.8] that Z is a Markov process with the infinitesimal generator matrix $Q = \lambda \cdot (K - \mathbb{1})$, where $\mathbb{1}$ is the suitable matrix identity. While the second and third cases of Eq. (10) follow trivially, we obtain the first case from the following calculation:

$$q_{II} = \lambda \cdot (k_{II} - 1) = -\lambda \cdot (1 - k_{II}) = -\lambda \cdot \left[\sum_{I \subseteq L \subseteq [d]} k_{IL} - k_{II} \right] = -\lambda \cdot \sum_{I \subsetneq L \subseteq [d]} k_{IL} = - \sum_{I \subsetneq K \subseteq [d]} \sum_{L \subseteq I} \lambda_{L \cup (K \setminus I)}. \qquad \square$$

Proof of Theorem 3.1. We can assume for both parts without loss of generality that $\boldsymbol{\tau}$ and Z are defined by the AM, compared with Theorem 2.2.

First, for Theorem 3.1(a), the claim becomes a trivial corollary of Lemma 3.4 after considering the AM representation of $\boldsymbol{\tau}$ and subsequently Z.

Second, for Theorem 3.1(b), also considering the AM representation of Z, we have the following calculation:

$$\begin{aligned} \tau_i &= \inf\{t > 0 : i \in Z_t\} = \inf\left\{ t > 0 : i \in \bigcup_{j=1}^{N_t} Y_j \right\} = \inf\left\{ t > 0 : \sum_{j=1}^{N_t} \mathbb{1}_{\{i \in Y_j\}} > 0 \right\} \\ &= \inf\left\{ t > 0 : \sum_{j=1}^{\infty} \mathbb{1}_{\{i \in Y_j, \quad W_1 + \cdots + W_j \le t\}} > 0 \right\} = \inf\left\{ t > 0 : \sum_{j=1}^{\min\{k \in \mathbb{N} : i \in Y_k\}} W_j \le t \right\} = \sum_{j=1}^{\min\{k \in \mathbb{N} : i \in Y_k\}} W_j. \end{aligned}$$

We obtain the claim by comparing the aforementioned equation with Eq. (4). $\square$

4 The Markovian exMO death-counting model

This section shows that the *death-counting process* of exMO distributions inherits the Markov property from the death-set process introduced in Section 3. Furthermore, we derive the Markov death-counting model (MDCM) for exMO distributions. This model requires significantly fewer parameters than the MDSM.

We define the *death-counting process* of a d-variate random vector $\boldsymbol{\tau}$ with death-set process Z by

$$Z^* : [0, \infty) \to [d]_0 := [d] \cup \{0\}, \quad t \mapsto |Z_t| = |\{i : \tau_i \le t\}|.$$

In the case of MO distributions, we again assume an AM representation. Hence, we have

$$Z_t^* = \left| \bigcup_{j=1}^{N_t} Y_j \right|, \quad t \ge 0.$$

The following result shows that the death-counting process of exMO distributions is Markov. Moreover, it shows that exMO distributions are represented by shuffled transition times of suitable Markovian death-counting processes.

Theorem 4.1. *Let $d \ge 2$.*

(*a*) *Let Z^* be the death-counting process of a d-dimensional exMO distributed random vector with exchangeable shock-arrival intensities $\boldsymbol{\lambda}$. Then, Z^* is a continuous-time, homogeneous Markov process on $[d]_0$ with infinitesimal generator matrix $Q^* = (q_{ij}^* : i, j \in [d]_0)$ defined by*

$$q_{ij}^* = \begin{cases} -\sum_{l=1}^{d-i} \binom{d-i}{l} \sum_{k=0}^{i} \binom{i}{k} \lambda_{k+l} & i = j, \\ \binom{d-i}{j-i} \sum_{k=0}^{i} \binom{i}{k} \lambda_{k+(j-i)} & i < j, \\ 0 & \text{else}. \end{cases} \tag{13}$$

(*b*) *Let Z^* be a continuous-time, homogeneous Markov process on $[d]_0$ with infinitesimal generator matrix Q^* as in Eq. (13). Furthermore, let Π, independent thereof, be a uniform random permutation on $[d]$ and define the random vector $\boldsymbol{\tau}$ by*

$$\tau_i := \inf\{t > 0 : \Pi(i) \le Z_t^*\}, \quad i \in [d]. \tag{14}$$

Then, $\boldsymbol{\tau}$ has an exMO distribution with exchangeable shock-arrival intensities $\boldsymbol{\lambda}$.

Again, the infinitesimal generator matrix Q^* in Eq. (13) often takes a simpler form for exMO distribution subclasses. Appendix B simplifies the infinitesimal generator Q^* for a reparametrization of exMO distributions, extMO distributions, and the Armageddon shock family.

The MDSM's generator matrix has 4^d entries and up to $3^d - 1$ nonzero entries. In contrast, the MDCM has only $(d+1)^2$ entries and up to $[d+1]d/2 + d$ nonzero entries. Two immediate consequences of this reduction of model parameters are as follows: First, the parameter number grows only quadratically instead of exponentially with the dimension. Second, the parameter number does not exceed technical maximums of integer binary representations for reasonable dimension sizes.

We also separate the proof of Theorem 4.1 into several incremental steps: First, we show that the sequence of death-count transitions from the AM is a discrete-time Markov chain. Second, we conclude that the death-count process is Markov and calculate its infinitesimal generator. Third, we use that the law of exchangeable random vector remains the same after ordering and subsequently shuffling its components uniform at random. Therefore, while we cannot recover the original random vector from the death-counting process alone, we can construct another random vector with the same law by applying a uniform random permutation.

Lemma 4.2. *Let $d \geq 2$, $\boldsymbol{p} = \{p_I : I \subseteq [d]\}$ be an exchangeable probability vector for the exchangeable iid sequence $Y_1, Y_2, \ldots$ on the power set of $[d]$, and define the discrete-time process*

$$\tilde{Z}^* : \mathbb{N}_0 \to [d]_0, \quad n \mapsto |\tilde{Z}_n| = \left| \bigcup_{j=1}^{n} Y_j \right|.$$

Then, $\tilde{Z}^*$ is a discrete-time Markov chain with transition matrix $K^* = (k^*_{ij} : i, j \in [d]_0)$ defined by

$$k^*_{ij} = \binom{d-i}{j-i} \cdot \begin{cases} \sum_{l=0}^{i} \binom{i}{l} p_{l+(j-i)} & i \leq j, \\ 0 & \text{else}. \end{cases} \tag{15}$$

Proof. We know from Lemma 3.3 that $\tilde{Z}$ is a Markov chain with transition matrix K from Eq. (12). We perform a simple calculation using exchangeability, grouping shock-arrival intensities associated with the same cardinality, to obtain

$$k_{IJ} = \sum_{l=0}^{i} \binom{i}{l} p_{l+(j-i)} =: k_{ij}, \quad I \subseteq J, \quad |I| = i, \quad |J| = j,$$

where we write $p_i \equiv p_I$ and denote $k_{ij} \equiv k_{IJ}$ for $|I| = i$ and $|J| = j$.

First, since $\tilde{Z}^*_0, \tilde{Z}^*_1, \ldots$ is a nondecreasing sequence of integer-valued random variables, we have for all sequences $i_1, \ldots, i_{n-1}, i, j$ that are not nondecreasing that

$$\mathbb{P}(\tilde{Z}^*_{n+1} = j | \tilde{Z}^*_n = i, \quad \tilde{Z}^*_l = i_l, \quad \forall l < n) = 0.$$

Second, for nondecreasing sequences of integer values $0 \leq i_1 \leq \cdots \leq i_{n-1} \leq i \leq j$, we denote the set of all nondecreasing set-sequences of the length $n+1$ with corresponding cardinalities $i_1, \ldots, i_{n-1}, i, j$ by $\mathcal{I}(i_1, \ldots, i_{n-1}, i, j)$ and have

$$\begin{aligned}
&\mathbb{P}(\tilde{Z}^*_{n+1} = j | \tilde{Z}^*_n = i, \tilde{Z}^*_l = i_l, \ \forall l < n) \\
&= \frac{\sum_{(I_1, \ldots, I_{n-1}, I, J) \in \mathcal{I}(i_1, \ldots, i_{n-1}, i, j)} \mathbb{P}(\tilde{Z}_{n+1} = J, \quad \tilde{Z}_n = I, \quad \tilde{Z}_l = I_l, \ \forall l < n)}{\sum_{(I_1, \ldots, I_{n-1}, I) \in \mathcal{I}(i_1, \ldots, i_{n-1}, i)} \mathbb{P}(\tilde{Z}_n = I, \quad \tilde{Z}_l = I_l, \ \forall l < n)} \\
&= \frac{\binom{d}{i_1}\binom{d-i_1}{i_2-i_1} \cdots \binom{d-i_{n-1}}{i_n-i_{n-1}}\binom{d-i}{j-i} \cdot k_{0\, i_1} k_{i_1 i_2} \cdots k_{i_{n-1} i} k_{ij}}{\binom{d}{i_1}\binom{d-i_1}{i_2-i_1} \cdots \binom{d-i_{n-1}}{i_n-i_{n-1}} \cdot k_{0\, i_1} k_{i_1 i_2} \cdots k_{i_{n-1} i}} \\
&= \binom{d-i}{j-i} \cdot k_{ij}.
\end{aligned}$$

□

Lemma 4.3. *Let $d \geq 2$, N be a (homogeneous) Poisson process with intensity $\lambda > 0$ and, independent thereof, let $\tilde{Z}^*$ be the Markov chain from* Lemma 4.2 *for an exchangeable probability vector $\boldsymbol{p}$. Define the continuous-time processes*

$$Z^* : [0, \infty) \to [d]_0, \quad t \mapsto \tilde{Z}^*_{N_t} = \left| \bigcup_{j=1}^{N_t} Y_j. \right|$$

Then, Z^* is a continuous-time Markov process with infinitesimal generator matrix $Q^* = (q^*_{ij} : i, j \in [d]_0)$ defined by Eq. (13) *with $\boldsymbol{\lambda} = \lambda \cdot \boldsymbol{p}$.*

Proof. We know from Lemma 4.2 that $\tilde{Z}^*$ is a Markov chain with transition matrix K^* from Eq. (15). Z^* thus represents a discrete-time Markov chain with transition matrix K^*, subordinated by a Poisson process with intensity λ. As in the proof of Lemma 3.4, we use [4, Expl. 13.2.8] to conclude that Z^* is a continuous-time Markov process with infinitesimal generator $Q^* = \lambda \cdot (K^* - \mathbb{1})$, where $\mathbb{1}$ is the suitable matrix identity. While the second and third cases of Eq. (13) follow trivially, we obtain the first case using the following calculation:

$$q^*_{ii} = \lambda \cdot (k^*_{ii} - 1) = -\lambda \left[\sum_{l=i}^{d} k^*_{il} - k^*_{ii} \right] = -\sum_{l=1}^{d-i} \binom{d-i}{l} \sum_{r=0}^{i} \binom{i}{r} \lambda_{r+l}. \qquad \square$$

Proof of Theorem 4.1. We assume without loss of generality that Z^* has a representation of the death-counting process for an exMO distributed random vector $\boldsymbol{\tau}$, which is generated by the AM (see Theorem 2.2). The corresponding death-set process is denoted by Z.

First, for Theorem 4.1(a), the claim becomes a trivial corollary of Lemma 4.3 after considering the AM representation of $\boldsymbol{\tau}$, Z, and Z^*.

Second, for Theorem 4.1(b), let $\hat{\Pi}$ be the (random) permutation of $[d]$ such that

$$\tau_{\hat{\Pi}(1)} \leq \cdots \leq \tau_{\hat{\Pi}(d)}.$$

Then, by using Theorem 3.1, we have

$$\begin{aligned} \tau_{\hat{\Pi}(i)} &= \inf\{t > 0 : \hat{\Pi}(i) \in Z_t\} \\ &= \inf\{t > 0 : \{\hat{\Pi}(j) : j \leq i\} \subseteq Z_t\} \\ &= \inf\{t > 0 : i \leq Z^*_t\}, \quad i \in [d]. \end{aligned}$$

Consequently, we obtain the claim with [17, Lem. 3.8]; the result used states that the law of an exchangeable random vector does not change under a (possibly dependent) reordering (in this case $\hat{\Pi}$) and a subsequent reordering according to an independent uniform permutation (e.g., Π). $\square$

Finally, based on Theorem 4.1, we can sketch a simple simulation algorithm for exMO distributions using the Markov process representation from Theorem 2.3, see Algorithm 1. The algorithm requires simulation algorithms for exponential and (finite-spaced) discrete random variables and a random permutator.

Algorithm 1. Sample from exMO distributions with the MDCM (see Theorem 2.3 and 4.1).

Require: (Vector) arguments `rates` and `probs` with state-dependent parameters for the sampling algorithms `sample_exponential` and `sample_discrete`, which sample from an exponential or discrete distribution (taking values $0, 1, \ldots, k-1$ for probabilities $p_0, p_1, \ldots, p_{k-1}$ and some $k \in \mathbb{N}$). Additionally, a method `shuffle`, which permutates vectors uniform at random.

```
1 : procedure SAMPLE_MDCM(d, rates, probs)
2 :   time = 0
3 :   i = 0
4 :   x = zeros(d)
```

```
    while i < d do
      i0 = i
      time += sample_exponential(rates[i])          ▷ waiting time
      i += 1 + sample_discrete(probs[i])            ▷ additional default-count
      for i0 ≤ j < d do
        x[j] = time
      end for
    end while
    shuffle(x)
    return x
end procedure
```

Remark. [15, Algo. 1] proposed a similar model: First, sample the order-statistic of $\boldsymbol{\tau}$ by recursively sampling shock sizes of marginal models using the representation from Theorem 2.2. Second, apply a permutation that is uniform at random. A close inspection yields that both representations' transition rates and probabilities are equal. From an implementation point of view, corresponding algorithms of both models differ in whether transition rates and probabilities are calculated once or repeatedly on-the-fly during simulation. Note, in particular, that this representation shares all numerical issues of the MDCM regarding mapping the input parameters to simulation parameters, which is discussed in Section 5.

5 Numerically stable approximation of extMO generators

Section 4 derived a novel stochastic representation for exMO distributions based on shuffling *Markovian* death-counting processes' transition times uniformly at random. However, for high-dimensional applications, the possible distributions are often restricted to one or a combination of multiple families of extMO distributions. Hence, we need to calculate the infinitesimal generator matrices of extMO distributions' death-counting processes, denoted by Q^*, to use the aforementioned representation for high-dimensional sampling. This section aims to answer the following question: *How can we approximate Q^* efficiently and numerically stable for a given Bernstein function ψ and dimension $d \geq 2$ to a satisfying accuracy*?

Consider an extMO distribution with Bernstein function ψ and dimension $d \geq 2$. Recall that the exchangeable shock-arrival intensities fulfill

$$\lambda_i = (-1)^{i-1}\Delta^i\psi(d-i), \quad i \in [d]. \tag{9 revisited}$$

Furthermore, we can deduce from Eq. (13) by a tedious but straightforward calculation, see Remark B.2, that

$$q^*_{ij} = \binom{d-i}{j-i} \cdot \begin{cases} -\psi(d-i) & i = j, \\ (-1)^{j-i-1}\Delta^{j-i}\psi(d-j) & i < j, \\ 0 & \text{else}. \end{cases} \tag{16}$$

Hence, a natural, *naïve* method to calculate Q^* is recursively calculating finite forward differences and appropriately changing the signs of the results.

This section shows in a numerical study that this naïve numerical computation of the infinitesimal generator matrix Q^* using Eq. (16) leads to sizeable distortions of transition intensities and probabilities. Subsequently, we propose an improved, staged numerical approximation of Q^*: First, approximate its first row, i.e., the shock-size-arrival intensities, with a numerical integration routine using one of two integral representations of Q^* with nonnegative integrands. Using these representations ensures nonnegativity and allows us to control the approximation's accuracy using a suitable numerical integration method with error bounds. Under weak additional assumptions, these integrals are well suited for numerical integration since their integrands are bounded, except for boundary cases. For the exceptions, we provide suitable alternatives. Second, calculate the remaining rows of Q^* using a simple recursion. Finally, we repeat our study for our proposed approach and demonstrate that it does not suffer from similar distortions as the naïve method.

Special cases with closed-form formulas for Q^*

Before discussing the naïve approach, we present two special cases with a *closed-form* representation of the death-counting process' infinitesimal generator matrix Q^*. We call a representation *closed form* if it can be evaluated quickly with high accuracy, e.g., if it involves only simple algebraic expressions and standard functions such as the exponential, gamma, or beta function. We will use these special cases for the validation of approximations of Q^*.

The first special case is the Poisson family and requires the following auxiliary result, which can be verified via induction.

Lemma 5.1. [3, pp. 92 and 130] *Let* $\psi(x) = 1 - \exp\{-\eta x\}$, $x \geq 0$, $\eta > 0$, *then*

$$(-1)^{i-1}\Delta^i\psi(x) = e^{-\eta x}[1 - e^{-\eta}]^i, \quad x \geq 0, \quad i \in \mathbb{N}.$$

By using this result and Eq. (13), we obtain a closed-form expression of Q^* for the Poisson family:

Proposition 5.2. *Consider an extMO distribution from the Poisson family with Bernstein function* $\psi(x) = [1 - \exp\{-\eta x\}]$, $x \geq 0$, $\eta > 0$. *Then*,

$$q^*_{ij} = \binom{d-i}{j-i} \cdot \begin{cases} -[1 - e^{-\eta(d-i)}] & i = j, \\ e^{-\eta(d-j)}[1 - e^{-\eta}]^{j-i} & i < j, \\ 0 & \textit{else}. \end{cases}$$

The second special case is the Armageddon family, i.e., (almost) affine-linear Bernstein functions with a possible jump after zero. Their generators can be represented in closed form after a straightforward calculation since the second-order finite forward differences of corresponding Bernstein functions are zero, except in zero, see Remark B.3.

Proposition 5.3. *Consider an extMO distribution from the Armageddon family with Beinstein function* $\psi(x) = \beta\mathbb{1}_{\{x>0\}} + \alpha x$, $x \geq 0$, *for* $\alpha, \beta \geq 0$ *with* $\alpha + \beta > 0$. *Then*,

$$q^*_{ij} = \binom{d-i}{j-i} \cdot \begin{cases} -\beta - (d-i)\alpha & i = j < d, \\ \alpha & i+1 = j < d, \\ \alpha + \beta & i+1 = j = d, \\ \beta & i+1 < j = d, \\ 0 & \textit{else}. \end{cases}$$

A simple calculation shows that the mapping $(\psi, d) \mapsto Q^*$ is linear in ψ, particularly, the infinitesimal generator of a convex combination of Bernstein functions is the convex combination of the corresponding infinitesimal generators. Hence, we can separately calculate the infinitesimal generator of the killing and drift part and the infinitesimal generator of the pure-jump part and aggregate them afterward.

Validation of numerical approximations of Q^*

For the cases without a closed form, numerically stable representation of Q^*, any nontrivial approximation for the mapping $(\psi, d) \mapsto Q^*$, especially in higher dimensions d, must be evaluated on its numerical stability. Therefore, we propose two validation criteria:

- A fundamental property of Markov generator matrices is that row sums are equal to zero. Furthermore, the diagonal entries $q^*_{ii} = -\psi(d-i)$, $i \in [d]$, can be calculated with the same accuracy as ψ. Hence, we can measure the total error of transition intensities by comparing the relative differences of the approximations of $-q^*_{ii}$ and $\sum_{j>i} q^*_{ij}$, $i \in [d]_0$: for a numerical approximation $Q^\dagger$ of Q^*, define

$$\varepsilon_1(d) \coloneqq \max_{i\in[d-1]_0} \left| \frac{\sum_{j>i} q^{\dagger}_{ij}}{-q^{\dagger}_{ii}} - 1 \right|.$$

– For families with closed-form formulas, we can calculate Q^* with high accuracy. Consequently, we can compare two numerically approximated versions of Q^*, approximated with the method in question and its closed-form counterpart, respectively, using the subsequently defined metric. For each row i, calculate the total variation distance (TVD) between the approximated transition distributions based on the weights $(\sum_{k>i} q^*_{ik})^{-1} \cdot q^*_{ij}$, $j > i$. In particular, for numerical approximations $Q^{\dagger}$ and $Q^{\ddagger}$ of Q^*, where the latter is calculated with the closed-form representation, define

$$\varepsilon_2(d) \coloneqq \max_{i\in[d-1]_0} \mathrm{TVD}\left(\left\{\frac{q^{\dagger}_{ij}}{\sum_{k>i} q^{\dagger}_{ik}} : j > i\right\}, \left\{\frac{q^{\ddagger}_{ij}}{\sum_{k>i} q^{\ddagger}_{ik}} : j > i\right\}\right).$$

Remark. (Calculating the TVD). Our subsequent numerical studies in `R` use the `TotalVarDist` method for `DiscreteDistribution` classes from the packages `distr` and `distrEx`, see [26], for calculating the TVDs between transition distributions.

The two validation criteria measure the distortions of waiting times and transition distributions. While the second validation criterion is only applicable for those families for which Q^* can be calculated with high accuracy using another method, e.g., the Poisson or Armageddon family, the first validation criterion is applicable for every family.

A critical validation aspect is the choice of test cases. Unfortunately, it is impossible to validate a method for every Bernstein function. Therefore, we need to choose a small subset that is large enough to allow meaningful conclusions about a method's numerical stability for all Bernstein functions. In the following, we outline the reasoning for our subjective selection.

– We consider four families, which can be viewed as *extreme points* of important (convex) subclasses: The *Armageddon family*, representing all (almost) affine-linear Bernstein functions; the *Poisson family*, extreme points of the entire Bernstein function class, see Eq. (5) and [28, Chp. 3]; the *Exponential family*, extreme points of complete Bernstein functions, see Eq. (6) and [28, Chp. 6]; and the *α-stable family*, extreme points of Laplace exponents of *completely self-decomposable* laws, see [28, Chp. 5].
– Using normalization by adding a linear part such that $\psi(1) = 1$, see page 314, all involved families have a unique mapping to (bivariate) lower-tail dependence coefficients (LTDCs) in $(0, 1)$. In particular, the *LTDC* of the bivariate margins is $\mathrm{LTDC} = 2 - \psi(2)/\psi(1)$, see [22, Example 5.21]. Thus, to cover a broad spectrum of each family, we choose the members corresponding to low, mid, and high bivariate lower-tail dependence parameters: $\mathrm{LTDC} \in \{0.05, 0.5, 0.95\}$.

Naïve numerical calculation of Q^*

Equation (16) suggests a naïve numerical calculation of Q^*. For this, set lower-triangular values of Q^* to zero, set the diagonal values to $-\psi(d - i)$, $i \in [d]_0$, and calculate the upper-triangular values for $j > i$ in three steps: First, recursively apply finite forward differences and appropriately change the result's sign to calculate $(-1)^{j-i-1}\Delta^{j-i}\psi(d - j)$ numerically. Second, set these values to zero if they are negative. This modification is necessary because these numerical calculations can be negative, albeit theoretically being nonnegative. Third, multiply the result with the binomial coefficient $\binom{d-i}{j-i}$.

Remark. (Binomial coefficients in `R`, see [24]) Binomial coefficients $\binom{n}{k}$, for nonnegative integers $0 \le k \le n$, have exact representations as 64-bit double-precision binary floating-point numbers, in the following

called *binary64* numbers, see [12], for $n \leq 50$. In R, the method `base::choose` calculates binomial coefficients as a *binary64* number via

$$\binom{n}{k} = \begin{cases} 1 & n = k, \\ n \cdot \prod_{j=2}^{k} \frac{n-j+1}{j} & k < 30, \\ n \cdot \prod_{j=2}^{n-k} \frac{n-j+1}{j} & n-k < 30 \quad \text{and} \quad k \geq 30, \\ \exp\{-\log(n+1) - \operatorname{logBeta}(n-k+1, k+1)\} & \text{else}. \end{cases}$$

Expressions are evaluated from left to right and from inner to outer, and logBeta, the logarithm of the beta function, is evaluated using the dedicated numerical routine `lbeta`. The latter case uses the Beta function's binomial coefficient representation

$$\binom{n}{k} = \frac{\Gamma(n+1)}{\Gamma(k+1)\Gamma(n-k+1)} = \frac{1}{(n+1)\cdot B(n-k+1, k+1)}.$$

The results in Figures 1 and 2 show that the naïve approach produces significant differences for low double-digit dimensions and distorts waiting time intensities and transition distributions beyond recognition for mid double-digit dimensions. More precisely, we observe the following:

- The maximum relative error ε_1 for the sum of transition intensities increases exponentially with the dimension once intensities are floored to zero. The range of dimensions requiring no flooring depends on the specific Bernstein function and parameter choice.
- The maximum TVD ε_2 also increases exponentially with the dimension until the maximum TVD, which is 1.
- Note that for the Armageddon family and LTDC = 50%, all values of ψ on $[d]_0$ have an exact representation as binary64 numbers. Hence, the naïve method calculates Q^* exactly for this exceptional circumstance.

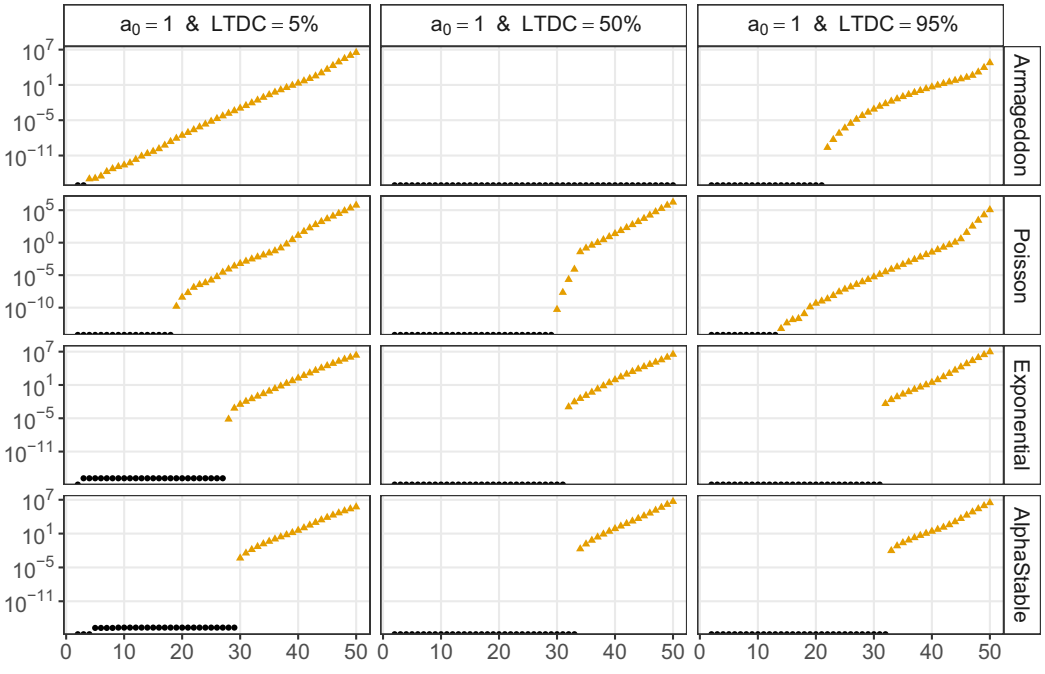

Figure 1: The maximum relative differences $\varepsilon_1(d)$ of the off-diagonal row sums for the naïve approximation of Q^* for $2 \leq d \leq 50$; yellow coloring and triangle shape indicate whether flooring was required.

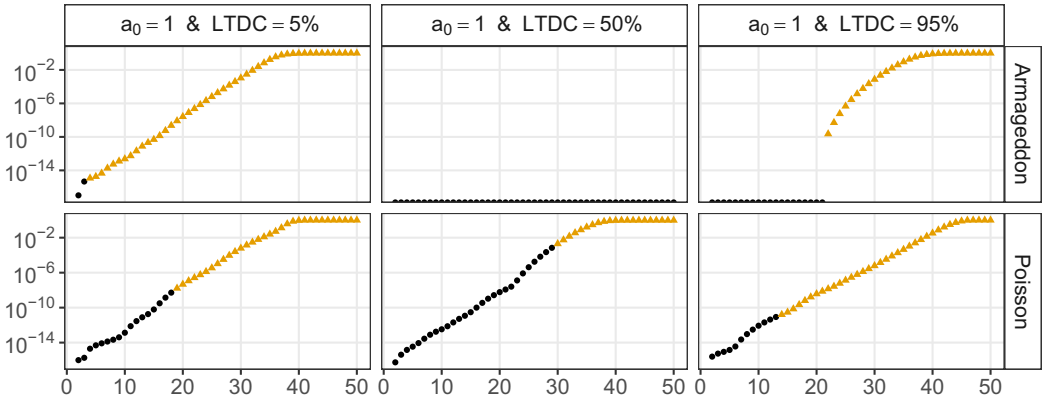

Figure 2: The maximum total variation distance $\varepsilon_2(d)$ of the transition distributions of naïve approximation and high-precision calculation of Q^* for $2 \le d \le 50$; yellow coloring and triangle shape indicate whether flooring was required.

Integral representations of Q^*

Our proposed approximation of Q^* uses two integral representations for the exchangeable *shock-size-arrival intensities*, which will be derived in the following. The idea is simple: Consider a Bernstein function with the representation

$$\psi(x) = \int_I \tilde{\psi}_u(x)\mu(\mathrm{d}u), \quad x \ge 0,$$

where μ is a measure on a set I and $\{\tilde{\psi}_u : u \in I\}$ is a family of Bernstein functions for which the (negative) alternating finite forward differences have a simple, closed-form representation. Note that such a representation exists for every Bernstein function with a drift and killing rate equal to zero due to the *Lévy–Khintchine representation* from Eq. (5) and Lemma 5.1. Then, by using the linearity of the integral, we obtain

$$(-1)^{i-1}\Delta^i\psi(x) = \int_I \underbrace{(-1)^{i-1}\Delta^i\tilde{\psi}_u(x)}_{\ge 0}\mu(\mathrm{d}u), \quad x \ge 0,\ i \in \mathbb{N}.$$

Assuming this integral can be numerically approximated, e.g., because μ is a discrete measure with finite support or if μ has a suitable density with respect to the Lebesgue measure, this representation has the advantage that the approximation is always nonnegative. Furthermore, numerical integration methods, such as Gauss–Kronrod quadrature, allow utilizing error estimates. Consequently, we deem it preferable to approximate directly the integrals

$$q^*_{ij} = \binom{d-i}{j-i} \cdot (-1)^{j-i-1}\Delta^{j-i}\psi(d-j) = \int_I \binom{d-i}{j-i} \cdot (-1)^{j-i-1}\Delta^{j-i}\tilde{\psi}_u(d-j)\mu(\mathrm{d}u).$$

The first integral representation for Q^* uses the *Lévy–Khintchine representation* from Eq. (5) and holds for all Bernstein functions:

Theorem 5.4. *Let $d \ge 2$ and consider a d-variate extMO distribution with Bernstein function ψ and Lévy triplet (a, b, ν) such that Eq. (5) is fulfilled. Then, for $0 \le i < j \le d$, we have*

$$\lambda_i = a\mathbb{1}_{\{i=d\}} + b\mathbb{1}_{\{i=1\}} + \int_0^\infty e^{-u(d-i)}[1-e^{-u}]^i\nu(\mathrm{d}u) \tag{17a}$$

and

$$q^*_{ij} = \binom{d-i}{j-i} \cdot [a\mathbb{1}_{\{j=d\}} + b\mathbb{1}_{\{j-i=1\}}] + \int_0^\infty \binom{d-i}{j-i} \cdot e^{-u(d-j)}[1-e^{-u}]^{j-i}\nu(\mathrm{d}u). \tag{17b}$$

Proof. The result is a direct corollary of Eq. (9), Eq. (16), Lemma 5.1, and Proposition 5.3. Note that the formula for the exchangeable shock-arrival intensities in Eq. (17a) had previously been stated in [17, p. 149]. □

The second integral representation for Q^* uses the *Stieltjes representation* from Eq. (6), which exists for all complete Bernstein functions.

Lemma 5.5. *Let ψ be a complete Bernstein function with Stieltjes triplet $(0, 0, \sigma)$ such that Eq. (6) is fulfilled. Then,*

$$(-1)^{i-1}\Delta^i\psi(x) = \int_0^\infty u \cdot B(1+i, x+u)\sigma(\mathrm{d}u), \quad x \geq 0,\ i \in \mathbb{N},$$

where $B(x,y) = \Gamma(x)\Gamma(y)/\Gamma(x+y)$, $x, y > 0$, is the Beta function.

Proof. Due to the linearity of the integral, it suffices to show that

$$(-1)^{i-1}\Delta^i \frac{x}{x+u} = u \cdot B(1+i, x+u), \quad x \geq 0,\ u > 0,\ i \in \mathbb{N}.$$

We prove this by induction. For this, note that we have

$$\begin{aligned}\Delta\frac{x}{x+u} &= \frac{x+1}{x+1+u} - \frac{x}{x+u} = \frac{(x+1)(x+u)}{(x+u+1)^{\underline{2}}} - \frac{x(x+u+1)}{(x+u+1)^{\underline{2}}} = u \cdot \frac{1}{(x+u+1)^{\underline{2}}} = u \cdot \frac{\Gamma(2)\Gamma(x+u)}{\Gamma(x+u+2)} \\ &= u \cdot B(1+1, x+u).\end{aligned}$$

Now, assume that the claim is true for $i-1$. Then,

$$\begin{aligned}(-1)^{i-1}\Delta^i\frac{x}{x+u} &= -\Delta\left[(-1)^{i-2}\Delta^{i-1}\frac{x}{x+u}\right] \overset{\text{ind.}}{=} -\Delta[u \cdot B(1+i-1, x+u)] \\ &= -\Delta\left[u \cdot \frac{\Gamma(i)\Gamma(x+u)}{\Gamma(x+u+i)}\right] = -\Delta\left[u \cdot \frac{(i-1)!}{(x+u+i-1)^{\underline{i}}}\right] \\ &= u(i-1)! \cdot \left[\frac{1}{(x+u+i-1)^{\underline{i}}} - \frac{1}{(x+u+i)^{\underline{i}}}\right] \\ &= u(i-1)! \cdot \left[\frac{(x+u+i)-(x+u)}{(x+u+i)^{\underline{i}}(x+u)}\right] = u \cdot \frac{i!}{(x+u+i)^{\underline{i+1}}} \\ &= u \cdot \frac{\Gamma(i+1)\Gamma(x+u)}{\Gamma(x+u+i+1)} = u \cdot B(1+i, x+u).\end{aligned}$$

□

Theorem 5.6. *Let $d \geq 2$ and consider a d-variate extMO distribution with complete Bernstein function ψ and Stieltjes triplet (a, b, σ) such that Eq. (6) is fulfilled. Then, for $0 \leq i < j \leq d$,*

$$\lambda_i = a\mathbb{1}_{\{i=d\}} + b\mathbb{1}_{\{i=1\}} + \int_0^\infty u \cdot B(1+i, d-i+u)\sigma(\mathrm{d}u) \tag{18a}$$

and

$$q^*_{ij} = \binom{d-i}{j-i} \cdot [a\mathbb{1}_{\{j=d\}} + b\mathbb{1}_{\{j-i=1\}}] + \int_0^\infty \binom{d-i}{j-i} \cdot u \cdot B(1+j-i, d-j+u)\sigma(\mathrm{d}u), \tag{18b}$$

where B is the Beta function.

Proof. The result is a direct corollary of Eq. (9), Eq. (16), Lemma 5.5, and Proposition 5.3. □

Numerical approximation of Q^* based on integral representations

We want to use the integral representations from Eqs. (17) and (18) to approximate Q^*. However, even if integrals are finite in theory, numerical integration routines can fail for various reasons. In the following, we provide further insight into using numerical integration methods for the two representations.

Remark 5.7. (Numerical integration in R with QUADPACK algorithms). We can perform numerical integration in R with the `stats::integrate` routine. The method uses the well-known *QUADPACK* algorithms *QAGI* (indefinite integrals) and *QAGS* (definite integrals), see [23,24]. These algorithms are also used in the *GNU scientific library* (*GSL*), see [8]. They are adaptive numerical integration algorithms based on a 15-point and 21-point *Gauss–Kronrod quadrature* with the convergence acceleration technique *Wynn's ε-algorithm* for a limit extrapolation, respectively. The former method transforms indefinite integrals with lower bound $a \in \mathbb{R}$ as follows:

$$\int_a^\infty f(x)\mathrm{d}x = \int_0^1 f\left(a + \frac{1-y}{y}\right)y^{-2}\mathrm{d}y.$$

Theoretically, the limit of the *m-panel*, *n*-point Gauss–Kronrod quadrature, which applies the n-point quadrature to m equidistant subintervals of the integration domain, converges to the integral for $m \to \infty$ if f is bounded and Riemann-integrable, see [23, Thm. 2.6]. This approximation also converges for some functions with singularities, e.g., $f(x) = x^\beta$, $\beta > -1$ for $x \in (0, 1)$; however, the convergence is very slow in this case if $\beta < 0$, see [23, p. 42]. A low convergence rate can mislead implementations to falsely detect divergence.

First, note that the integral is the inner product of weights and integrand values if the Lévy measure, or Stieltjes measure, is discrete with finite support.

In the remainder of this subsection, assume that the Lévy measure, or Stieltjes measure, has a continuous density with respect to the Lebesgue measure on $(0, \infty)$. While there are examples for which this is not the case, e.g., the Stieltjes density of no. 5 in [28, p. 304], it is fulfilled for all examples with nondiscrete representation measures discussed in this article.

The following boundary conditions for continuous Lévy and Stieltjes densities allow identifying the cases for which numerical representations of q_{ij}^* have singularities.

Lemma 5.8. *Let f be a continuous Lévy density, i.e., a continuous function on $(0, \infty)$ such that $\int_0^\infty (1 \wedge u)f(u)\mathrm{d}u < \infty$. Then,*

$$\lim_{x\to 0} x^2 f(x) = 0 \tag{19a}$$

and

$$\lim_{x\to\infty} xf(x) = 0. \tag{19b}$$

Proof. First, for Eq. (19a), note that $\int_0^\infty (1 \wedge u)f(u)\mathrm{d}u < \infty$ implies by

$$0 = \lim_{y\to 0}\int_0^y uf(u)\mathrm{d}u = \lim_{y\searrow 0} y\cdot\zeta(y)f(\zeta(y)) \geq \lim_{x\searrow 0} x^2 f(x) \geq 0,$$

where $\zeta(y) \in [0, y]$ are determined by the *mean value theorem for integration* using continuity of f. Second, for Eq. (19b), using the substitution $u = (1-t)t^{-1}$, we have

$$\infty > \int_0^\infty (1 \wedge u) f(u) \mathrm{d}u \geq \int_0^\infty \frac{u}{1+u} f(u) \mathrm{d}u = \int_0^1 \frac{1-t}{t^2} f((1-t)t^{-1}) \mathrm{d}t.$$

Hence,

$$\begin{aligned} 0 &= \lim_{z \to 0} \int_0^z \frac{1-t}{t^2} f((1-t)t^{-1}) \mathrm{d}t = \lim_{z \to 0} z \cdot \frac{1-\zeta(z)}{\zeta(z)^2} f((1-\zeta(z))\zeta(z)^{-1}) \\ &\geq \lim_{y \to 0} y \cdot \frac{1-y}{y^2} f((1-y)y^{-1}) = \lim_{x \to \infty} x f(x) \geq 0, \end{aligned}$$

where $\zeta(z) \in [0, z]$ are again determined by the *mean value theorem for integration*. □

Lemma 5.9. *Let f be a continuous Stieltjes density, i.e., a continuous function on $(0, \infty)$ such that $\int_0^\infty (1+u)^{-1} g(u) \mathrm{d}u < \infty$. Then,*

$$\lim_{x \to 0} x g(x) = 0 \tag{20a}$$

and

$$\lim_{x \to \infty} g(x) = 0. \tag{20b}$$

Proof. First, for Eq. (20a), note that $\int_0^\infty (1+u)^{-1} g(u) \mathrm{d}u < \infty$ implies

$$0 = \lim_{y \to 0} \int_0^y (1+u)^{-1} g(u) \mathrm{d}u = \lim_{y \to 0} y(1+\zeta(y))^{-1} g(\zeta(y)) \geq \lim_{x \to 0} x g(x) \geq 0,$$

where $\zeta(y) \in [0, y]$ are determined by the *mean value theorem for integration* using continuity of g. Second, for Eq. (20b), consider that, using the substitution $u = (1-t)t^{-1}$, we have

$$\infty > \int_0^\infty (1+u)^{-1} g(u) \mathrm{d}u = \int_0^1 g((1-t)t^{-1}) t^{-1} \mathrm{d}t.$$

Hence,

$$\begin{aligned} 0 &= \lim_{z \to 0} \int_0^z g((1-t)t^{-1}) t^{-1} \mathrm{d}t = \lim_{z \to 0} z \cdot g((1-\zeta(z))\zeta(z)^{-1}) \zeta(z)^{-1} \\ &\geq \lim_{y \to 0} g((1-y)y^{-1}) = \lim_{x \to \infty} g(x) \geq 0, \end{aligned}$$

where $\zeta(z) \in [0, z]$ are again determined by the *mean value theorem for integration*. □

By using the boundary results from Lemma 5.8, we obtain for a continuous Lévy density f and numbers $\gamma > 0$, $x \geq 0$, and $k \in \mathbb{N}$

$$\lim_{t \to 0} \gamma \cdot e^{-x(1-t)t^{-1}} [1 - e^{-(1-t)t^{-1}}]^k f((1-t)t^{-1}) t^{-2} = \gamma \cdot \lim_{u \to \infty} e^{-xu} u^2 f(u) \begin{cases} \text{unknown} & x = 0, \\ = 0 & x > 0 \end{cases}$$

and

$$\lim_{t \to 1} \gamma \cdot e^{-x(1-t)t^{-1}} [1 - e^{-(1-t)t^{-1}}]^k f((1-t)t^{-1}) t^{-2} = \gamma \cdot \lim_{u \to 0} u^k f(u) \begin{cases} \text{unknown} & k = 1, \\ = 0 & k \geq 2. \end{cases}$$

Furthermore, by using the boundary results from Lemma 5.9, we obtain for a continuous Stieltjes density g and numbers $\gamma > 0$, $x \geq 0$, and $k \in \mathbb{N}$

$$\lim_{t\to 0} \gamma\cdot(1-t)t^{-3}B(1+k,x+(1-t)t^{-1})g((1-t)t^{-1}) = \gamma\cdot\lim_{u\to\infty}\frac{u^3\cdot k!\cdot g(u)}{(x+u+k)^{\underline{k+1}}}\begin{cases}\text{unknown} & k=1,\\ =0 & k\geq 2,\end{cases}$$

and

$$\lim_{t\to 1} \gamma\cdot(1-t)t^{-3}B(1+k,x+(1-t)t^{-1})g((1-t)t^{-1}) = \gamma\cdot\lim_{u\to 0}\frac{u\cdot k!\cdot g(u)}{(x+u+k)^{\underline{k+1}}}\begin{cases}\text{unknown} & x=0,\\ =0 & x>0.\end{cases}$$

Thus, for a continuous Lévy density, or Stieltjes density, the integrand in Eq. (17), or Eq. (18), after a substitution $u = (1-t)t^{-1}$, tends to zero at both ends whenever $j < d$ and $j - i > 1$. Consequently, under these assumptions, the *QUADPACK* algorithms are well suited for approximating the integral Eq. (17), or Eq. (18), if $j < d$ and $j - i > 1$.

Improved numerical approximation of Q^*

The previous subsection outlined that, given a continuous Lévy density, or Stieltjes density, we can expect *QUADPACK* algorithms to work well for the respective integral representations of q^*_{ij} whenever $j - i > 1$ and $j < d$. That does not imply that they will fail if $j - i = 1$ or $j = d$. However, if they do, we can use that these exceptions are boundary cases:

- For $j - i = 1$, we have

$$q^*_{i,i+1} = (d-i)\cdot[\psi(d-i) - \psi(d-i-1)].$$

  Hence, we can calculate $q^*_{i,i+1}$ with the naïve approach as the result of a single finite forward difference. This calculation is sufficiently exact and does not suffer from the numerical issues of recursively calculating finite differences.
- For $j = d$, we have multiple options. Here, the problem is a possible singularity for the integrand. The first option is to use the continuity of the integrand by replacing the expression $d - j$ iteratively with an increasing sequence of real numbers $d - j + \varepsilon_k$, $k \in \mathbb{N}$, until the numerical integration succeeds. The second option is to use that row sums of Q^* have to be equal to zero. Thus, this condition implicitly defines an approximation of q^*_{id}, given approximations of all other values, flooring it to zero if the resulting approximation is negative. Of both approaches, we suggest using the latter over the former since it is not iterative and flooring should only be necessary if the value is already close to zero, given that the remaining values are sufficiently accurate.

To avoid rounding issues, we also suggest using the following recursive representation for products with binomial coefficients for $0 \leq k \leq n$:

$$\left[\binom{n}{k}\cdot x\right] = \begin{cases}\frac{n}{k}\cdot\left[\binom{n-1}{k-1}\cdot x\right] & 1\leq k\leq \lfloor n/2\rfloor,\\ \left[\binom{n}{n-k}\cdot x\right] & k>\lfloor n/2\rfloor,\\ x & k=0.\end{cases}$$

The following result allows using a recursion for numerically calculating the remaining rows from the first row of Q^*. Consequently, it is sufficient to approximate only the first row with numerical integration.

Theorem 5.10. *Let $d \geq 2$ and consider a d-variate extMO distribution with Bernstein function ψ and infinitesimal generator matrix Q^* for the corresponding law of the death-counting process. Then, for $i < j$,*

$$q^*_{i+1,j+1} = \frac{d-j}{d-i}\cdot q^*_{i,j} + \frac{j+1-i}{d-i}\cdot q^*_{i,j+1}.$$

Proof. We have

$$\begin{aligned}(-1)^{j-i-1}\Delta^{j-i}\psi(d-j-1) &= (-1)^{j-i-1}\Delta^{j-i}[\psi(d-j)+\psi(d-j-1)-\psi(d-j)]\\ &= (-1)^{j-i-1}\Delta^{j-i}\psi(d-j)+(-1)^{j-i}\Delta^{j-i+1}\psi(d-j-1).\end{aligned}$$

Consequently,

$$q^*_{i+1,j+1} = \frac{\binom{d-i-1}{j-1}}{\binom{d-i}{j-i}}q^*_{i,j} + \frac{\binom{d-i-1}{j-i}}{\binom{d-i}{j+1-i}}q^*_{i,j+1} = \frac{d-j}{d-i}\cdot q^*_{i,j} + \frac{j+1-i}{d-i}\cdot q^*_{i,j+1}. \qquad \square$$

The following remark summarizes the proposed approximation of Q^*.

Remark 5.11. Assuming a continuous Lévy density, or Stieltjes density, we can approximate Q^* as follows:
(1) Set all of the lower-triangular values to zero.
(2) Numerically calculate the diagonal values as $-\psi(d-i)$, $i \in [d]_0$.
(3) Approximate $q^*_{0,j}$, $1 < j < d$, using numerical integration, e.g., with the *QUADPACK* algorithms, based on one of the integral representations Eqs. (17) and (18).
(4) For $j = 1$, calculate $q^*_{0,1}$ as $d[\psi(d) - \psi(d-1)]$ numerically.
(5) For $j = d$, set $q^*_{0,d}$ to the value, equating the first-row sum of Q^* to zero, and then floor this value by zero.
(6) Calculate the remaining rows of Q^* numerically using the recursion from Theorem 5.10.

We repeated the previous numerical studies for the proposed approach using the relative tolerance $\sqrt{\varepsilon} \approx 1.490116 \times 10^{-8}$ for the accuracy of the numerical integration, where $\varepsilon = 2^{-52}$ is the *machine epsilon* for a *binary64* number. We can see in Figures 3 and 4 that $\varepsilon_1(d)$ and $\varepsilon_2(d)$ stay below or around $\sqrt{\varepsilon}$ up to $d = 125$.

6 Benchmarks of extMO simulation algorithms

Previously, we derived the Markov death-counting model (MDCM) for exMO distributions in Section 4. Furthermore, we have shown how their inputs, the infinitesimal Markov generators of their death-counting processes, can be approximated in a numerically stable way from Bernstein functions, parametrizing extMO distributions, in Section 5. In combination with Algorithm 1, we obtain a novel simulation algorithm for high-dimensional extMO distributions. This section aims to compare the runtime of this algorithm with that of alternative sampling algorithms for extMO distributions, summarized in Table 1. We also investigate the algorithm's setup activities' proportion of the overall runtime.

We performed the presented benchmarks on a Windows consumer laptop. In addition, we also repeated them on other devices and operating systems. If not mentioned otherwise, the presented findings were representative of all machines. However, we want to stress that our intention with the subsequent analysis is merely to indicate how the runtimes compare; hardware and software changes can significantly impact such measurements.

The following is a high-level summary of our key findings from theoretical considerations and subsequent benchmarks.

- The general MO simulation algorithms ESM and AM appear ill suited for simulating extMO distributions in higher dimensions without tweaking the algorithms to special cases. In particular, both have relatively large memory requirements for $d \approx 30$, and the former shows significantly larger runtimes up from $d \approx 8$ than all other benchmarked algorithms.
- The MDCM was slower than the LFM for most tested configurations. However, the former required a large proportion of its runtime for the initial setup. Consequently, the gaps between both algorithms' runtimes decreased when increasing the sample size. In addition, we found a way to choose the parameters so that

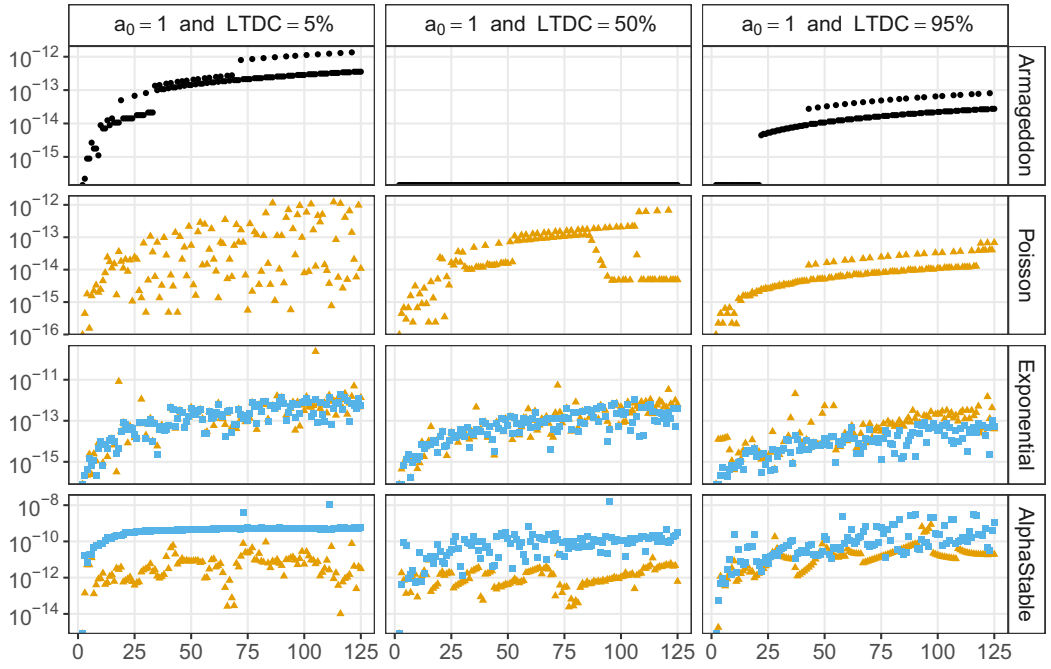

Figure 3: The maximum relative differences $\varepsilon_1(d)$ of the off-diagonal row-sums for the proposed calculation of Q^* for $2 \leq d \leq 50$; coloring and shape indicate the used integral representation with none (black, round), i.e., closed-form, Lévy (yellow, triangle), and Stieltjes (blue, square).

the LFM's (expected) runtime becomes arbitrarily large — such a scenario does not exist for the MDCM as its dimension bounds the transition number.

Overall, we conclude that the MDCM is a viable option for simulating extMO distributions in dimensions at least up to $d = 128$. The LFM is an alternative for suitable extMO distributions. However, it has issues with its runtime explosions in corner cases, and it has limited applicability. In particular, exact simulation using the LFM requires the underlying Lévy subordinator to be a compound Poisson process and being able to simulate from its jump distribution. In contrast, if the Lévy subordinator is not compound Poisson, the corresponding extMO distribution cannot be sampled exactly with the LFM but requires approximations similar to those in [7, Sec. 5.3]. The MDCM also relies on numerically integrating Q^*. However, the corresponding approximation errors can be bound with an appropriate numerical integration method, as

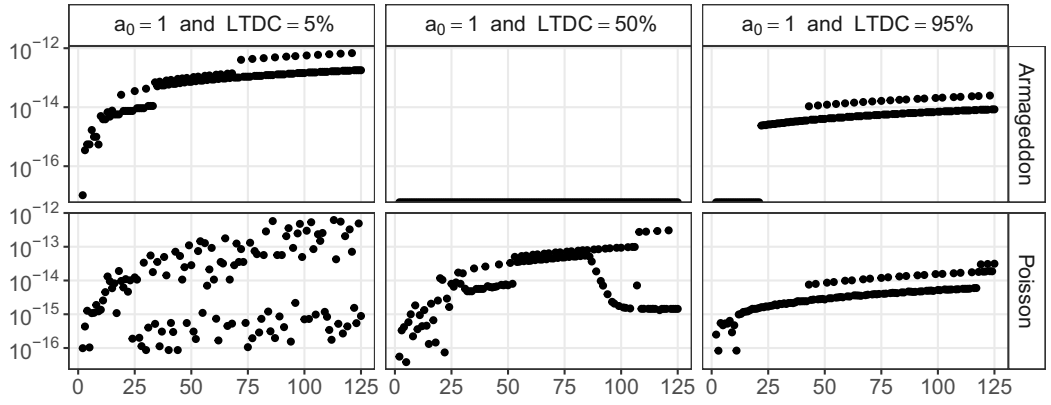

Figure 4: The maximum total variation distance $\varepsilon_2(d)$ of the transition distributions of the proposed and high-precision calculation of Q^* for $2 \leq d \leq 50$.

Table 1: An overview of our implementations for the runtime comparison; for more details on the software used, see our statement following the conclusion

Algorithm	Input[e]	Restrictions	
		Subfamily	Max. dimension[f]
ESM[a]	$\{\lambda_I : I \subseteq [d]\}$	MO	$\log_2(n_{\max} + 1) - 1$
AM[b]	$\{\lambda_I : I \subseteq [d]\}$	MO	$\log_2(n_{\max} + 1) - 1$
MDCM[c]	$\{\eta_i : i \in [d]\}$	exMO	$n_{\max}$
LFM[d]	a, b, $\nu(0, \infty)$, and jump dist. param.	extMO[d]	$n_{\max}$
Mod. ESM[a]	$\alpha = \lambda_1$ and $\beta = \lambda_d$	Armageddon	$n_{\max}$

[a] Optimized implementation of [17, Algo. 3.1], see Section 2.1, skipping a shock at runtime if the shock-arrival intensity is zero and using bit arithmetics for death verification. We modified the algorithm for the Armageddon family to sampling $d + 1$ exponential random variables and performing a loop with d bivariate minimum operations.
[b] Optimized implementation of [17, Algo. 3.3], see Section 2.1, using Walker's alias algorithm for discrete sampling, see [31], and bit arithmetics for death verification.
[c] Implementation of Algorithm 1 using the representation of Theorem 2.3 with Walker's alias algorithm for discrete sampling.
[d] The LFM requires the subordinator to be a compound Poisson process with a feasible jump distribution sampling algorithm. We implemented [17, Algo. 3.7] with deterministic, exponential, and Pareto distributed jumps.
[e] The input parameters for ESM, AM, and MDCM are calculated in `R` using the approach from Section 5.
[f] Our implementation's (technical) maximum dimension is expressed as a function of the maximum vector size $n_{\max}$. E.g., in the case of the vector size being a 32-bit unsigned integer, we have $n_{\max} = 2^{32} - 1$; `R` with `LongVector`-support has a technical maximum of $n_{\max} = 2^{52}$, but considering a 2^{30}-length `binary64` double vector requires roughly $8.6GB$ memory, we believe that general MO distributions are infeasible long before that technical maximum is reached. Note that implementations for larger dimensions are possible but require specialized data structures and could be suboptimal for smaller dimensions. In particular, optimizations based on bit arithmetics might not be possible anymore.

discussed in Section 5. Furthermore, statistical tests did not reject any distributional assumption for our implementation of the MDCM for any subfamilies discussed in this section; see Appendix D. Finally, we want to highlight that we can use the MDCM for arbitrary extMO distributions with closed-form, continuous Lévy or Stieltjes densities without requiring further specialization.

For the ensuing comparisons, we describe extMO distributions by the properties of their subordinators in the LFM representation from Theorem 2.4, as it is exceptionally well suited for obtaining a basic understanding of the probability law. For this, recall that we can characterize every extMO distribution by a Bernstein function ψ, which defines the law of a (potentially killed) Lévy subordinator. Components corresponding to the extMO distributed random vector are killed once the subordinator passes their individual unit exponential barrier values. For compound Poisson subordinators, we distinguish subordinator laws by their jump intensity and jump size distribution: The jump intensity translates into the overall speed with which the subordinator surpasses the barrier values. Thus, it corresponds to the random vector's marginal rate. The distribution of jump sizes predefines the chances of the subordinator simultaneously surpassing multiple barrier values, and therefore, it corresponds to the random vector's dependence structure. Simply put, a high probability of larger jumps increases the chance of simultaneous deaths, while a high probability of smaller jumps increases the likelihood of individual deaths. This logic culminates into the pure-drift and pure-killing corner cases, corresponding to the independence and comonotonicity, respectively, pure-jump Lévy subordinators with *infinite activity*, which can be approximated by compound Poisson processes, and convex combinations of those above.

We have chosen the exponential family with possible drift and killing as a representative example for our subsequent benchmarks for two reasons: First, its Bernstein function has a simple form. Second, its LFM representation requires the simulation of jumps but does not require potentially expensive simulation techniques such as *rejection sampling*. However, whenever possible, we performed the following benchmarks for all other families from Section 2.4 without observing noteworthy structural differences.

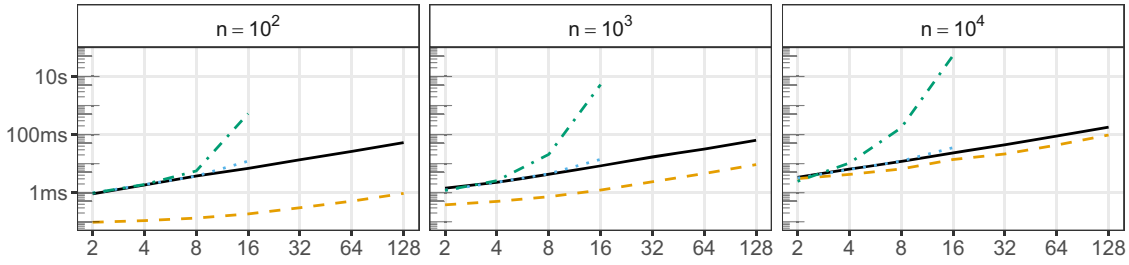

Figure 5: The median runtime of the MDCM (solid, black), LFM (dashed, yellow), AM (dotted, blue), and ESM (dash-dotted, green) algorithms for an extMO distribution from the exponential family with drift and without killing, calibrated to unit exponential margins and a bivariate lower-tail dependence coefficient of 50%. We measured the AM's and ESM's runtimes for $d \leq 16$.

We begin by comparing the MDCM's simulation runtime to that of the LFM, AM, and ESM for the exponential family, using normalization by adding a linear part, see Section 2.4. Furthermore, we fix the bivariate margin to an extMO distribution with unit margins and a lower-tail dependence coefficient of 50%, see [22, Exmpl. 5.21]. We make the following observations from the benchmark results in Figures 5 and 6:

- We conclude that the ESM is ill suited for higher dimensions due to exploding runtimes for increasing dimensions. In contrast, the AM seems to be less problematic: although it has higher setup requirements than the ESM, the overall runtime is slightly higher than that of the MDCM and LFM but significantly smaller than that of the ESM. We attribute this in part to using the highly efficient *Walker's alias method* for discrete sampling and also in part to a binary representation of shock sets, allowing a quick death verification. However, using a binary representation for shock sets makes it challenging to scale this particular implementation to higher dimensions, e.g., our implementation is limited to $d = 30$ with 16 GB memory. Furthermore, Figure 6 shows that the setup activities' proportion of the median runtime is similar for the MDCM and the AM, but diminishes for the ESM with increasing dimensions.
- In this benchmark, the MDCM's median runtime was measurably slower than that of the LFM. However, the gap shrinks significantly when increasing the sample size from $n = 10^2$ to $n = 10^4$.
- In contrast to the LFM, the MDCM requires setup activities, mapping the Bernstein function to the infinitesimal generator, contributing significantly to the overall runtime. We observe from Figure 6 that while the proportional runtime of setup activities is sizeable for small sample sizes, it becomes less pronounced for larger sample sizes. In particular, for $d = 128$, the runtime proportion of the setup decreases significantly by a factor of approximately 3 when increasing the sample size from $n = 10^2$ to $n = 10^4$. Note that the proportional runtime of setup activities could be decreased as, contrary to the sampling algorithm, we implemented them in `R` and not `C++`.

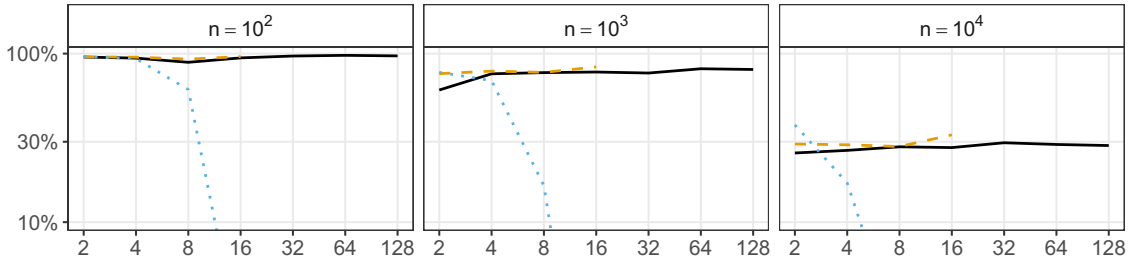

Figure 6: The setup activities' proportion of the median runtime of the MDCM (solid, black), AM (dashed, yellow), and ESM (dotted, blue) algorithms for extMO distributions from the exponential family with drift and without killing, calibrated to unit exponential margins and a bivariate lower-tail dependence coefficient of 50%. We measured the setup activities' median runtime in a separate benchmark. Similar to Figure 5, we measured the AM's and ESM's runtimes for $d \leq 16$.

We continue with a short theoretical comparison of the MDCM and LFM for the case without drift or killing to highlight the possibility of exploding runtimes for the LFM: Both stochastic models have embedded transition processes counting the number of dead components. For the LFM, the compound Poisson subordinator can surpass several barrier values and trigger the death of the corresponding components with each new increment. The counting process accumulating the death toll over these transitions has a similar property to the AM: it can loop in the same state for several transitions. In particular, we show in Appendix C that the number of transitions to surpass the barrier value of the first component, subsequently called *barrier-exceedance count*, has a geometric distribution with success probability $p = 1 - \kappa(1)$, where κ is the Laplace transform of the jump distribution. Consequently, the expected number of transitions for the subordinator surpassing all barrier values is at least $1/[1 - \kappa(1)]$ and can become arbitrarily large depending on the jump distribution. Conversely, the death counting process of the MDCM cannot loop in the same state. Thus, the total number of transitions until reaching the absorbing state cannot exceed dimension d. For more details on the expected number of transitions of the MDCM and LFM, see Appendix C.

In the second benchmark, we demonstrate the issue of exploding runtimes for the LFM using a special case of the exponential family: a compound Poisson subordinator with exponentially distributed jumps without drift or killing. By decreasing p, the success probability of the geometric distribution corresponding to the barrier-exceedance count from the previous paragraph, we can produce a particularly adverse parametrization for the LFM. A short calculation yields that $p = 50\%$, $p = 10\%$, and $p = 1\%$, respectively, correspond to rates $\eta = 1$, $\eta = 9$, and $\eta = 99$ of the jump distribution. We conclude from Figure 7 that the benchmarks for smaller success probabilities p highlight an advantage of the MDCM over the LFM: The dimension bounds the number of transitions in the former model. Thus, the median runtime cannot *explode* if the success probability approaches zero as in the LFM.

In the previous benchmark, we chose parameters to highlight a disadvantage of the LFM for small success probabilities of the first component's barrier-exceedance count distribution. Now, we probe a parametrization more to the LFM's advantage and, in particular, to the MDCM's disadvantage. For this, note that we can expect the fastest runtime of the LFM for a pure-drift subordinator, corresponding to an independence distribution: The LFM algorithm samples and sorts the barrier values and iterates over the barrier values, setting the random variables to the barrier values. In particular, it requires no simulation of subordinator increments. Conversely, the independence case constitutes the worst case for the MDCM as the death-counting process will always require d transitions into the absorbing state. Nevertheless, we infer

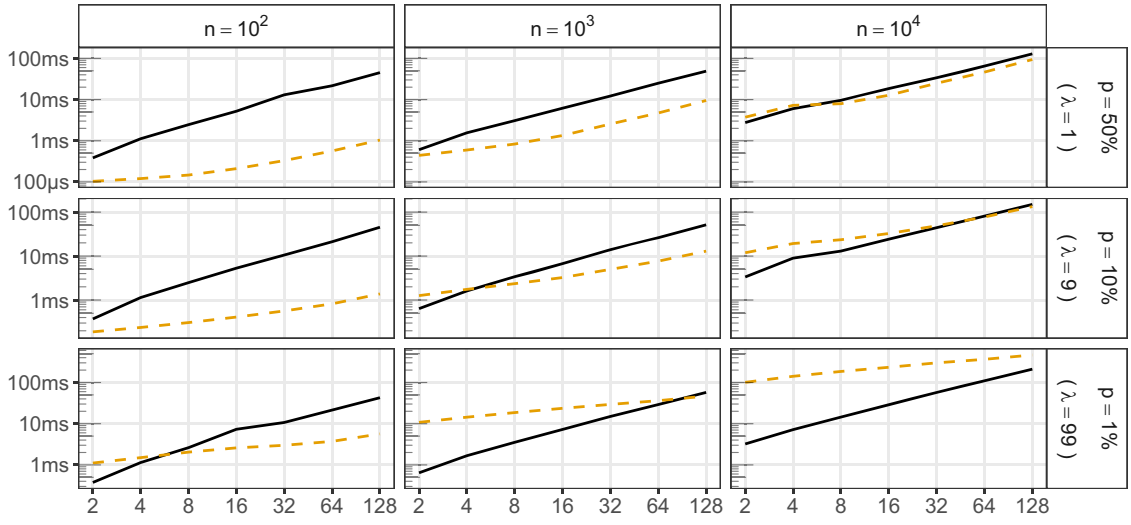

Figure 7: The median runtime of the MDCM (solid, black) and LFM (dashed, yellow) algorithms for extMO distributions from the exponential family without drift or killing. We used three configurations for various success probabilities p from the first component's (geometric) barrier-exceedance count distribution.

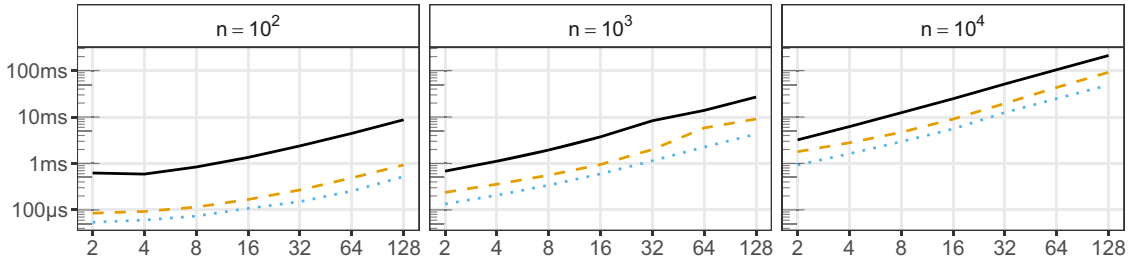

Figure 8: The median runtime of the MDCM (solid, black) and LFM algorithms (dashed, yellow) for the extMO distribution with a pure-drift subordinator (independence case). We included the optimized ESM algorithm for the Armageddon family (dotted, blue) as a reference. It hints at the costs of the sorting operation of the LFM algorithm, which is superfluous for a pure-drift subordinator.

from the benchmark results in Figure 8 that the LFM algorithm is only approximately twice as fast as the MDCM algorithm in this parametrization for larger sample sizes.

7 Conclusion

We started this article with the question: "Can we find low-parametric Markov-based models for extendible Marshall–Olkin distributions with a numerically stable implementation?"

We showed that the death-counting process of exchangeable Marshall–Olkin distributions is Markov. Furthermore, we proved that its shuffled transition times' distribution equals the original one. This result allows us to represent exchangeable Marshall–Olkin distributions directly with Markovian death-counting processes.

Using this Markov representation requires calculating its infinitesimal generator from the distribution's parameters. Many well-known examples of low-parametric subfamilies of extendible Marshall–Olkin distributions are parametrized via Bernstein functions, the Lévy exponents of the subordinators from the Lévy frailty model representation. However, we found that naïvely calculating Markov generators from Bernstein functions is not numerically stable. Therefore, we derived a numerically stable approximation of the generator's first row using integral representations of Bernstein functions and summation identities and calculate its remaining rows using a recursion. We conducted a numerical study to demonstrate this approach's numerical stability for various examples up to dimension $d = 128$.

We proposed a new simulation algorithm by combining extendible Marshall–Olkin distributions' Markov representations with the numerically stable approximation of their generators. However, this algorithm necessitates significant setup activities for calculating the generators from Bernstein functions, requiring numerical integration and recursions. We benchmarked the runtime of our new algorithm to that of alternative sampling algorithms, corresponding to the Lévy frailty model, the Arnold model, and the exogenous shock model, and obtained the following findings: First, we found that the setup activities' proportion of our algorithm's overall runtime was sizeable for small sample sizes, e.g., 10^2, but decreased significantly when moving toward larger sample sizes, e.g., 10^4. Second, we found that both specialized algorithms — our algorithm and the algorithm corresponding to the Lévy frailty model — were faster than the general algorithms, corresponding to the Arnold model and exogenous shock model. Third, we found that our algorithm was, in many cases, slower than the algorithm corresponding to the Lévy frailty model. However, the gap between the measured runtimes decreased when increasing the sample size and, in contrast to its counterpart, our simulation algorithm's runtime is bounded by the dimension. Finally, we analyzed worst-case parametrizations for the Markov death-counting model and the Lévy frailty model. We found that the latter's runtime explodes for particular parameter asymptotics. In contrast, the former's runtime is bound for a fixed dimension.

Apart from runtime differences, the Lévy frailty model and Markov death-counting model have significant methodological differences. The Lévy frailty model comprises specialized sampling algorithms, each depending on a specific jump distribution algorithm. In contrast, the Markov death-counting model is a general sampling algorithm for all extendible Marshall–Olkin distributions, requiring no specialization if it has suitable Lévy or Stieltjes density. The Markov death-counting model requires an approximation of the simulation parameters except for special cases, but the approximation uses numerical integration methods, which can bound the error. Moreover, the Lévy frailty model also requires approximations if the corresponding Lévy subordinator is not of compound Poisson type.

We conclude that our novel Markov-based simulation algorithm is well suited for simulating extendible Marshall–Olkin distributions. In contrast to the Lévy frailty model, our Markov representation covers all extendible Marshall–Olkin distributions and does not require tailored jump distribution algorithms.

Acknowledgement: Many thanks to Lexuri Fernández, Matthias Scherer, and the anonymous referees for their feedback on the article.

Funding information: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflict of interest: The author states no conflict of interest.

Software used: We used `R` (see [24]) for implementing the MDCM, LFM, AM, and ESM simulation algorithms, the corresponding parameter mapping routines from Section 5, and for performing the benchmarks and numerical studies. We implemented the simulation algorithms in `C++17` (see [11]) using the `C++` interface of the R-package `Rcpp` (see [6]). We used the R-package `bench` (see [10]) for our benchmark studies.

Appendix

A The infinitesimal generator matrix Q for special cases

This appendix contains the MDSM's generator matrix representations for the exchangeable subclass, the extendible subclass, and the Armageddon shock subclass. The following derivations constitute mostly tedious but straightforward calculations. However, we use their results indirectly in the main part of this article, e.g., to derive Eq. (16) or Proposition 5.3.

Remark A.1. For an exMO distribution with exchangeable shock-arrival intensities $\boldsymbol{\lambda} = \lambda \cdot \boldsymbol{p}$, we have

$$k_{IJ} = \begin{cases} \sum_{l=0}^{i} \binom{i}{l} p_{l+(j-i)} & I \subseteq J \quad \text{with} \quad |I| = i, \quad |J| = j, \\ 0 & \text{else}, \end{cases}$$

and

$$q_{IJ} = \begin{cases} -\sum_{k=1}^{d-i}\binom{d-i}{k}\sum_{l=0}^{i}\binom{i}{l}\lambda_{l+k} & I = J, \quad |I| = i, \\ \sum_{l=0}^{i}\binom{i}{l}\lambda_{l+(j-i)} & I \subsetneq J, \ |I| = i, \ |J| = j, \\ 0 & \text{else}. \end{cases}$$

Proof. We obtain both results from Eqs. (10) and (12) by using the exchangeability and by grouping all shock-arrival intensities, or shock-arrival probabilities, belonging to the same cardinalities. □

ExMO distributions are often reparametrized with so-called *d-monotone sequences* instead of the exchangeable shock-arrival intensities. The reparametrization was proposed in [13] and extensively studied by [14, Chp. 3]. The latter serves as a reference for the subsequently summarized results on this reparametrization. We define for the exchangeable shock-arrival intensities

$$a_k := \sum_{j=0}^{d-k-1}\binom{d-k-1}{j}\lambda_{j+1}, \quad k \in [d-1]_0 = \{0, \ldots, d-1\}.$$

Then, the sequence $a_0, a_1, \ldots, a_{d-1}$ is *d-monotone*, i.e.,

$$(-1)^{i-1}\Delta^{i-1}a_{d-i} = \sum_{j=0}^{i-1}(-1)^j\binom{i-1}{j}a_{d-i+j} \geq 0, \quad i \in [d].$$

In particular, [14, Chp. 3] shows that

$$\lambda_i = (-1)^{i-1}\Delta^{i-1}a_{d-i}, \quad i \in [d], \tag{A.1}$$

and, moreover, every d-monotone sequence defines an exMO distribution. Furthermore, [14, Chp. 4] shows that the reparametrization of an extMO distribution with Bernstein function ψ is

$$a_i = \psi(i+1) - \psi(i), \quad i \in [d-1]_0. \tag{A.2}$$

This reparametrization bridges the gap toward the parametrization of extMO distributions via Bernstein functions and has the following property: the first k sequence elements describe the law of k-margins, i.e., the subvector $(\tau_1, \ldots, \tau_k)$, $k \in [d]$, has the reparametrization $a_0, \ldots, a_{k-1}$ and the margin τ_1 is exponentially distributed with rate a_0. We have the following representation of the MDSM's Markov generator matrix in terms of the reparametrization:

Remark A.2. For an exMO distribution with reparametrization $\boldsymbol{a}$, we have

$$k_{IJ} = \frac{1}{\sum_{k=0}^{d-1}a_k}\begin{cases} \sum_{k=0}^{d-1}a_k - \sum_{k=0}^{d-i-1}a_k & I = J, \quad |I| = i, \\ (-1)^{j-i-1}\Delta^{j-i-1}a_{d-j} & I \subsetneq J, \quad |I| = i, \quad |J| = j, \\ 0 & \text{else}, \end{cases}$$

and

$$q_{IJ} = \begin{cases} -\sum_{k=0}^{d-i-1}a_k & I = J, \quad |I| = i, \\ (-1)^{j-i-1}\Delta^{j-i-1}a_{d-j} & I \subsetneq J, \quad |I| = i, \quad |J| = j, \\ 0 & \text{else}. \end{cases}$$

For the proof of Remark A.2, we require the following auxiliary lemma:

Lemma A.3. *Let $d \in \mathbb{N}$, $c_0, c_1, \ldots, c_d$ be a real sequence, and $i, j \in [d]_0$ with $i \leq j$. Then,*

$$\sum_{k=0}^{i}\binom{i}{k}(-1)^{k+j-i-1}\Delta^{k+j-i}c_{d-k-(j-i)} = (-1)^{j-i-1}\Delta^{j-i}c_{d-j}. \tag{A.3}$$

Proof. We first proof the claim for $j - i = 0$, i.e.,

$$\sum_{k=0}^{i}\binom{i}{k}(-1)^{k-1}\Delta^{k}c_{d-k} = -c_{d-i}. \tag{A.4}$$

For this, we have

$$\sum_{k=0}^{i}\binom{i}{k}(-1)^{k-1}\Delta^{k}c_{d-k} \overset{(*)}{=} \sum_{k=0}^{i}\binom{i}{k}\sum_{l=0}^{k}(-1)^{l-1}\binom{k}{l}c_{d-k+l} = \sum_{m=0}^{i} c_{d-m}\underbrace{\sum_{\substack{k\in[i]_0,\ l\in[k]_0\\ k-l=m}}(-1)^{l-1}\binom{i}{k}\binom{k}{l}}_{\overset{(\dagger)}{=}-\mathbb{1}_{\{i=m\}}} = -c_{d-i},$$

where $(*)$ follows from [14, Lem. 2.5.2] and $(\dagger)$ follows with

$$\sum_{\substack{k\in[i]_0,\ l\in[k]_0\\ k-l=m}}(-1)^{l-1}\binom{i}{k}\binom{k}{l} = \sum_{l=0}^{i-m}(-1)^{l-1}\binom{i}{l+m}\binom{l+m}{l} = \sum_{l=0}^{i-m}(-1)^{l-1}\binom{i}{m}\binom{i-m}{l} = -\mathbb{1}_{\{i=m\}}.$$

For the general statement, we have

$$\sum_{k=0}^{i}\binom{i}{k}(-1)^{k+j-i-1}\Delta^{k+j-i}c_{d-k-(j-i)} = \sum_{k=0}^{i}\binom{i}{k}(-1)^{k-1}\Delta^{k}[(-1)^{j-i}\Delta^{j-i}c_{d-k-(j-i)}] \overset{\text{Eq. (A.4)}}{=} -[(-1)^{j-i}\Delta^{j-i}c_{d-j}].$$

□

Proof of Remark A.2. Let $\varnothing \neq I \subseteq J \subseteq [d]$ with $|I| = i$ and $|J| = j$ and recall from Eq. (A.1) that $\lambda_i = (-1)^{i-1}\Delta^{i-1}a_{d-i}$, $i \in [d]$. Hence, we have for $i < j$

$$\begin{aligned}
\lambda k_{IJ} &\overset{\text{Remark (A.1)}}{=} \sum_{k=0}^{i}\binom{i}{k}\lambda_{k+j-i} = \sum_{k=0}^{i}\binom{i}{k}(-1)^{k+j-i-1}\Delta^{k+j-i-1}a_{d-k-(j-i)}\\
&\overset{(*)}{=} \sum_{k=0}^{i}\binom{i}{k}(-1)^{k+j-i-1}\Delta^{k+j-i}\left[\sum_{l=1}^{d-k-(j-i)}a_{l-1}\right]\\
&\overset{\text{Lemma (A.3)}}{=} (-1)^{j-i-1}\Delta^{j-i}\left[\sum_{l=1}^{d-j}a_{l-1}\right]\\
&= (-1)^{j-i-1}\Delta^{j-i-1}a_{d-j},
\end{aligned}$$

where we assume in $(*)$ that the finite forward difference operator is applied to the sequence $0, a_0, a_0 + a_1, \ldots, \sum_{k=1}^{d}a_{k-1}$. Furthermore, we have from [17, p. 134]

$$\lambda = \sum_{k=0}^{d-1}a_k.$$

Finally, we obtain the claim after obtaining with a similar calculation

$$\lambda k_{II} = \sum_{l=0}^{d-1}a_l - \sum_{l=0}^{d-i-1}a_l.\square$$

Remark A.4. For an extMO distribution with Bernstein function ψ, we have

$$k_{IJ} = \frac{1}{\psi(d)}\begin{cases}\psi(d) - \psi(d-i) & I = J,\quad |I| = i,\\ (-1)^{j-i-1}\Delta^{j-i}\psi(d-j) & I \subsetneq J,\quad |I| = i,\quad |J| = j,\\ 0 & \text{else},\end{cases}$$

and

$$q_{IJ} = \begin{cases} -\psi(d-i) & I = J, \quad |I| = i, \\ (-1)^{j-i-1}\Delta^{j-i}\psi(d-j) & I \subsetneq J, \quad |I| = i, \quad |J| = j, \\ 0 & \text{else}. \end{cases}$$

Proof. The claim follows from Remark A.2 with Eq. (A.2). □

Remark A.5. For an extMO distribution from the Armageddon family with parameters $\alpha, \beta \geq 0$ with $\alpha + \beta > 0$, we have

$$k_{IJ} = \frac{1}{\beta + d\alpha} \begin{cases} \beta + d\alpha & I = J, \quad |I| = d, \\ i\alpha & I = J, \quad |I| < d, \\ \alpha & I \subsetneq J, \quad |I| = i, \quad |J| = j, \quad j - i = 1, \quad j < d, \\ \alpha + \beta & I \subsetneq J, \quad |I| = i, \quad |J| = j, \quad j - i = 1, \quad j = d, \\ \beta & I \subsetneq J, \quad |I| = i, \quad |J| = j, \quad j - i > 1, \quad j = d, \\ 0 & \text{else}, \end{cases}$$

and

$$q_{IJ} = \begin{cases} -\beta - (d-i)\alpha & I = J, \quad |I| < d, \\ \alpha & I \subsetneq J, \quad |I| = i, \quad |J| = j, \quad j - i = 1, \quad j < d, \\ \alpha + \beta & I \subsetneq J, \quad |I| = i, \quad |J| = j, \quad j - i = 1, \quad j = d, \\ \beta & I \subsetneq J, \quad |I| = i, \quad |J| = j, \quad j - i > 1, \quad j = d, \\ 0 & \text{else}. \end{cases}$$

Proof. The claim follows for $k \geq 1$ with the following calculations:

$$\psi(x) = \beta \mathbb{1}_{\{x>0\}} + \alpha x, \Delta\psi(x) = \beta \mathbb{1}_{\{x=0\}} + \alpha,$$

and

$$(-1)^{k-1}\Delta^k\psi(x) = \beta \mathbb{1}_{\{x=0\}}.$$ □

B The infinitesimal generator matrix Q^* for special cases

This appendix contains the MDCM's generator matrix representations for the exchangeable reparametrization, the extendible subclass, and the Armageddon shock subclass. The results are all direct corollaries from Appendix A and Theorem 4.1.

Remark B.1. For an exMO distribution with reparametrization $\boldsymbol{a}$, we have

$$k^*_{ij} = \frac{\binom{d-i}{j-i}}{\sum_{k=0}^{d-1} a_k} \cdot \begin{cases} \sum_{k=0}^{d-1} a_k - \sum_{k=0}^{d-i-1} a_k & i = j, \\ (-1)^{j-i-1}\Delta^{j-i-1}a_{d-j} & i < j, \\ 0 & \text{else}, \end{cases}$$

and

$$q^*_{ij} = \binom{d-i}{j-i} \cdot \begin{cases} -\sum_{k=0}^{d-i-1} a_k & i = j, \\ (-1)^{j-i-1}\Delta^{j-i-1}a_{d-j} & i < j, \\ 0 & \text{else}. \end{cases}$$

Remark B.2. For an extMO distribution with Bernstein function ψ, we have

$$k_{ij}^* = \frac{\binom{d-i}{j-i}}{\psi(d)} \cdot \begin{cases} \psi(d) - \psi(d-i) & i = j, \\ (-1)^{j-i-1}\Delta^{j-i}\psi(d-j) & i < j, \\ 0 & \text{else}, \end{cases}$$

and

$$q_{ij}^* = \binom{d-i}{j-i} \cdot \begin{cases} -\psi(d-i) & i = j, \\ (-1)^{j-i-1}\Delta^{j-i}\psi(d-j) & i < j, \\ 0 & \text{else}. \end{cases}$$

Remark B.3. For an extMO distribution from the Armageddon family with parameters $\alpha, \beta \geq 0$ with $\alpha + \beta > 0$, we have

$$k_{ij}^* = \frac{\binom{d-i}{j-i}}{\beta + d\alpha} \cdot \begin{cases} \beta + d\alpha & i = j = d, \\ i\alpha & i = j < d, \\ \alpha & i+1 = j < d, \\ \alpha + \beta & i+1 = j = d, \\ \beta & i+1 < j = d, \\ 0 & \text{else}, \end{cases}$$

and

$$q_{ij}^* = \binom{d-i}{j-i} \cdot \begin{cases} -\beta - (d-i)\alpha & i = j < d, \\ \alpha & i+1 = j < d, \\ \alpha + \beta & i+1 = j = d, \\ \beta & i+1 < j = d, \\ 0 & \text{else}. \end{cases}$$

C Runtime boundaries for the MDCM and LFM

The following section compares the runtime of the MDCM algorithm to that of the LFM algorithm on a theoretical basis. For this, recall that the MDCM samples discrete transitions with conditionally independent exponential waiting times until all components are extinct, determining the order with a random shuffling afterward, and the LFM samples discrete transitions with exponential waiting times until the compound Poisson subordinator surpassed all unit exponential barrier values. We provide formulas to calculate or bound the expected number of sampled waiting times for both models. These confirm that the expected number of waiting times is bounded for the MDCM but can become arbitrarily large for the LFM.

Proposition C.1. *Let $d \geq 2$ and consider the LFM from* Theorem 2.4 *for an extMO distribution with Bernstein function $\psi(x) = c(1 - \kappa(x))$ for $c > 0$ and a completely monotone function κ*, i.e., *ψ is the Lévy exponent of a pure-jump compound Poisson subordinator with intensity c and whose jumps have the Laplace function κ. Furthermore, let K_i be the number of jumps required for surpassing the ith barrier value and K be the number of jumps required for surpassing all barrier values. Then, K_i is geometrically distributed with success probability $1 - \kappa(1)$ and*

$$\frac{1}{1-\kappa(1)} \leq \mathbb{E}[K] \leq \frac{d}{1-\kappa(1)}.$$

Proof. We use the notation from Theorem 2.4, and let Λ be defined by

$$\Lambda_t = \sum_{j=1}^{N_t} X_j, \quad t \geq 0,$$

for a Poisson process N and, independent thereof, iid jumps $\{X_j : j \in \mathbb{N}\}$ with Laplace function κ, and define

$$K_i := \min\left\{k \in \mathbb{N} : E_i \leq \sum_{j=1}^{k} X_j\right\}, \quad i \in [d].$$

We can derive with a simple calculation that K_i, $i \in [d]$, have geometric distributions with success probability $1 - \kappa(1)$ by using the tower property and conditioning on $\{X_j : j \in \mathbb{N}\}$ for the survival function of K_i. Consequently, we obtain the claim by using $K = \max\{K_1, \ldots, K_d\}$ and

$$\frac{1}{1-\kappa(1)} = \mathbb{E}[K_1] \leq \mathbb{E}[K] \leq \sum_{i=1}^{d} \mathbb{E}[K_i] = d \cdot \mathbb{E}[K_1] = \frac{d}{1-\kappa(1)}. \qquad \square$$

Proposition C.2. *Let $d \geq 2$ and consider the MDCM from* Theorem 4.1 *for an infinitesimal generator matrix Q^*. Furthermore, let M be the number of transitions until the Markov chain process is absorbed, i.e., until all components are dead. Then*

$$\mathbb{E}[M] = \vec{e}_0^{\top} S \cdot \mathbf{1} \leq d,$$

where $S = (s_{ij})_{i,j \in [d-1]_0}$ is the fundamental matrix of the embedded Markov chain, (implicitly) defined by

$$(S^{-1})_{ij} = \begin{cases} 1 & i = j, \\ q^*_{ij} & \\ \hline q^*_{ii} & i \neq j. \end{cases}$$

Proof. First note that the transition matrix $T = (t_{ij} : i, j \in [d]_0)$ of the embedded Markov process if defined by

$$t_{ij} = \begin{cases} 1 & i = j = d, \\ -\dfrac{q^*_{ij}}{q^*_{ii}} & j > i, \\ 0 & j \leq i, \end{cases}$$

and that d is the only recurrent, absorbing state. By using [4, Thm. 6.2.3], we conclude that $S = [\mathrm{Id} - T]^{-1}$ is the *fundamental matrix* of the embedded chain and obtain the claim using the arguments from [4, Sec. 6.2.1]:

$$\mathbb{E}[M] = \sum_{j=0}^{d-1} \sum_{n=0}^{\infty} (T^n)_{0j} = \sum_{j=0}^{d-1} \left(\sum_{n=0}^{\infty} T^n \right)_{0j} = \sum_{j=0}^{d-1} s_{0j}.$$

We obtain the upper bound from the observation that each transition increases the death count by at least one, limiting the number of transitions until absorption by d. $\square$

We conclude from Propositions C.1 and C.2 that while the MDCM cannot have more than d transitions, the expected number of transitions in the LFM is not bounded and can be significantly larger when the probability of small jumps in the corresponding Lévy subordinator is high. We created Figure A1 to demonstrate these differences for the exponential family. The plot highlights that the MDCM requires significantly fewer expected transitions than the LFM if the expected jump sizes are tiny.

D Statistical tests for extMO distributions

A challenge for implementing simulation algorithms is developing statistical tests to verify their goodness-of-fit. For example, R uses the *Dvoretzky–Kiefer–Wolfowitz inequality* with the tight constant derived in [20] for testing their univariate distribution algorithms. Other popular tests include the *Kolmogorov–Smirnov tests* and the χ^2*-test*. For multivariate distributions, a recent advance is the derivation of a tight constant for the multivariate *Dvoretzky–Kiefer–Wolfowitz inequality*, see [21].

We propose a simple alternative approach using the *min-stability* of MO distribution: Consider iid d-variate extMO distributed random vectors $\boldsymbol{\tau}_k = (\tau_{k,1}, \ldots, \tau_{k,d})$, $k \in \{1, \ldots, n\}$, with corresponding Bernstein function ψ and define

$$U_k \coloneqq 1 - \exp\{-\psi(d) \cdot \min_{i \in [d]} \tau_{k,i}\}, \quad k \in [n].$$

A straightforward calculation shows that the overall minimum of $\boldsymbol{\tau}_1$ has an exponential distribution with rate $\psi(d)$. Hence, $U_1, \ldots, U_n$ is an iid standard uniform sequence. Consequently, we can apply *Kolmogorov–Smirnov tests* for goodness-of-fit testing, see [18] and [29]. We employed this method to extensively test all implementations discussed in Section 6 using a p-value threshold of 1% and a *Bonferroni correction* for the total number of tests.

E Alias method for sampling on finite-state spaces

The following section shortly sketches the *alias method* for sampling discrete random variables on finite-state spaces; see [31] for the details. Let $\{p_k : k \in \{1, \ldots, n\}\}$ be a probability counting measure for the finite-state space $[n] = \{1, \ldots, n\}$. Suppose there exist mappings $f : [n] \mapsto [n]$ and $q : [n] \mapsto [0, 1]$ such that

$$p_k = \frac{1}{n} \cdot \left[(1 - q(k)) + \sum_{j \in [n]} q(j) \mathbb{1}_{\{f(j)=k\}} \right]. \tag{$*$}$$

Consider a probability space supporting the following random variables: a uniform random variable Y on $[n]$ and, independent thereof, a uniform random variable U on $[0, 1]$. Define I by $I = \mathbb{1}_{\{U \le q(Y)\}}$ such that I conditioned on Y has a Bernoulli distribution with success probability $q(Y)$. Furthermore, define X by

$$X \coloneqq \begin{cases} f(Y) & I = 1, \\ Y & I = 0, \end{cases}$$

and a straightforward calculation shows that $\mathbb{P}(X = k) = p_k$.

Note that the mappings f and q fulfilling condition $(*)$ exist but are not unique. We sketch one possibility in Algorithm 2 that uses a strategy that recursively moves probability mass from the element

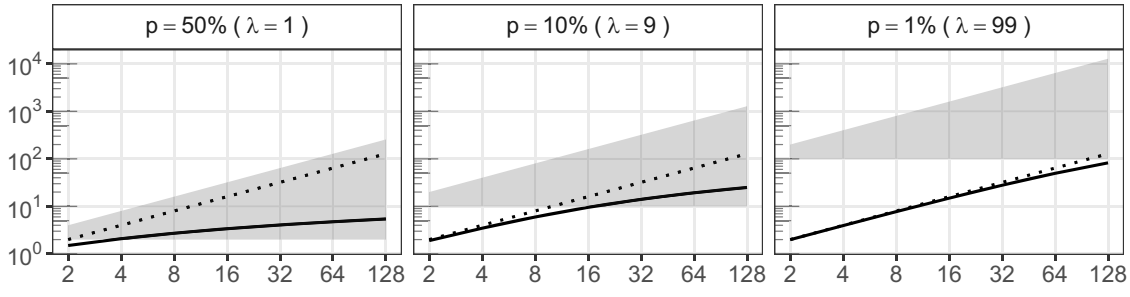

Figure A1: The expected number of transitions in the MDCM (solid line) and expected number of transitions in the LFM (gray area) for the Exponential family; the dotted line marks the identity.

with the largest mass to that with the least mass until all probabilities are equal. By reflecting these probability mass transfers adequately in the success probabilities of the second-order Bernoulli experiment and conditional element-switch, the overall probabilities remain unchanged.

Algorithm 2. Create mappings f and q for the alias method.

```
Require: (Vector) argument p with probabilities
procedure CREATE_ALIAS_MAPPINGS(p)
  n = size(p)
  w = p
  f = sequence(1, n), q = zeros(n)
  while max(w) > 1. / n do
    l = argmin(w), k = argmax(w)
    f[l] = k
    q[l] = (1. / n - w[l]) * n
    w[k] -= (1. / n - w[l])
    w[l] = 1. / n
  end while
  return f, q
end procedure
```

The appeal of the alias method is that, given the mappings f and q, it requires at most the generation of two uniform random variables on $\{1, \ldots, n\}$ and $[0, 1]$, at most two reference operations, and a comparison per sample.

References

[1] Aldous, D. J. (1985). Exchangeability and related topics. *École d'Été de Probabilités de Saint-Flour XIII – 1983*. Berlin/Heidelberg, Germany: Springer. doi: 10.1007/bfb0099421.

[2] Arnold, B. C. (1975). A characterization of the exponential distribution by multivariate geometric compounding. *Sankhyā: The Indian Journal of Statistics, Series A*, *37*(1), 164–173, 0581-572X.

[3] Berg, C., Christensen, J. P. R. & Ressel, P. (1984). *Harmonic analysis on semigroups*. New York, NY: Springer. doi: 10.1007/978-1-4612-1128-0.

[4] Brémaud, P. (2020). *Markov chains* (2nd ed.). Cham, Switzerland: Springer. doi: 10.1007/978-3-030-45982-6.

[5] Brigo, D., Mai, J.-F. & Scherer, M. (2016). Markov multivariate survival indicators for default simulation as a new characterization of the Marshall–Olkin law. *Statistics & Probability Letters*, *114*, 60–66. doi: 10.1016/j.spl.2016.03.013.

[6] Eddelbuettel, D. & François, R. (2011). Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, *40*(8), 1–18. doi: 10.18637/jss.v040.i08.

[7] Fernández, L. (2015). *Selected topics in financial engineering: First-exit times and dependence structures of Marshall–Olkin kind* (PhD thesis). Bilbao, Spain: University of the Basque Country. https://addi.ehu.es/handle/10810/16067.

[8] Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Alken, P., ..., Ulerich, R. (2009). GNU scientific library reference manual (3rd ed.). Network theory Ltd. ISBN: 978-0-9546120-7-8.

[9] Gautschi, W. (2012). *Numerical analysis* (2nd ed.). Boston, MA: Birkhäuser. doi: 10.1007/978-0-8176-8259-0.

[10] Hester, J. & Vaughan, D. (2021). *bench: High precision timing of R expressions, R package version 1.1.2*. https://CRAN.R-project.org/package=bench.

[11] ISO C++ Standards Committee. (2017). *ISO/IEC 14882:2017 Programming languages – C*. International Organization for Standardization (ISO).

[12] IEEE-SA Standards Board. (2017). *IEEE standard for floating-point arithmetic*. IEEE Std 754-2019 (Revision of IEEE 754-2008), 1–84. doi: 10.1109/IEEESTD.2019.8766229.

[13] Mai, J.-F. & Scherer, M. (2009). Lévy-frailty copulas. *Journal of Multivariate Analysis*, *100*(7), 1567–1585. doi: 10.1016/j.jmva.2009.01.010.

[14] Mai, J.-F. (2010). *Extendibility of Marshall–Olkin distributions via Lévy subordinators and an application to portfolio credit risk* (PhD thesis). Munich, Germany: Technical University of Munich. https://nbn-resolving.org/urn:nbn:de:bvb:91-diss-20100628-969547-1-8.
[15] Mai, J.-F. & Scherer, M. (2013). Sampling exchangeable and hierarchical Marshall–Olkin distributions. *Communications in Statistics-Theory and Methods*, *42*(4), 619–632. doi: 10.1080/03610926.2011.615437.
[16] Mai, J.-F. (2014). *Multivariate exponential distributions with latent factor structure and related topics* (Habilitation). Munich, Germany: Technical University of Munich. https://nbn-resolving.org/urn:nbn:de:bvb:91-diss-20140220-1236170-0-7.
[17] Mai, J.-F. & Scherer, M. (2017). *Simulating copulas: Stochastic models, sampling algorithms and applications* (2nd ed.). Singapore: World Scientific Publishing. doi: 10.1142/10265.
[18] Marsaglia, G., Tsang, W. W. & Wang, J. (2003). Evaluating Kolmogorov's distribution. *Journal of Statistical Software*, *8*(18), 1–4. doi: 10.18637/jss.v008.i18.
[19] Marshall, A. W. & Olkin, I. (1967). A multivariate exponential distribution. *Journal of the American Statistical Association*, *62*(317), 30–44. doi: 10.1080/01621459.1967.10482885.
[20] Massart, P. (1990). The tight constant in the Dvoretzky–Kiefer–Wolfowitz inequality. *The Annals of Probability*, *18*(3), 1269–1283. doi: 10.1214/aop/1176990746.
[21] Naaman, M. (2021). On the tight constant in the multivariate Dvoretzky–Kiefer–Wolfowitz inequality. *Statistics & Probability Letters*, *173*, 1–8. doi: 10.1016/j.spl.2021.109088.
[22] Nelsen, R. B. (2006). *An introduction to copulas* (2nd ed.). New York, NY: Springer. doi: 10.1007/0-387-28678-0.
[23] Piessens, R., de Doncker-Kapenga, E., Überhuber, C. W. & Kahaner, D. K. (1983). *Quadpack*. Berlin/Heidelberg, Germany: Springer. doi: 10.1007/978-3-642-61786-7.
[24] R Core Team. (2021). *R: a language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.
[25] Resnick, S. I. (1987). *Extreme values, regular variation, and point processes*. New York, NY: Springer. doi: 10.1007/978-0-387-75953-1.
[26] Ruckdeschel, P., Kohl, M., Stabla, T. & Camphausen, F. (2006). S4 classes for distributions. *R News*, *6*(2), 2–6. https://CRAN.R-project.org/doc/Rnews/Rnews_2006-2.pdf.
[27] Sato, K.-I. (1999). *Lévy processes and infinitely divisible distributions*. Cambridge, UK: Cambridge University Press.
[28] Schilling, R. L., Song, R. & Vondracek, Z. (2012). *Bernstein functions* (2nd ed.). Berlin/Boston: De Gruyter. doi: 10.1515/9783110269338.
[29] Simard, R. & L'Ecuyer, P. (2011). Computing the two-sided Kolmogorov-Smirnov distribution. *Journal of Statistical Software*, *39*(11), 1–18. doi: 10.18637/jss.v039.i11.
[30] Sun, Y., Mendoza-Arriaga, R. & Linetsky, V. (2017). Marshall–Olkin distributions, subordinators, efficient simulation, and applications to credit risk. *Advances in Applied Probability*, *49*(2), 481–514. doi: 10.1017/apr.2017.10.
[31] Walker, A. J. (1977). An efficient method for generating discrete random variables with general distributions. *ACM Transactions on Mathematical Software*, *3*(3), 253–256. doi: 10.1145/355744.355749.