Generating function methods for the efficient computation of expected allocations

Christopher Blier-Wong[*], Hélène Cossette, and Etienne Marceau

École d'actuariat, Université Laval, Québec, Canada

May 18, 2022[†]

Abstract

Consider a risk portfolio with aggregate loss random variable $S = X_1 + \cdots + X_n$ defined as the sum of the n individual losses $X_1, \dots, X_n$. The expected allocation, $E[X_i \times 1_{\{S=k\}}]$, $i = 1, \dots, n$ and $k \in \mathbb{N}$, is a vital quantity for capital allocation and peer-to-peer insurance. For example, one uses this value to compute peer-to-peer contributions under the conditional mean risk-sharing rule and risk contributions under Euler risk allocation paradigm. This paper introduces an ordinary generating function for expected allocations, a power series representation of the expected allocation of an individual risk given the total risks in the portfolio when all risks are discrete. First, we provide a simple relationship between the ordinary generating function for expected allocations and the probability generating function. Then, leveraging properties of ordinary generating functions, we reveal new theoretical results on closed-formed solutions to capital allocation problems, especially when dealing with frequency distributions in the $(a, b, 0)$ class. Then, we present an efficient algorithm to recover the expected allocations using the fast Fourier transform, providing a new practical tool to compute expected allocations quickly. The latter approach is exceptionally efficient for a portfolio of independent risks.

Keywords: Capital allocation, conditional mean risk sharing, fast Fourier transform, generating functions

1 Introduction

Consider a portfolio of n risks $\boldsymbol{X} = (X_1, \dots, X_n)$ where each random variable (rv) has a lattice-type support $h\mathbb{N} = \{hk | k \in \mathbb{N}, h \in \mathbb{R}^+\}$, where $\mathbb{N}$ is the set of non-negative integers $\{0, 1, 2, \dots\}$. The joint cumulative distribution function (cdf) is $F_{\boldsymbol{X}}(\boldsymbol{x})$, for $\boldsymbol{x} \in \{h\mathbb{N}\} \times \cdots \times \{h\mathbb{N}\} = \{h\mathbb{N}\}^n$ and marginal cdfs are noted $F_{X_i}(x) = \mathbb{P}(X_i \leq x)$, for $x \in h\mathbb{N}$ and $i \in \{1, \dots, n\}$. The rv representing the portfolio aggregate loss is $S = X_1 + \cdots + X_n$.

[*]Corresponding author, christopher.blier-wong.1@ulaval.ca

[†]Preprint presented to the 2022 Actuarial Colloquia

The objective of this paper is to study methods based on ordinary generating functions to compute the expectation which we present in the following definition.

Definition 1.1 (Expected allocation). Let $\boldsymbol{X}$ be a vector of positive random variables, and $S = X_1 + \cdots + X_n$. The expected allocation of X_i and the total outcome s is defined as $E[X_i \times 1_{\{S=s\}}]$, for $i \in \{1, \ldots, n\}$ and $s \geq 0$.

Although rarely considered directly, expected allocations play an important role in peer-to-peer insurance pricing and for risk allocation based on Euler's rule. In peer-to-peer insurance pricing schemes, one is interested in computing the contribution for a participant according to a risk sharing rule. The conditional mean risk sharing rule, studied in, e.g., [Denuit and Dhaene, 2012], is a popular choice, where the price for the ith participant is the expected contribution of risk $X_i, i \in \{1, \ldots, n\}$ given that the actual loss S is s, that is,

$$E[X_i|S = s] = \frac{E[X_i \times 1_{\{S=s\}}]}{\Pr(S = s)}, \quad s \in h\mathbb{N}. \tag{1}$$

The authors of [Denuit et al., 2021a] provide a list of desirable properties for risk sharing rules, and prove that the conditional mean risk sharing rule satisfies 11 out of the 12 desirable properties.

Another application of expected allocations in peer-to-peer insurance happens when the pool transfers parts of its risks to reinsurance companies. Fix two levels $0 < \ell_1 < \ell_2 < \infty$ for $\ell_1 \in h\mathbb{N}$ and $\ell_2 \in h\mathbb{N}$. Consider a pool which retains a first level of risk up to level $0 < \ell_1 < \infty$, then transfers the losses from ℓ_1 to ℓ_2 to a first reinsurer, and finally all losses above level ℓ_2 to a second reinsurer. Then one is interested in computing the contribution of a participant to the retained losses and to the transferred losses in each layer, that is,

$$E[X_1 \times 1_{\{S \leq \ell_1\}}] = \sum_{x \in \{0, h, \ldots, \ell_1\}} E[X_1 \times 1_{\{S=x\}}] \tag{2}$$

$$E[X_1 \times 1_{\{\ell_1 < S \leq \ell_2\}}] = \sum_{x \in \{\ell_1+h, \ldots, \ell_2\}} E[X_1 \times 1_{\{S=x\}}] = E[X_1 \times 1_{\{S \leq \ell_2\}}] - E[X_1 \times 1_{\{S \leq \ell_1\}}] \tag{3}$$

$$E[X_1 \times 1_{\{S > \ell_2\}}] = \sum_{x \in \{\ell_2+h, \ldots\}} E[X_1 \times 1_{\{S=x\}}] = E[X_1] - E[X_1 \times 1_{\{S \leq \ell_2\}}]. \tag{4}$$

Such expressions also appear when computing the allocations to different layers of collateralized debt obligations, see, for instance, [Tasche, 2007] for details.

Expected allocations also appear in risk allocation based on Euler's rule. Regulatory capital requirements are typically risk measures based on the aggregate rv of the portfolio of an insurance company. One risk measure of theoretical and practical interest is the Tail Value at Risk (TVaR). Risk allocation is an important research area in actuarial science, quantitative risk management and operations research, which aims to compute the contribution of each risk, based on the total required (or available) capital. When one uses the TVaR as a regulatory capital requirement risk measure, one may perform TVaR-based capital allocation along with the Euler risk sharing paradigm.

Define the generalized inverse of S at level $\kappa, 0 < \kappa < 1$ by

$$F_S^{-1}(\kappa) = \inf_{x \in \mathbb{R}} \left\{F_S(x) \geq \kappa\right\}.$$

Define the Value at Risk as $VaR_\kappa(S) = F_S^{-1}(\kappa)$ and the Tail Value at Risk at level κ as

$$TVaR_\kappa(S) = \frac{1}{1-\kappa}\int_\kappa^1 F_S^{-1}(u)\,\mathrm{d}u = \frac{E[S \times 1_{\{S>VaR_\kappa(S)\}}] + VaR_\kappa(S)(F_S(VaR_\kappa(S)) - \kappa)}{1-\kappa}.$$

One can further define the TVaR-based capital allocation rule, introduced by [Tasche, 1999], is

$$TVaR_\kappa(X_i; S) = \frac{E\left[X_i \times 1_{\{S>s_0\}}\right] + E\left[X_i \times 1_{\{S=s_0\}}\right]\beta}{1-\kappa}, \tag{5}$$

where $s_0 = VaR_\kappa(S)$ and

$$\beta = \begin{cases} \frac{F_S(s_0)-\kappa}{\Pr(S=s_0)}, & \text{if } \Pr(S = s_0) > 0 \\ 0, & \text{otherwise} \end{cases}.$$

One can compute

$$E\left[X_i \times 1_{\{S>s\}}\right] = E[X_i] - E\left[X_i \times 1_{\{S\leq s\}}\right] = E[X_i] - \sum_{j\in\{h,2h,\ldots,s\}} E\left[X_i \times 1_{\{S=j\}}\right],$$

for $s \in h\mathbb{N}$.

The Euler-based TVaR decomposition is a top-down risk allocation method of capital allocation. Indeed, by the additive property of the expected value, the full allocation property [McNeil et al., 2015] holds:

$$TVaR_\kappa(S) = \sum_{i=1}^n TVaR_\kappa(X_i; S).$$

Equations (1) to (5) require the computation of the expected allocation $E\left[X_i \times 1_{\{S=s\}}\right]$ for $s \in h\mathbb{N}$. One therefore seeks an efficient method to compute these values; this is the main objective of the current paper.

In the actuarial science literature, one finds two common approaches to compute expected allocations. Under the first approach, one uses the direct method for computing expected allocations is through summation or integration. Let $S_{-1} = X_2 + \cdots + X_n$, then one may compute

$$E\left[X_1 \times 1_{\{S=s\}}\right] = \sum_{x\in\{0,h,2h,\ldots,s\}} x f_{X_1,S_{-1}}(x, s-x)$$

for $s \in h\mathbb{N}$, where the summation is replaced by an integral in the continuous case. In Section 5 of [Bargès et al., 2009], the authors use this approach to compute TVaR-based allocations for a mixture of exponential distributions when the dependence structure is a FGM copula. The second approach is used notably in [Furman and Landsman, 2005, Furman and Landsman, 2008], where the authors use the size-biased transform of a random variable to compute the expected allocations. See also [Arratia et al., 2019] for a review on the size-biased transform and its applications in several contexts. The method states that

$$E\left[X_1 \times 1_{\{S=s\}}\right] = E[X_1]\Pr(\widetilde{X}_1 + S_{-1} = s), \tag{6}$$

where $\widetilde{X}_1$ is the size-biased transform of X_1 with probability mass function (pmf)

$$f_{\widetilde{X}_1}(x) = x f_{X_1}(x)/E[X_1], \quad x \in h\mathbb{N}. \tag{7}$$

The authors of [Denuit and Dhaene, 2012] use the size-biased transform method to derive properties of the conditional mean risk sharing rule.

The main result of this paper is a representation of the ordinary generating function (OGF) for expected allocations in terms of the joint probability generating functions (pgf) of the random vector $\boldsymbol{X}$. We then present efficient algorithms to compute the expected allocation $E[X_i \times 1_{\{S=s\}}]$, for $i \in \{1, \ldots, n\}$ and $s \in h\mathbb{N}$. In some cases, one can extract the values explicitly from the OGF of expected allocations, expressed either as a function of the pmf of S or as a convolution operation. One can use efficient algorithms like the Legendre polynomial or the fast Fourier transform (FFT) to recover the expected allocations directly from the OGF.

Our approach provides a generating function method, as opposed to a direct computation method. Generating function methods, sometimes referred to as transform methods, are also used to compute the pmf of aggregate rvs or compound distributions, see, e.g., [Embrechts et al., 1993], [Grubel and Hermesmeier, 1999], and [Embrechts and Frei, 2009] for applications of transform methods in actuarial science.

To simplify the notation in the theory developed in the remainder of this paper, we set $h = 1$, that is, we consider only rvs which have integer support. Therefore, for the remainder of this paper, the n-variate random vector $\boldsymbol{X}$ takes values in $\mathbb{N}^n$. One may transform a rv with lattice-type support $h\mathbb{N}$ into one of integer support $\mathbb{N}$ by multiplying the rv by the constant h^{-1}. By linearity of the expected allocation, one may easily recover expected allocations for the original rv.

Typically, one uses conditional mean risk sharing and capital allocation for small-sized pools or few lines of businesses. However, the efficient methods based on OGFs proposed in this paper enable one to use these techniques even for a large portfolio of risks. It follows that traditional insurers will have the ability to perform risk allocation at the customer level. The paper is organized as follows. In Section 2, we present the OGF for expected allocations and provide a relationship with the pgf. We also provide an efficient method to extract the expected allocations from the OGF. Section 4 presents expressions for the expected allocation in the case of (compound) $(a, b, 0)$ distributions. In Section 5, we provide three results when the components of the random vector are dependent. Section 6 discusses further generalizations of our results for continuous rvs.

2 Ordinary generating functions for expected allocations

2.1 Ordinary generating functions

Ordinary generating functions are a useful mathematical tool since they capture every value of a sequence into one formula. See Chapter 7 of [Graham et al., 1994] or the monograph [Wilf, 2006] for details on generating functions, and Chapter 3 of [Sedgewick and Flajolet, 2013] for efficient algorithms to extract the values of the sequence. Following Section 3.1 of [Sedgewick and Flajolet, 2013], we provide the following definition of ordinary generating functions.

Definition 2.1 (Ordinary generating function). For a sequence $\{a_k\}_{k\in\mathbb{N}_0}$, the function

$$A(z) = \sum_{k=0}^{\infty} a_k z^k, \quad |z| \leq 1, \tag{8}$$

is its ordinary generating function (OGF). We use the notation $[z^k]A(z)$ to refer to the coefficient a_k, $k \in \mathbb{N}$.

In the following theorem, we summarize the relevant operations that one can perform on generating functions (see Theorem 3.1 and Table 3.2 of [Sedgewick and Flajolet, 2013] for details).

Theorem 2.2. *If $A(z) = \sum_{k=0}^{\infty} A_k z^k$ and $B(z) = \sum_{k=0}^{\infty} b_k z^k$ are two OGFs, then the following operations produce OFGs with the corresponding sequences:*

1. *Addition* $A(z) + B(z) = \sum_{k=1}^{\infty}(a_k + b_k)z^k$.
2. *Right shift* $zA(z) = \sum_{k=1}^{\infty} a_{k-1}z^k$.
3. *Index multiply* $A'(z) = \sum_{k=0}^{\infty}(k+1)a_{k+1}z^k$.
4. *Convolution* $A(z)B(z) = \sum_{k=0}^{\infty}\left(\sum_{j=0}^{k} a_j b_{k-j}\right) z^k$.
5. *Partial sum* $A(z)/(1-z) = \sum_{k=0}^{\infty}\left(\sum_{j=0}^{k} a_j\right) z^k$.

When $a_k \geq 0$ for $k \in \mathbb{N}$ and $\sum_{k=0}^{\infty} a_k = 1$, then the OGF is the probability generating function (pgf), an important tool in all areas of probability, actuarial science, and statistics (see, for example, Section 5.1 in [Grimmett and Stirzaker, 2020], and Sections 1.2 and 2.4 in [Panjer and Willmot, 1992]).

In this paper we will rely on a multivariate ordinary generating function capturing the values of the pmf of a discrete random vector. The pgf of a vector of discrete rvs $\boldsymbol{X} = (X_1, \ldots, X_n)$, as described in Section 34.2.1 of [Johnson et al., 1997], is

$$\mathcal{P}_{\boldsymbol{X}}(t_1, \ldots, t_n) = E\left[t_1^{X_1} \ldots t_n^{X_n}\right] = \sum_{k_1=0}^{\infty} \cdots \sum_{k_n=0}^{\infty} t_1^{k_1} \ldots t_n^{k_n} f_{\boldsymbol{X}}(k_1, \ldots, k_n), \quad |t_j| \leq 1, j \in \{1, \ldots, n\}.$$

As stated in Theorem 1 of [Wang, 1998], a useful property of multivariate pgfs is that one obtains the pgf of the aggregate random variable S through the relation

$$\mathcal{P}_S(t) = E\left[t^S\right] = E\left[t^{X_1+\cdots+X_n}\right] = E\left[t^{X_1} \ldots t^{X_n}\right] = \mathcal{P}_{\boldsymbol{X}}(t, \ldots, t), \quad |t| \leq 1. \tag{9}$$

When the components of $\boldsymbol{X}$ are independent, (9) becomes

$$\mathcal{P}_S(t) = \mathcal{P}_{X_1}(t) \times \cdots \times \mathcal{P}_{X_n}(t), \tag{10}$$

for $|t| \leq 1$. One may use pgfs to extract factorial moments, mixed moments and pmfs, see Section 4.2 of [Wang, 1998] or Section 5.1 of [Grimmett and Stirzaker, 2020] for details.

2.2 An ordinary generating function for expected allocations

In this paper our interest is that of computing expected allocations, hence we define the function $\mathcal{P}_S^{[i]}(t)$ as the OGF of the sequence of expected allocations for the random variable X_i, that is,

$$\mathcal{P}_S^{[i]}(t) := \sum_{k=0}^{\infty} t^k E\left[X_i \times 1_{\{S=k\}}\right],$$

for $i \in \{1, \ldots, n\}$. For the remainder of this paper, unless otherwise specified, we develop formulas for $i = 1$. One may obtain the other expected allocations by appropriate reindexing. The following theorem is at the centre of the results in the remainder of this paper and provides a link between the OGF of the expected allocations of the random variable X_1 and the multivariate pgf.

Theorem 2.3. *Let $\boldsymbol{X}$ be a vector of rvs with $\mathcal{P}_{\boldsymbol{X}}(t_1, \ldots, t_n)$ as the joint pgf of $\boldsymbol{X}$. The OGF for the sequence of expected allocations $\{E[X_1 \times 1_{\{S=k\}}]\}_{k\in\mathbb{N}}$ is given by*

$$\mathcal{P}_S^{[1]}(t) = \left[t_1 \times \frac{\partial}{\partial t_1} P_{\boldsymbol{X}}(t_1, \ldots, t_n)\right]_{t_1=\cdots=t_n=t}.$$

Proof. Define $S_{-1} = X_2 + \cdots + X_n$. Then, the pgf of (X_1, S_{-1}) is

$$\mathcal{P}_{X_1,S_{-1}}(t_1, t_{-1}) = E\left[t_1^{X_1} t_{-1}^{X_2+\cdots+X_n}\right] = \mathcal{P}_{\boldsymbol{X}}(t_1, t_{-1}, \ldots, t_{-1}).$$

Then, we define

$$\mathcal{P}_{X_1,S_{-1}}^{[1]}(t_1, t_{-1}) = t_1 \times \frac{\partial}{\partial t_1}\mathcal{P}_{X_1,S_{-1}}(t_1, t_{-1}) = E\left[X_1 t_1^{X_1} t_{-1}^{S_{-1}}\right] = \sum_{k_1=0}^{\infty}\sum_{k_2=0}^{\infty} k_1 t_1^{k_1} t_{-1}^{k_2} f_{X_1,S_{-1}}(k_1, k_2). \tag{11}$$

Finally, let $\mathcal{P}_S^{[1]}(t) = \mathcal{P}_{X_1,S_{-1}}^{[1]}(t, t)$, which is given by

$$\begin{aligned}\mathcal{P}_S^{[1]}(t) = E\left[X_1 t^S\right] &= \sum_{k_1=0}^{\infty}\sum_{k_2=0}^{\infty} k_1 t^{k_1} t^{k_2} f_{X_1,S_{-1}}(k_1, k_2) \\ &= \sum_{k=0}^{\infty} t^k \sum_{k_1=0}^{k} k_1 f_{X_1,S_{-1}}(k_1, k - k_1) = \sum_{k=0}^{\infty} t^k E\left[X_1 \times 1_{\{S=k\}}\right],\end{aligned} \tag{12}$$

where (12) is the power series representation of expected allocations, as desired. □

From the uniqueness theorem of pgfs (see, for instance, Section 5.1 of [Grimmett and Stirzaker, 2020]), one can recover the values of $E\left[X_1 \times 1_{\{S=k\}}\right]$, $k \in \mathbb{N}$ by differentiating

$$[t^k]\mathcal{P}_S^{[1]}(t) = E\left[X_1 \times 1_{\{S=k\}}\right] = \frac{1}{k!}\left.\frac{\mathrm{d}^k}{\mathrm{d}t^k}\mathcal{P}_S^{[1]}(t)\right|_{t=0},$$

or using an algorithm to extract the coefficients of a polynomial. The entire Section 2.3 provides a method using FFT to extract the coefficients from the OGF for expected allocations. Consequently, (12) is a powerful tool to capture every expected allocation for X_1 within a single function.

An especially convenient corollary holds for the allocation of a rv which is independent from the remaining risks.

Corollary 2.4. *If X_1 and S_{-1} are independent, then we have*

$$\mathcal{P}_S^{[1]}(t) = t\mathcal{P}'_{X_1}(t)\mathcal{P}_{S_{-1}}(t). \tag{13}$$

Remark 2.5. Our method provides a simpler proof of the size-biased transform method of computing expected allocations for discrete rvs. With $\widetilde{X}$ representing the size-biased transform of the rv X, along with the definition of the size-biased transform in (7), the relationship between the pgfs of X and $\widetilde{X}$ is

$$\mathcal{P}_{\widetilde{X}}(t) = E\left[t^{\widetilde{X}}\right] = \sum_{k=0}^{\infty} t^k f_{\widetilde{X}}(k) = \frac{t}{E[X]} \sum_{k=0}^{\infty} k t^{k-1} f_X(k) = \frac{t}{E[X]} \frac{\mathrm{d}}{\mathrm{d}t} \sum_{k=0}^{\infty} t^k f_X(k) = \frac{t}{E[X]} P_X'(t).$$

One can obtain the pgf of $\widetilde{X}$ by applying operation 2 (right shift) and 3 (index multiply) of OGFs. From (12), we have

$$\mathcal{P}_S^{[1]}(t) = E\left[X_1 t^S\right] = \sum_{k=0}^{\infty} t^k \sum_{k_1=0}^{k} k_1 f_{X_1,S_{-1}}(k_1, k-k_1) = \sum_{k=0}^{\infty} t^k \sum_{k_1=0}^{k} \frac{f_{\widetilde{X}_1,S_{-1}}(k_1, k-k_1)}{E[X_1]},$$

then $E[X_1]\mathcal{P}_S^{[1]}(t)$ is the pgf of $\widetilde{X}_1 + S_{-1}$, so (6) follows immediately. △

Next we turn our attention to computing the expected cumulative allocation defined by

$$E\left[X_i \times 1_{\{S \leq k\}}\right] = \sum_{j=0}^{k} E\left[X_i \times 1_{\{S=j\}}\right], \tag{14}$$

using a more efficient method to compute the expectations. Once again, we find an OGF for this quantity.

Theorem 2.6. *Let $\boldsymbol{X}$ be a vector of rvs and S be the aggregate rv. We note $\mathcal{P}_{\boldsymbol{X}}(t_1, \ldots, t_n)$ as the joint pgf of $\boldsymbol{X}$. The function*

$$\mathcal{P}_S^{[1]}(t)/(1-t) = \frac{1}{1-t}\left[t_1 \times \frac{\partial}{\partial t_1} P_{\boldsymbol{X}}(t_1, \ldots, t_n)\right]_{t_1=\cdots=t_n=t}$$

is the ogf of the sequence of expected allocations $\{E[X_1 \times 1_{\{S \leq k\}}]\}_{k \in \mathbb{N}}$.

Proof. Applying operation 5 (partial sum) of OGFs, we have

$$\frac{\mathcal{P}_S^{[1]}(t)}{1-t} = \sum_{k=0}^{\infty} t^k \left(\sum_{j=0}^{k} E\left[X_1 \times 1_{\{S=j\}}\right]\right) = \sum_{k=0}^{\infty} t^k E\left[X_1 \times 1_{\{S \leq k\}}\right]. \tag{15}$$

□

Once again, one may differentiate the OGF for cumulative expected allocations in (15) to extract the cumulative expected allocations, or use the FFT method that follows.

2.3 Outline of the FFT-based computation strategy

Equipped with an OGF for expected allocations, one may seek to solve for the expected allocations analytically. However, in most cases this will be tedious or impossible. One therefore requires numerical algorithms to compute the expected allocations. We now provide an algorithm to recover the expected allocations.

A significant advantage of working with pgfs (and more generally, with OGFs) is that the fast Fourier transform algorithm of [Cooley and Tukey, 1965] provides an efficient method to extract the values of OGFs, as explained in Chapter 30 of [Cormen et al., 2009]. See also [Embrechts et al., 1993] for applications of the fast Fourier transform in actuarial science.

Define the characteristic function of S as

$$\phi_S(t) := E\left[e^{itS}\right] = \mathcal{P}_S\left(e^{it}\right)$$

and analogously, the characteristic version of the OGF for expected allocations,

$$\phi_S^{[1]}(t) := \sum_{k=0}^{\infty} e^{itk} E\left[X_1 \times 1_{\{S=k\}}\right] = \mathcal{P}_S^{[1]}\left(e^{it}\right).$$

In this section, we aim to recover the values of $E[X_1 \times 1_{\{S=k\}}]$ using the discrete Fourier transform (DFT). Set $\boldsymbol{f}_X = (f_X(0), f_X(1), \ldots, f_X(k_{max}-1))$ for a truncation point $k_{max} \in \mathbb{N}$. Here we assume that $f_X(k) = 0$ for $k \geq k_{max}$ such that there is no truncation error. The DFT of $\boldsymbol{f}_X$, noted $\widehat{\boldsymbol{f}}_X = (\widehat{f}_X(0), \widehat{f}_X(1), \ldots, \widehat{f}_X(k_{max}-1))$, is

$$\widehat{f}_X(k) = \sum_{j=0}^{k_{max}-1} f_X(j) e^{i2\pi jk/k_{max}}, \quad k = 0, \ldots, k_{max}-1. \tag{16}$$

The inverse DFT can recover the original sequence with

$$f_X(k) = \frac{1}{k_{max}} \sum_{j=0}^{k_{max}-1} \mathrm{Re}\left(\widehat{f}_X(j) e^{-i2\pi jk/k_{max}}\right), \quad k = 0, \ldots, k_{max}-1. \tag{17}$$

The authors of [Embrechts and Frei, 2009] explain how computing the pmf of a compound sum is more efficient with the FFT than using Panjer recursion or direct convolution. We now show how to apply the FFT algorithm to compute expected allocations. Let $\mu_{1:k} = E[X_1 \times 1_{\{S=k\}}]$ for $k = 0, \ldots, k_{max}-1$ and $\boldsymbol{\mu}_1 = (\mu_{1:0}, \ldots, \mu_{1:(k_{max}-1)})$ with the obvious case $\mu_{1:0} = 0$. Then, the discrete Fourier transform of $\boldsymbol{\mu}_1$, noted $\widehat{\boldsymbol{\mu}}_1 = (\widehat{\mu}_{1:0}, \ldots, \widehat{\mu}_{1:(k_{max}-1)})$, is

$$\widehat{\mu}_{1:j} = \mathcal{P}_S^{[1]}\left(e^{i2\pi j/k_{max}}\right), \quad j = 0, \ldots, k_{max}-1. \tag{18}$$

For notational convenience we write the vector $\{e^{i2\pi j/k_{max}}\}_{0 \leq j \leq k_{max-1}}$ as $\widehat{\boldsymbol{e}}_1$. Then, we have that $\widehat{\boldsymbol{\mu}}_1 = \mathcal{P}_S^{[1]}(\widehat{\boldsymbol{e}}_1)$. Computing the inverse DFT of (18) yields the values of $E[X_1 \times 1_{\{S=k\}}]$ for $k = 0, \ldots, k_{max}-1$. If k_{max} is a power of 2, algorithms like the FFT of [Cooley and Tukey, 1965] are especially efficient.

Note that computing the cumulative expected allocations is trickier since division by $(1-t)$ is undefined for $|t| = 1$. One, therefore, requires simplifications before applying the FFT to the OGF of expected cumulative allocations. In practice, one only obtains a slight numerical advantage from using the FFT method for expected cumulative allocations, which has algorithmic complexity $O(n \log n)$. If one computes expected allocations with the FFT method and takes the cumulative sum of the result, the algorithmic complexity becomes $O(n\log n + n)$, which is only slightly longer than using the cumulative expected allocations.

One consideration when using the FFT method is that one must select a truncation point large enough such that $f_S(k_{max}) = 0$. One could have a large value of k_{max} if S is a large portfolio or if individual risks have heavy tails. In the context of peer-to-peer insurance with a stop loss reinsurance contract with trigger ω, we have $f_S(x) = 0$ for $x > \omega$, and $E\left[X_1 \times 1_{\{S=k\}}\right] = E\left[X_1 \times 1_{\{S=\omega\}}\right]$ for all $k \geq \pi$; thus stop loss contracts sets an upper bound to the truncation point required.

If X_1 is a discrete rv, independent of S_{-1}, then the OGF for expected allocations is given by (13). If we have no closed-form solution for $\mathcal{P}'_{X_1}(t)$, then one can compute the DFT of $t\mathcal{P}'_{X_1}(t)$ by using the pmf of X_1 and the properties of OGFs. One can compute the pgf of X_1 as $\mathcal{P}_{X_1}(t) = \sum_{k=0}^{\infty} t^k f_{X_1}(k)$ and $t\mathcal{P}'_{X_1}(t) = \sum_{k=0}^{\infty} t^k k f_{X_1}(k)$. It follows that one can compute the DFT of $t\mathcal{P}'_{X_1}(t)$ as the DFT of the vector $\{kf_{X_1}(k)\}_{k\in\mathbb{N}_0}$. We can compute the DFT of $t\mathcal{P}'_{X_1}(t)/(1-t)$ as the DFT of the partial sum of the vector $\{kf_{X_1}(k)\}_{k\in\mathbb{N}_0}$.

One can use the methods described in this paper with continuous rvs by discretizing the continuous cdfs. A simple approach is the rounding method, computing the pmf with

$$\widetilde{f}_X(kh) =: \begin{cases} F_X\left(kh + \frac{h}{2}\right), & k = 0 \\ F_X\left(kh + \frac{h}{2}\right) - F\left(kh - \frac{h}{2}\right), & k \in \mathbb{N}_1 \end{cases}$$

for a step size $h \in \mathbb{R}^+$. For other methods (including the lower and upper method) and their application to the FFT method of aggregating, see [Embrechts and Frei, 2009]. The remark also applies when X_1 is a compound rv with continuous severity.

3 Applications of the FFT method

3.1 A large portfolio of independent compound Poisson rvs

In a first application, we consider a portfolio or pool of 10 000 contracts. Each risk X_i is independent and follows a compound Poisson distribution with mean λ_i, with severity rv $B_i \sim NBinom(r_i, q_i)$, for $i = 1, \ldots, 10\ 000$. We set each risk to have different parameters. For illustration purposes we simulate the parameters (λ_i, r_i, q_i) for $i = 1, \ldots, 10\ 000$ according to

$$\begin{cases} \lambda_i & \sim Exp(10) \\ r_i & \sim Unif(\{1,2,3,4,5,6\}) \\ q_i & \sim Unif([0.4, 0.5]) \end{cases}$$

such that on average, $\lambda_i = 0.1$, $r_i = 3.5$ and $q_i = 0.45$. We present the simulated parameters along with the expected values for the first 8 contracts in Table 1.

We present the code in Appendix A, using R version 4.0.4. In Figure 1 (left), we present a plot of $\sum_{i=1}^{n} E[X_i \mid S = k]$, for $k \in \{0, 1, \ldots, k_{max} - 1\}$. The theoretical value for this expectation is k, so one should obtain a straight line, which does occur for $3650 \leq k \leq 5250$. For values of k outside this range, we obtain erratic results, which is due to underflow issues. Indeed, looking at Figure 1, one

i	1	2	3	4	5	6	7	8
λ_i	0.161152	0.031859	0.027368	0.238748	0.115137	0.470203	0.146247	0.011747
q_i	0.489756	0.423367	0.455898	0.451500	0.486834	0.440405	0.440082	0.481335
r_i	2	6	1	4	6	5	3	1
$E[X_i]$	0.335788	0.260354	0.032662	1.160162	0.728190	2.987289	0.558214	0.012658

Table 1: First 8 set of parameters

notices that the density function of S is non-zero for $3800 \leq k \leq 5000$, hence both $E[X_i \times 1_{\{S=k\}}]$ and $\Pr(S = k)$ are smaller than the precision when using double-precision floating-point format with IEEE 754. Hence, the ratio of two underflow values cause unreliable results and one should discard the corresponding allocations. However, the validation curve is linear over the range where S has a non-zero density, so the conditional means of interest are available. The underflow issue occurs with most applications of FFT to identify expected allocations. One can define a range of valid expected allocations by discarding the values of k such that $|k - \sum_{i=1}^{n} E[X_i \mid S = k]|$ is larger than some tolerance (we use 10^{-8}).

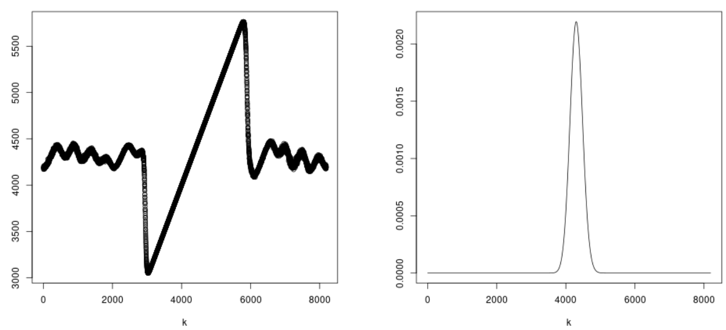

Figure 1: Left: total conditional means. Right: density function of S.

The listing in Appendix A computes $1600 \times 10\ 000$ conditional means at once and takes approximately 16 seconds on a personal computer (with a Intel®Core™i5-7600K CPU @ 3.80GHz CPU).

In Figure 2 we present the plots for the distribution of conditional means, that is, $\Pr(E[X_i|S] = k)$ for $i = 1, \ldots, 8$, with $E[X_j]$ in vertical lines. As shown in [Denuit and Robert, 2021b], the distribution of conditional means is asymptotically normal and we observe this in Figure 2.

Note that we optimize our algorithm for speed, so we store intermediate values of f_B and $\widehat{\boldsymbol{f}}_X$ in vectors. The code required about 10Gb of RAM to store the intermediate values, which may become high for home computers. In this situation, we recommend storing intermediate values in a separate file or not storing the values at all (which will require computing $\widehat{\boldsymbol{f}}_X$ several times, but take much less RAM).

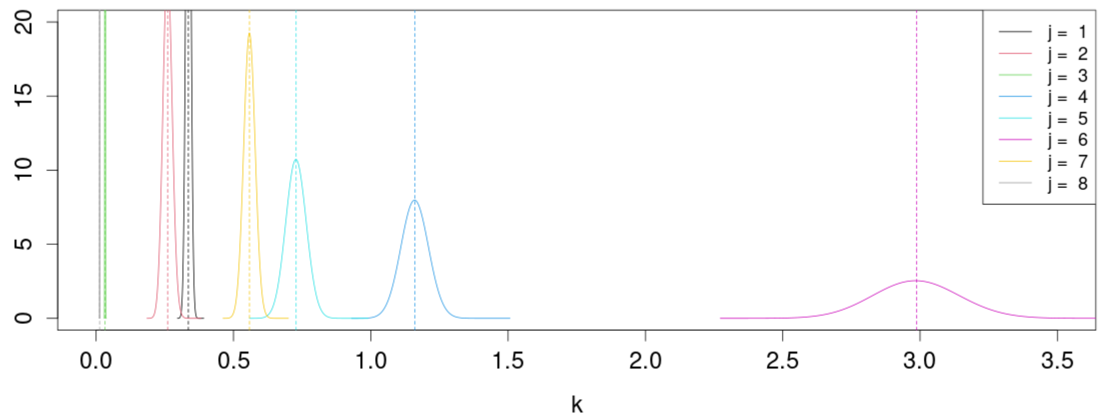

Figure 2: Density function of conditional means for the first eight contracts.

3.2 A portfolio of heavy tailed rvs

Next, we consider the computation of expected allocations for a portfolio of heavy-tailed distributions. In particular, we select rvs for which the variance does not exist; hence the central limit theorem results of [Denuit and Robert, 2021b] do not hold. We consider a portfolio of size $n \in \{3, 100, 1000\}$ and compare the behaviour of the first three contracts. Our goal is to show empirically that the distribution of conditional means converges to the individual mean. We set $X_i, i \in \{1, \ldots, n\}$, to follow an arithmetized Pareto distribution defined using the moment matching method, see, e.g., Appendix E.2 of [Klugman et al., 2018]. Further, we select parameters $\alpha_i \in [1.3, 1.9]$, for $i \in \{1, \ldots, n\}$, such that the variance of individual risks does not exist. For the first three risks, we select $(\alpha_1, \alpha_2, \alpha_3) = (1.3, 1.6, 1.9)$ and $(\lambda_1, \lambda_2, \lambda_3) = (10(1.3-1), 10(1.6-1), 10(1.9-1))$ such that $E[X_j] \approx 10$ for $i \in \{1, 2, 3\}$. We write the approximate symbol since the mean may not be preserved exactly due to truncation, since Pareto rvs are heavy tailed. For the remaining risks $X_i, i \in \{4, \ldots, 1000\}$, we simulate the parameters according to

$$\begin{cases} \alpha_i & \sim Unif([1.3, 1.9]) \\ \lambda_i & \sim Unif([5, 15]) \end{cases}$$

such that $50/9 \leq E[X_j] \leq 50$ for $i \in \{4, \ldots, 1000\}$, and the variance does not exist for any risk in the portfolio.

In Figure 3 we present the cdf of the conditional means for risks X_1, X_2 and X_3. The dashed, dotted, and dash-dotted lines present the cdf of conditional means for $n = 3, 100$ and 1000 respectively. Due to the heavy tailed risks, one must select a large truncation point k_{max} to avoid aliasing (see, e.g., [Grubel and Hermesmeier, 1999] and [Embrechts and Frei, 2009] for discussions on aliasing with FFT methods for aggregation). Hence, we compute $1\,000 \times 2^{20}$ values, which takes approximately 9 minutes on a personal laptop. To facilitate comparisons, we present the cdf of X_i in red and the expected value of X_i in green (vertical line), $i \in \{1, 2, 3\}$. Each cdf cross once, hence following the Karlin-Novikoff criteria and since they share the same mean, then the conditional means are ordered under the convex order, which is expected, see e.g., [Denuit and Dhaene, 2012]. One may observe that the cdfs of the conditional means approach the cdf of a degenerate rv at the mean. The conditional mean of X_3 approaches the degenerate rv at its mean faster since its tail is lighter than X_1 or X_2. Indeed, one observes that the cdf of $E[X_3 \times 1_{\{X_1+\cdots+X_{1000}=k\}}]$ is almost vertical, while the cdf of $E[X_1 \times 1_{\{X_1+\cdots+X_{1000}=k\}}]$ is not.

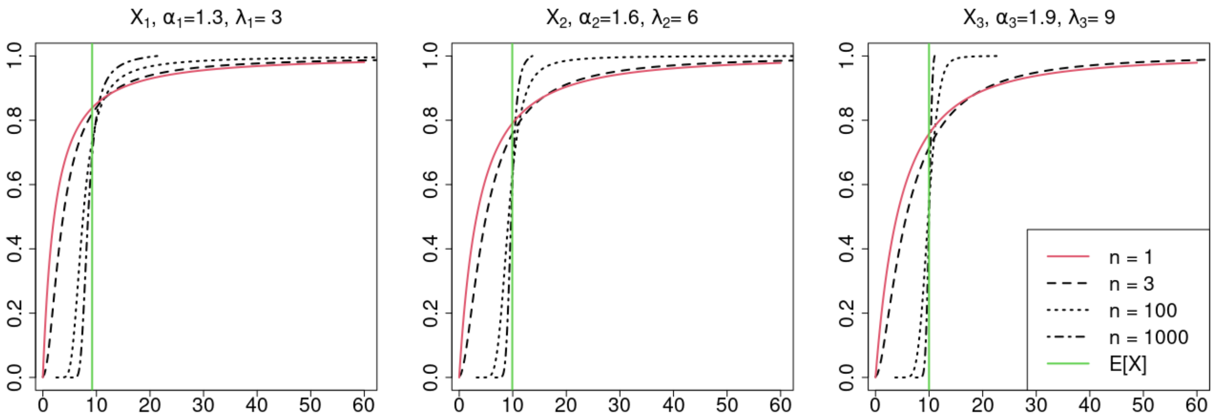

Figure 3: Cumulative distribution function of conditional means for $n = 1, 3, 100, 1000$.

In this application, we show empirically that the conditional mean $E[X_1|S]$ converges in probability to the expected value $E[X_1]$ as the size of the portfolio increases. However, future research remains to show that this observation is true in general, that is, providing a law of large numbers result for the conditional mean, generalizing the results of [Denuit and Robert, 2020] and [Denuit and Robert, 2021b].

3.3 Application: small portfolio of heterogeneous claims

Let $\boldsymbol{I} = (I_1, \ldots, I_n)$ be a vector of independent Bernoulli rvs with marginal probabilities $q_i \in (0, 1)$, for $i \in \{1, \ldots, n\}$. Further define the rv $X_i = b_i \times I_i$, with $b_i \in \mathbb{N}_1$, for $i \in \{1, \ldots, n\}$. This model is sometimes called the individual risk model (with a fixed payment amount) and has applications for instance in life insurance, where death benefits are usually known in advance, or for insurance-linked securities in situations where investors recover their initial investment unless a trigger event occurs before the maturity date. The interested reader may refer to [Klugman et al., 2013] for detailed examples of the individual risk model. The multivariate pgf of $\boldsymbol{X} = (X_1, \ldots, X_n)$ is

$$\mathcal{P}_{\boldsymbol{X}}(t_1, \ldots, t_n) = \prod_{i=1}^{n} (1 - q_i + q_i t_i^{b_i}), \tag{19}$$

while the OGF of the sequence of expected allocations for risk X_1 is

$$\mathcal{P}_S^{[1]}(t) = q_1 t^{b_1} \prod_{i=2}^{n} (1 - q_i + q_i t^{b_i}).$$

To compute exact values of the pmf and expected allocations using the FFT approach, one must select $k_{max} \geq 1 + \sum_{i=1}^{n} b_i$ (or alternatively, select the smallest m such that $2^m \geq 1 + \sum_{i=1}^{n} b_i$).

Note that in this model, difficulties arise from the fact that the rvs have a discrete and heterogeneous support. As we will see from the numerical example from this setup, the FFT-based method proposed in the current paper handles these cases with ease, while theoretical results rely on number theory. We first define some notation. Let $\mathcal{B}$ represent the σ-algebra generated from the random vector with pgf in (19), that is, $\mathcal{B} = \{(x_1, \ldots, x_n) : x_i \in \{0, b_i\}, 1 \leq i \leq n\}$. Further, let $\mathcal{B}_k$ represent the subset of events that sum to k, for $k \in \mathbb{N}_0$. We call this subset of events the partition of k over $\mathcal{B}$. For a specific partition of size ℓ with elements $\{x_1, \ldots, x_\ell\}$, we say that x_i, $i \in \{1, \ldots, \ell\}$ is a part of the partition. We use the notation $|\cdot|$ to represent cardinality. We say that k is a possible outcome of the total losses if $|\mathcal{B}_k| > 0$. If $|\mathcal{B}_k| = 0$, we have $\Pr(S = k) = 0$, and correspondingly $E[X_i|S = k] = 0$ for all $i \in \{1, \ldots, n\}$. Further, if $|\mathcal{B}_k| = 1$, we have that

$$E[X_i|S = k] = \begin{cases} b_i, & \text{if } b_i \text{ is a part of } \mathcal{B}_k \\ 0, & \text{otherwise} \end{cases}.$$

Notice that for $|\mathcal{B}_k| = 1$, no diversification occurs for participants with losses. Such situations may occur when the number of contracts is small and the coverage amounts are heterogeneous. If there exists an outcome in $\mathcal{B}_k$ such that b_ℓ is not a part of the outcome, then we have $E[X_\ell|S = k] < b_\ell$. Also, if $b_i = b_j$ for some $i \neq j$ and b_i or b_j is a part of an element of $\mathcal{B}_k$, then we have $E[X_i|S = k] < b_i$. Fortunately, the OGF method provides a numerical solution to compute the expected allocations without further notions of number theory. See also Example 4.1 of [Denuit et al., 2021b] for a situation where no diversification occurs for some participants due to partitions of odd numbers.

We consider a portfolio of $n = 6$ risks. We present the parameters for this example in Table 2, and the code to replicate this study is in the Appendix.

i	1	2	3	4	5	6
b_i	1	3	10	4	5	10
q_i	0.8	0.2	0.3	0.05	0.15	0.25

Table 2: Marginal parameters for small portfolio of heterogeneous claims.

In Figure 4, we present the conditional means along with the pmf and cdf of conditional means for risks X_1, X_2 and X_3. We describe each panel in the following:

- The left panel presents the conditional means. Note that for the claim severity values in Table 2, we have $|\mathcal{B}_2| = |\mathcal{B}_{31}| = 0$, hence the events $S = 2$ and $S = 31$ are impossible and we have $\Pr(S = 2) = \Pr(S = 31) = 0$. When computed using the pgf in (33) and the FFT algorithm, we have $\Pr(S = 2) = \Pr(S = 31) \approx 10^{-16}$ since this is the underflow error using double precision with IEEE 754. Hence, the conditional means should be 0 for $k = 2$ and $k = 31$, division of two underflowed values generates erratic results. These values should be rejected from the analysis, but we show them in red here as a warning of numerical problems with the FFT method if one is not wary of underflow versus true zeroes when using the FFT method. As in other applications, one should observe the total conditional means (row 4 of Figure 4) and retain the values that form a step function with steps of 1. Conditional means which deviate from its expected total should be discarded, either due to underflow or to division by zero. However, the events which cause numerical issues have zero or negligible probability (under 10^{-16}), hence the expectations of interest do not suffer from underflow.

  Also of interest is the shape of conditional means as a function of k. For $i = 1$, we have unpredictable expected allocations since the outcome 1 is often a part of $\mathcal{B}_k, k \in \mathbb{N}_1$, and the claim probability is higher than the remaining risks. For $i = 2$, we have predictable expected allocations since 3 is a part of $\mathcal{B}_k$ for cyclical values of k, and 3 does not divide the other values of $b_i, i \in \{1, 3, 4, 5, 6\}$. Finally, we have $b_3 = 10$ and $b_6 = 10$. Hence, the conditional allocations are often shared between risks X_3 and X_6, though not perfectly due to their different claim probability. In row 3 of Figure 4, we also have a mass around 7.5 since the outcomes $X_1 = 1$, $X_4 = 4$ and $X_5 = 5$ yields $S = 10$. Once again the allocation is above 7.5 due to the higher probability of occurrence compared to the probability of events $X_1 = 1$, $X_4 = 4$ and $X_5 = 5$ occurring at once.

- The middle panel presents the pmf of $E[X_i|S]$, for $i \in \{1, 2, 3\}$. The support of this random variable is the set of values $\{k \in \mathbb{N} | E[X_i|S = k] > 0\}$. Notice that the support for the expected allocations is sparse, for small portfolios with heterogeneous values of $b_i, i \in \{1, \ldots, n\}$.
- The right panel presents the cdf of expected allocations, which simplifies the interpretation of the middle panel since the probability masses may appear close together.

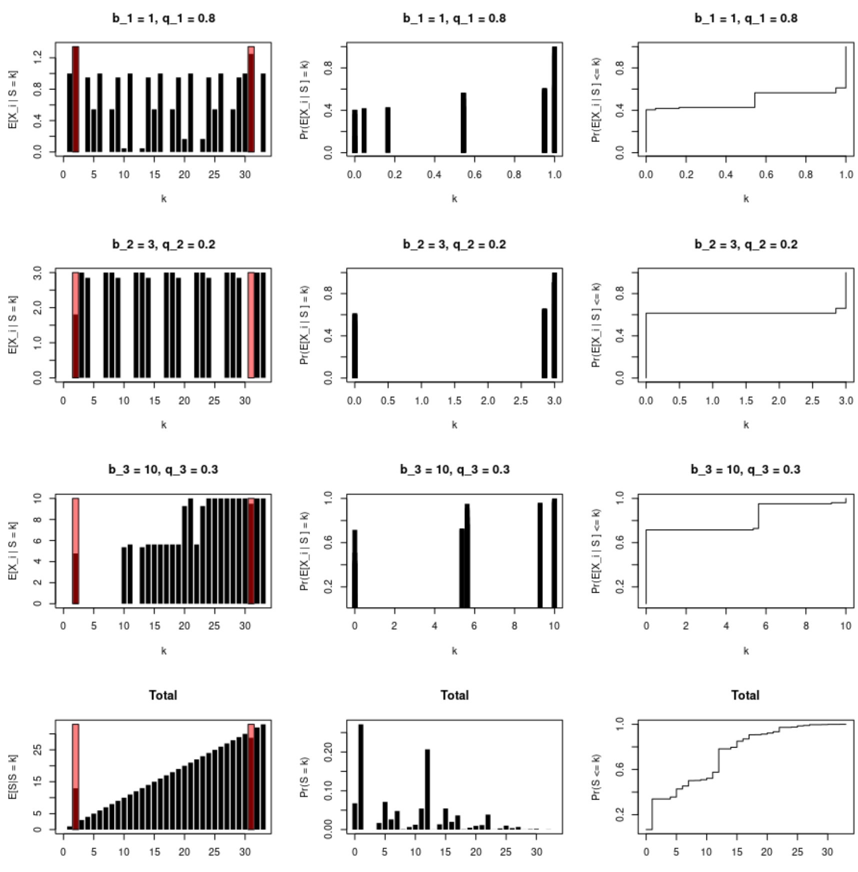

Figure 4: Left: conditional means. Middle: pmf of conditional means. Right: cdf of conditional means.

Note that as more participants enter the pool, more risks may diversify, that is, $\mathcal{B}_k$ has a higher cardinality for all $k \in \mathbb{N}_1$. It follows that the risks diversify and the pdf of expected allocations is less sparse. In Figure 5, we replicate the above study but add 69 participants where we sample the parameters according to $q_i \sim Unif(0,1)$ and $b_i \sim Unif(\{1,2,\ldots,10\})$. We present the results for risk 3 (with $b_3 = 10$) and the total pool in Figure 5. We observe once again that there are numerical issues for large values of k in the left panel. However, the middle panel is much less sparse than in Figure 4.

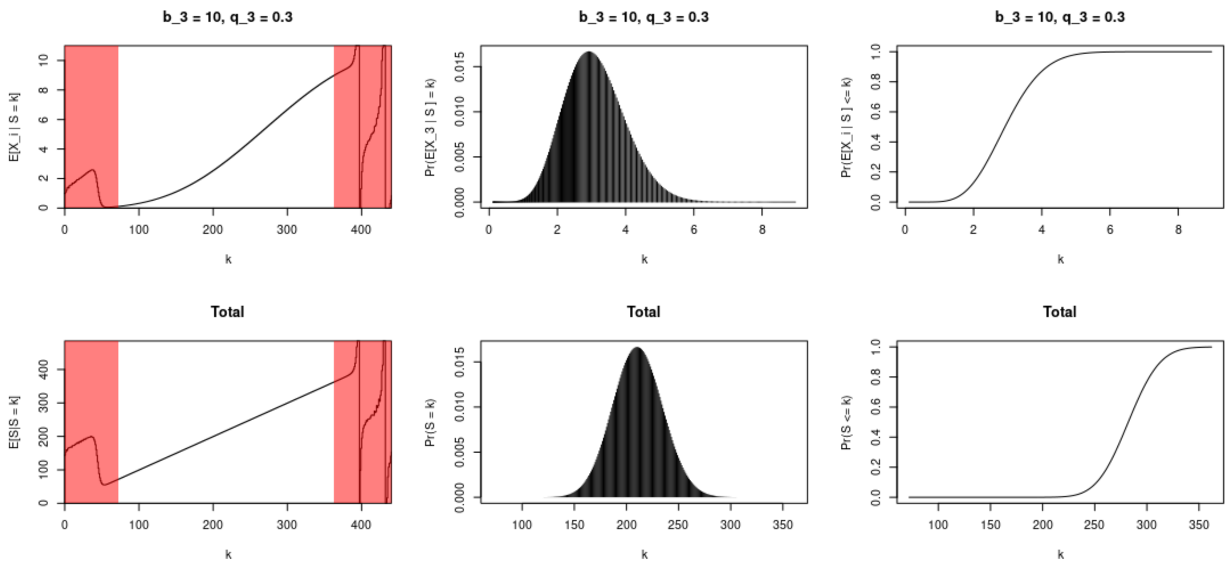

Figure 5: Pool of 75 participants. Left: conditional means. Middle: pmf of conditional means. Right: cdf of conditional means.

4 Implications for independent (a, b, 0) distributions

We consider the $(a,b,0)$ family of distributions in [Klugman et al., 2018]. These distributions satisfy the recurrence relation $f_M(k) = (a + b/k)f_M(k-1)$ for $k \in \mathbb{N}_1$. Members of this family are the Poisson (with $a = 0$ and $b = \lambda$), binomial (with $a = -q/(1-q)$ and $b = (n+1)q/(1-q)$) and negative binomial with pmf given by

$$f_M(k) = \binom{r+k-1}{k} q^r (1-q)^k, \quad k \in \mathbb{N}_0$$

(with $a = 1 - q$ and $b = (r-1)(1-q)$). Note that each distribution have different starting values within the recurrence relation, but the starting values aren't required within the current paper.

When risks are independent (compound) $(a,b,0)$ distributions, we will first develop explicit results for the expected allocations, these results will be sufficient to compute expected allocations for small portfolios. Then, we will present an FFT-based algorithm for large portfolios of compound Poisson distributions.

The following lemma concerning the pgf of $(a,b,0)$ distributions will be useful, this lemma appears notably in Section 4.5 of [Dickson, 2017].

Lemma 4.1. *For a rv M with pmf in the $(a,b,0)$ class, we have $\mathcal{P}'_M(t) = (a+b)/(1-at)\mathcal{P}_M(t)$.*

4.1 Allocations when one risk follows an (a, b, 0) distribution

The following theorem presents an efficient formula to compute expected allocations.

Theorem 4.2. *Let X_1 be a rv with pmf f_{X_1} in the $(a, b, 0)$ family of distributions, independent of S_{-1}. For $|a| < 1$ and $k \in \mathbb{N}_1$, we have*

$$[t^k]\mathcal{P}_S^{[1]}(t) = E\left[X_1 \times 1_{\{S=k\}}\right] = (a+b)\sum_{j=0}^{k-1} a^j f_S(k-1-j) \tag{20}$$

and

$$[t^k]\left\{\frac{\mathcal{P}_S^{[1]}(t)}{1-t}\right\} = E\left[X_1 \times 1_{\{S\le k\}}\right] = (a+b)\sum_{j=0}^{k-1} a^j F_S(k-1-j) \tag{21a}$$

$$= (a+b)\sum_{j=0}^{k-1} \frac{1-a^{j+1}}{1-a} f_S(k-1-j). \tag{21b}$$

Proof. Applying Lemma 4.1, the OGF for expected allocations is

$$\mathcal{P}_S^{[1]}(t) = t\mathcal{P}'_{X_1}(t)\mathcal{P}_{X_2,\ldots,X_n}(t) = t\frac{a+b}{1-at}\mathcal{P}_{X_1}(t)\mathcal{P}_{X_2,\ldots,X_n}(t) = t\frac{a+b}{1-at}\mathcal{P}_S(t). \tag{22}$$

Then, (20) follows from Property 4 of OGFs. The relation in (21a) follows from another application of Property 4 to (20). Alternatively, the OGF for expected cumulative allocations is

$$\frac{\mathcal{P}_S^{[1]}(t)}{1-t} = (a+b)\frac{t}{(1-at)(1-t)}\mathcal{P}_S(t) = \frac{a+b}{a-1}\mathcal{P}_S(t)\left(\frac{1}{1-at} - \frac{1}{1-t}\right) = \frac{a+b}{a-1}\mathcal{P}_S(t)\sum_{k=0}^{\infty} t^k\left(a^k - 1\right) \tag{23}$$

for $|t| < 1$. Then, (21b) also follows from the convolution property of OGFs. $\square$

Notice that (20) and (21b) require the same number of computations, so it isn't more complex to compute cumulative allocations than individual valued allocations. We also have the relationship

$$(a-1)E\left[X_1 \times 1_{\{S\le k\}}\right] = aE\left[X_1 \times 1_{\{S=k\}}\right] - (a+b)F_S(k-1), \quad |a| < 1, \quad k \in \mathbb{N}_0. \tag{24}$$

Note the resemblance between (24) and the Stein-Chen identity (see [Chen, 1975, Weiß and Aleksandrov, 2022]). As both the expected allocation and the expected cumulative allocation are required to compute (5), the relationship in (24) is quite useful, since one does not need to compute every expected allocation in (14).

We list the implications of Theorem 4.2 in Table 3. As a particular case, if $X_1, \ldots, X_n$ are independent Poisson rvs with respective rates $\lambda_1, \ldots, \lambda_n$, we have $S \sim Pois(\lambda_S)$, with $\lambda_S = \lambda_1 + \cdots + \lambda_n$. Then we recover the result presented in Section 10.3 of [Marceau, 2013, page 413],

$$[t^k]\mathcal{P}_S^{[1]}(t) = E\left[X_1 \times 1_{\{S=k\}}\right] = \lambda_1\frac{\lambda_S^{k-1}e^{-\lambda_S}}{(k-1)!} = \frac{\lambda_1}{\lambda_S}k\Pr(S=k), \quad k \in \mathbb{N}_0.$$

Thus, we have $E[X_1|S = k] = \lambda_1/\lambda_S k$ which is a linear function of k, hence the conditional mean here is called a proportional or linear allocation rule. For binomial distributions, one requires the success probability to satisfy $q < 1/2$ such that $|a| < 1$. Alternately, if $q > 1/2$, one could express the problem in terms of failure probability $1 - q$ and then apply Theorem 4.2.

	Poisson	Negative binomial	Binomial
a	0	$1-q$	$-q/(1-q)$, for $0<q<1/2$
b	λ	$(r-1)(1-q)$	$(n+1)1/(1-q)$
$E[X_1 \times 1_{\{S=k\}}]$	$\lambda f_S(k-1)$	$r\sum_{j=1}^{k}(1-q)^j f_S(k-j)$	$-n\sum_{j=1}^{k}\left(-\frac{q}{1-q}\right)^j f_S(k-j)$
$E[X_1 \times 1_{\{S\leq k\}}]$ (v1)	$\lambda F_S(k-1)$	$r\sum_{j=1}^{k}(1-q)^j F_S(k-j)$	$-n\sum_{j=1}^{k}\left(-\frac{q}{1-q}\right)^j F_S(k-j)$
$E[X_1 \times 1_{\{S\leq k\}}]$ (v2)	–	$r\frac{1-q}{q}\sum_{j=1}^{k}\left(1-(1-q)^j\right) f_S(k-j)$	$-n\sum_{j=1}^{k}\frac{1-\left(-\frac{q}{1-q}\right)^{j+1}}{1+\frac{q}{1-q}} f_S(k-j)$

Table 3: Implications of Theorem 4.2 for all distributions.

4.2 Allocations when one risk follows a compound (a, b, 0) distribution

Let M be a frequency rv with support on $\mathbb{N}_0$. Let $\{B_1, B_2, \dots\}$ form a sequence of independent and non-negative severity rvs, also independent of M. Within the context of the current paper, we assume that the severity rvs have support on $\mathbb{N}_0$. In this section, we consider cases where the rv X is defined as a random sum, that is,

$$X = \begin{cases} 0, & M = 0 \\ \sum_{j=1}^{M} B_j, & M > 0 \end{cases}.$$

The pmf of X is

$$\Pr(X = kh) = \begin{cases} \Pr(M = 0), & kh = 0 \\ \sum_{j=1}^{\infty} \Pr(M = j) \Pr(B_1 + \cdots + B_j = kh), & k \in \mathbb{N}_0 \end{cases}.$$

Evaluation of $\Pr(B_1 + \cdots + B_j = kh)$ is analytically and computationally expensive since direct computation is the result of $j-1$ convolutions. Fortunately, [Panjer, 1981] and others have developed efficient recursive relationships to compute the pmf of X when M is a rv in the $(a, b, 0)$ class of distributions called the Panjer recursion. We are now interested in the OGF for expected allocations for compound distributions where M is a rv in the $(a, b, 0)$ class of distributions such that we may have an efficient algorithm for expected allocations.

Proposition 4.3. *Let X_1 be a compound rv with frequency rv M_1 in the $(a, b, 0)$ family of distributions with $|a| < 1$ and discrete severity rv B_1, with X_1 independent of $(X_2, \dots, X_n)$. The expected OGF of expected allocations is*

$$\mathcal{P}_S^{[1]}(t) = t\mathcal{P}'_{B_1}(t)\mathcal{P}'_{M_1}\left(\mathcal{P}_{B_1}(t)\right)\mathcal{P}_{X_2,\dots,X_n}(t). \tag{25}$$

Further, if $|a\mathcal{P}_{B_1}(t)| < 1$ for all $|t| < 1$, then

$$\mathcal{P}_S^{[1]}(t) = t\mathcal{P}'_{B_1}(t)\frac{a+b}{1-a\mathcal{P}_{B_1}(t)}\mathcal{P}_S(t). \tag{26}$$

Proof. The pgf of the compound rv X_1 is $\mathcal{P}_{X_1}(t) = \mathcal{P}_{M_1}\left(\mathcal{P}_{B_1}(t)\right)$, then (25) follows directly from (12). The relation in (26) follows from the chain rule and Lemma 4.1. □

Example 4.4 (Independent compound Poisson distributions with discrete severity)**.** *Let X_1 be a rv in the class of compound Poisson distributions, whose severity distribution is discrete with support $\mathbb{N}_0$. We have*

$$\mathcal{P}_S^{[1]}(t) = \lambda_1 t\mathcal{P}'_{B_1}(t)\mathcal{P}_{M_1}(\mathcal{P}_{B_1}(t))\mathcal{P}_{X_2,\dots,X_n}(t) = \lambda_1 t\mathcal{P}'_{B_1}(t)\mathcal{P}_S(t). \tag{27}$$

It follows that

$$[t^k]\mathcal{P}_S^{[1]}(t) = E[X_1 \times 1_{\{S=k\}}] = \lambda_1 \sum_{l=1}^{k} l f_{B_1}(l) f_S(k-l), \quad k \in \mathbb{N}_1$$

and

$$[t^k]\left\{\frac{\mathcal{P}_S^{[1]}(t)}{1-t}\right\} = E[X_1 \times 1_{\{S\leq k\}}] = \lambda_1 \sum_{l=1}^{k} E\left[B_1 \times 1_{\{B_1 \leq l\}}\right] f_S(k-l), \quad k \in \mathbb{N}_1$$
$$= \lambda_1 \sum_{l=1}^{k} l f_{B_1}(l) F_S(k-l), \quad k \in \mathbb{N}_1.$$

△

Remark 4.5. The results of Theorem 4.3 are analogous to the results in Section 4 of [Denuit and Robert, 2020] for the discrete case. One can recover continuous versions of the results from [Denuit and Robert, 2020] using the continuous version of the OGF for expected cumulative allocations (see Section 6). △

4.3 An algorithm for a portfolio of independent compound Poisson distributions

Consider a portfolio of n participants, where X_i is a compound Poisson rv with frequency parameter λ_i and discrete severity rv B_i for $i = 1, \ldots, n$. We have $\mathcal{P}_S(t) = \prod_{i=1}^{n} \mathcal{P}_{M_i}\left(P_{B_i}(t)\right)$. For Poisson distributions, the OGF of expected allocations for the ith risk, $i \in \{1, \ldots, n\}$, is $\lambda_i t \mathcal{P}'_{B_i}(t)\mathcal{P}_S(t)$.

We use the fast Fourier transform to compute the expected allocations. Using FFT to compute the values of f_S is the most computationally intensive step. Fortunately, since the term $\mathcal{P}_S(t)$ is present for the pgf of S, the OGF for expected allocations and expected cumulative allocations, one must only compute the DFT of f_S once, then one can obtain the values of (i) the pmf of S, (ii) the expected allocation $E[X_i \times 1_{\{S=k\}}]$ and (iii) $E[X_i \times 1_{\{S\leq k\}}]$ with fewer computations. In Algorithm 1, we present a method to compute the expected allocations in an efficient manner using the FFT. One can change line 6 by the cumulative sum of the vector to compute cumulative expected allocations.

4.3.1 Application: small portfolio of independent compound Poisson distributed rvs

We replicate case 1 of the application in Section 6.1 of [Denuit, 2019]. Consider four participants in a pool, and each participant contributes a compound Poisson loss, with a discrete severity whose support is $\{1, 2, 3, 4\}$. We present the parameters for each participant in Table 4. We provide the R code in Appendix A, the numerical values that follow come from R version 4.0.4. Other than the setup and validation code, the actual computation of conditional means takes fewer than 15 lines (even if the number of participants grows). We recover the values in [Denuit, 2019]. We investigate numerical issues from the small portfolio for the remainder of the application. One notices numerical issues when working with the FFT algorithm and events with zero probability. Indeed, the validation curve is linear between $k = 0$ and $k = 37$. However, one has $\sum_{i=1}^{n} E[X_i \mid S =$

Algorithm 1: Conditional expected allocations for compound Poisson distributions.

Input: Parameters $\lambda_i, \boldsymbol{f}_{B_i}$ for $i = 1, \ldots, n$.

Output: Expected allocations $E[X_i|S = k]$ for $k = 0, \ldots, k_{max} - 1$ and $i = 1, \ldots, n$

for $i = 1, \ldots, n$ **do**

  Compute $\widehat{\boldsymbol{f}}_{X_i} = \mathcal{P}_{X_i}(\widehat{\boldsymbol{e}}_1)$ or with (16);

Compute the DFT of S as the element-wise product $\widehat{\boldsymbol{f}}_S = \prod_{i=1}^{n} \widehat{\boldsymbol{f}}_{X_i}$;

Compute $\boldsymbol{f}_S$ by taking the inverse DFT of $\widehat{\boldsymbol{f}}_S$;

for $i = 1, \ldots, n$ **do**

  Compute the DFT $\widehat{\boldsymbol{\phi}}_{B_i}$ of the vector $\{(k+1)f_{B_i}(k+1)\}_{0 \leq k \leq k_{max}-1}$;

  Compute element-wise $\widehat{\boldsymbol{\mu}}_i = \lambda_i \widehat{\boldsymbol{e}}_1 \times \widehat{\boldsymbol{\phi}}_{B_i} \times \widehat{\boldsymbol{f}}_S$;

  Compute $\boldsymbol{\mu}_i$ as the inverse DFT of $\widehat{\boldsymbol{\mu}}_i$;

  Compute $\{E[X_i|S = k]\}_{0 \leq k \leq k_{max}-1}$ by the element-wise division $\boldsymbol{\mu}_i / \boldsymbol{f}_S$;

Return $\{E[X_i|S = k]\}_{0 \leq k \leq k_{max}-1}$ for $i = 1, \ldots, n$.

i	λ_i	$f_{C_i}(1)$	$f_{C_i}(2)$	$f_{C_i}(3)$	$f_{C_i}(4)$
1	0.08	0.1	0.2	0.4	0.3
2	0.08	0.15	0.25	0.3	0.3
3	0.1	0.1	0.2	0.3	0.4
4	0.1	0.15	0.25	0.3	0.3

Table 4: Parameters for small portfolio.

38] = 38.05, which is slightly higher than 38. The FFT method of computing expected allocations completely breaks down for larger numbers, for example, we have $\sum_{i=1}^{n} E[X_i \mid S = 43] = 116$ and $\sum_{i=1}^{n} E[X_i \mid S = 63] = -146$. This occurs due to underflow and rounding errors, as described in the large portfolio application.

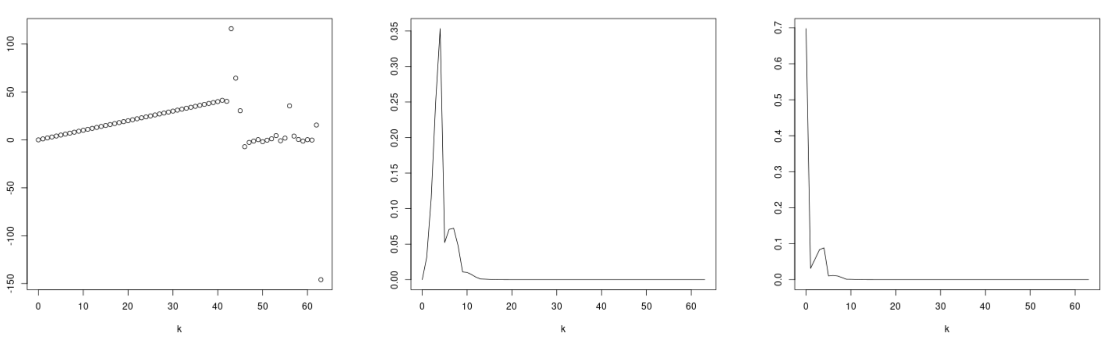

Figure 6: Left: $\sum_{i=1}^{n} E[X_i|S = k]$. Middle: $\sum_{i=1}^{n} E[X_i \times 1_{\{S=k\}}]$. Right: $f_S(k)$.

5 Dependent random variables

One may also use the methods described in this paper to compute expected allocations for dependent rvs. One obtains convenient results when the multivariate pgf is simple to differentiate, which is sometimes the case for mixture models (which include common shock models). The results from this section supplement the literature on capital allocation for mixture models as studied in Section 3 of [Cossette et al., 2018] or Section 4 of [Denuit and Robert, 2021a].

5.1 Common shock example

As a first example, we present a common shock model. Multivariate Poisson distributions based on common shocks are studied notably in [Teicher, 1954] and [Mahamunulu, 1967]. The interested reader may also consult [Lindskog and McNeil, 2003] for actuarial applications of common shock Poisson models.

Example 5.1 (Hierarchical common Poisson shocks). *Let $Y_A \sim Pois(\lambda_A)$ for $A \in \{\{1,2\}^3 \cup \{1,2\}^2 \cup \{0,1,2\}\}$ be independent rvs. We construct dependent rvs through the common shock framework $X_{ijk} = Y_{ijk} + Y_{ij} + Y_i + Y_0$ for $(i,j,k) \in \{1,2\}^3$. This is a special case of the multivariate Poisson distribution from [Mahamunulu, 1967], and we illustrate the dependence structure in Figure 7. Let $S = \sum_{(i,j,k)\in\{1,2\}^3} X_{ijk}$. Then one may verify that S follows a compound Poisson distribution,*

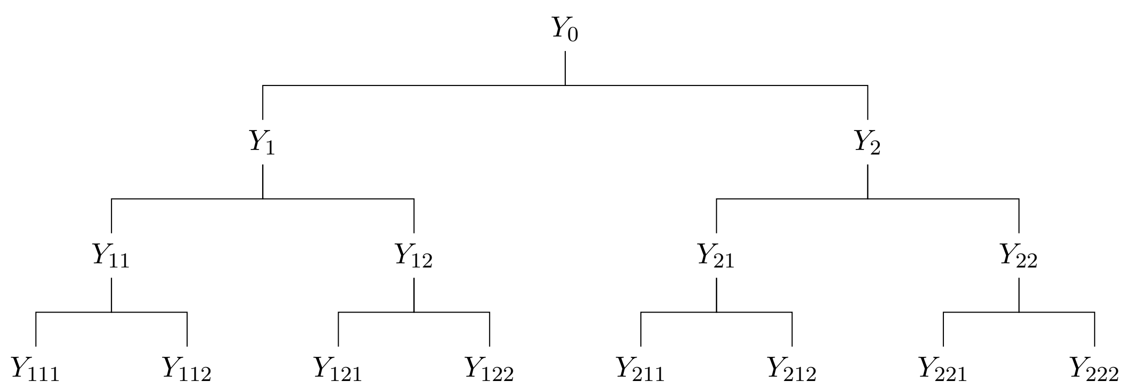

Figure 7: Hierarchical Poisson common shock structure.

so one may use Panjer recursion or FFT to compute the values of the probability mass function of S. Further, the OGF for the expected allocations of risk X_{ijk}, for $(i,j,k) \in \{1,2\}^3$, is

$$\mathcal{P}_S^{[ijk]}(t) = \left(\lambda_{ijk}t + \lambda_{ij}t^2 + \lambda_i t^4 + \lambda_0 t^8\right)\mathcal{P}_S(t).$$

For $(i,j,k) \in \{1,2\}^3$, we deduce that

$$E\left[X_{ijk} \times 1_{\{S=k\}}\right] = \begin{cases} 0, & k = 0 \\ \lambda_{ijk} f_S(k-1), & k = 1 \\ \lambda_{ijk} f_S(k-1) + \lambda_{ij} f_S(k-2), & k = 2,3 \\ \lambda_{ijk} f_S(k-1) + \lambda_{ij} f_S(k-2) + \lambda_i f_S(k-4), & k = 4,\ldots,7 \\ \lambda_{ijk} f_S(k-1) + \lambda_{ij} f_S(k-2) + \lambda_i f_S(k-4) + \lambda_0 f_S(k-8), & k = 8,9,\ldots \end{cases}$$

and

$$E\left[X_{ijk}\times 1_{\{S\leq k\}}\right]=\begin{cases}0, & k=0\\ \lambda_{ijk}F_S(k-1), & k=1\\ \lambda_{ijk}F_S(k-1)+\lambda_{ij}F_S(k-2), & k=2,3\\ \lambda_{ijk}F_S(k-1)+\lambda_{ij}F_S(k-2)+\lambda_iF_S(k-4), & k=4,\dots,7\\ \lambda_{ijk}F_S(k-1)+\lambda_{ij}F_S(k-2)+\lambda_iF_S(k-4)+\lambda_0F_S(k-8), & k=8,9,\dots\end{cases}.$$

More general Poisson common shock models as proposed in [Mahamunulu, 1967] yield similar expressions for expected and cumulative expected allocations. △

5.2 Common mixture example

Next, we consider a multivariate mixed Poisson distribution. We induce dependence using a mixture random vector $\boldsymbol{\Theta}=(\Theta_1,\dots,\Theta_n)$ with $E[\Theta_i]=1$ for $i=1,\dots,n$. Consider a vector of conditionally independent rvs $(X_i|\Theta_i=\theta_i)\sim Poisson(\lambda_i\theta_i)$ for $i=1,\dots,n$. The joint pgf of $(X_1,\dots,X_n)$ is

$$\mathcal{P}_{\boldsymbol{X}}(t_1,\dots,t_n)=E_{\boldsymbol{\Theta}}\left[e^{\Theta_1\lambda_1(t_1-1)}\dots e^{\Theta_n\lambda_n(t_n-1)}\right]=\mathcal{M}_{\boldsymbol{\Theta}}(\lambda_1(t_1-1),\dots,\lambda_n(t_n-1)), \tag{28}$$

where $\mathcal{M}_{\boldsymbol{\Theta}}(t_1,\dots,t_n)$ is the moment generating function of $\boldsymbol{\Theta}$. Then, the OGF for expected allocations is

$$\mathcal{P}_S^{[1]}(t)=\lambda_1 t\left[\frac{\partial}{\partial x}\mathcal{M}_{\boldsymbol{\Theta}}(x,\lambda_2(t-1),\dots,\lambda_n(t-1))\right]\Bigg|_{x=\lambda_1(t-1)}. \tag{29}$$

Example 5.2 (Poisson-gamma common mixture)**.** *We consider a mixture distribution from a bivariate gamma common shock model from [Mathai and Moschopoulos, 1991]. Let us define three independent rvs $Y_i, i\in\{0,1,2\}$ where $Y_0\sim Gamma(\gamma_0,\beta_0)$, and $Y_i\sim Gamma(r_i-\gamma_0,r_i)$ for $i\in\{1,2\}$ with $0\leq\gamma_0\leq\min(r_1,r_2)$. Let $\Theta_i=\beta_0/r_iY_0+Y_i$ for $i=1,2$. Then the pair of rvs (Θ_1,Θ_2) follows a bivariate gamma distribution with marginals $\Theta_i\sim Ga(r_i,r_i)$, $i=1,2$ and γ_0 is a dependence parameter. Consequently, the mixed Poisson random vector (X_1,X_2) is a bivariate negative binomial rv. We have*

$$\mathcal{M}_{\Theta_1,\Theta_2}(x_1,x_2)=\left(1-\frac{x_1}{r_1}\right)^{-(r_1-\gamma_0)}\left(1-\frac{x_2}{r_2}\right)^{-(r_2-\gamma_0)}\left(1-\frac{x_1}{r_1}-\frac{x_2}{r_2}\right)^{-\gamma_0} \tag{30}$$

and

$$\frac{\partial}{\partial x_1}\mathcal{M}_{\Theta_1,\Theta_2}(x_1,x_2)=\left(\frac{r_1-\gamma_0}{r_1}\frac{1}{1-x_1/r_1}+\frac{\gamma_0}{r_1}\frac{1}{1-x_1/r_1-x_2/r_2}\right)\mathcal{M}_{\Theta_1,\Theta_2}(x_1,x_2). \tag{31}$$

It follows from (28) and (30) that

$$\mathcal{P}_S(t)=(1-\zeta_1(t-1))^{-(r_1-\gamma_0)}(1-\zeta_2(t-1))^{-(r_2-\gamma_0)}(1-\zeta_{12}(t-1))^{-\gamma_0},$$

where $\zeta_1=\lambda_1/r_1$, $\zeta_2=\lambda_2/r_2$ and $\zeta_{12}=\lambda_1/r_1+\lambda_2/r_2$. We recognize that S is the sum of three independent negative binomial rvs with parameters $(r_1-\gamma_0,1/(1-\zeta_1))$, $(r_2-\gamma_0,1/(1-\zeta_2))$ and $(\gamma_0,1/(1-\zeta_{12}))$. From (29) and (31) we get the following expression for the OGF for expected allocations:

$$\mathcal{P}_S^{[1]}(t)=\lambda_1 t\left(\frac{1-\gamma_0/r_1}{1-\zeta_1(t-1)}+\frac{\gamma_0/r_1}{1-\zeta_{12}(t-1)}\right)\mathcal{P}_S(t).$$

Finally, we can recover the expected allocations using FFT or with the convolution

$$[t^k]\mathcal{P}_S^{[1]}(t) = E\left[X_1 \times 1_{\{S=k\}}\right] = \lambda_1 \sum_{j=0}^{k-1}\left[\left(1-\frac{\gamma_0}{r_1}\right)\frac{1}{1+\zeta_1}\left(\frac{\zeta_1}{1+\zeta_1}\right)^j + \frac{\gamma_0}{r_1}\frac{1}{1+\zeta_{12}}\left(\frac{\zeta_{12}}{1+\zeta_{12}}\right)^j\right] f_S(k-1-j)$$

and

$$E\left[X_1 \times 1_{\{S\le k\}}\right] = \lambda_1 \sum_{j=0}^{k-1}\left[\left(1-\frac{\gamma_0}{r_1}\right)\left\{1-\left(\frac{\zeta_1}{1+\zeta_1}\right)^{j+1}\right\} + \frac{\gamma_0}{r_1}\left\{1-\left(\frac{\zeta_{12}}{1+\zeta_{12}}\right)^{j+1}\right\}\right] f_S(k-1-j) \tag{32a}$$

$$= \lambda_1 \sum_{j=0}^{k-1}\left[\left(1-\frac{\gamma_0}{r_1}\right)\frac{1}{1+\zeta_1}\left(\frac{\zeta_1}{1+\zeta_1}\right)^j + \frac{\gamma_0}{r_1}\frac{1}{1+\zeta_{12}}\left(\frac{\zeta_{12}}{1+\zeta_{12}}\right)^j\right] F_S(k-1-j). \tag{32b}$$

In (32a), the cumulative operator is applied to the geometric series, while in (32b) it is applied to the pmf of S. △

5.3 Archimedean copula example

Finally, we consider a multivariate Bernoulli distribution whose dependence structure is an Archimedean copula. Let $(I_1, \dots, I_n)$ form a random vector, where the marginal distributions are Bernoulli with success probability $q_i \in (0,1)$, for $i \in \{1, \dots, n\}$. Following [Marshall and Olkin, 1988], we define the random vector according to $\Pr(I_i = 1|\Theta = \theta) = r_i^\theta$, where Θ is a mixing random variable with strictly positive support. The relationship between the parameters r_i and q_i is

$$\Pr(I_i = 1) = E_\Theta\left[r_i^\Theta\right] = \mathcal{L}_\Theta(-\ln r_i),$$

from which it follows that $r_i = \exp\{-\mathcal{L}_\Theta^{-1}(q_i)\}$, where $\mathcal{L}_\Theta(t)$ and $\mathcal{L}_\Theta^{-1}(t)$ are respectively the Laplace-Stieltjes transform and the inverse Laplace-Stieltjes transform of the mixing rv. Further define the rv $X_i = b_i \times I_i$, with $b_i \in \mathbb{N}_1$ for $i \in \{1, \dots, n\}$. Note that the rvs $(X_i|\Theta = \theta)$ are conditionally independent, for $i \in \{1, \dots, n\}$ and $\theta > 0$. It follows that the multivariate pgf of $\boldsymbol{X} = (X_1, \dots, X_n)$ is

$$\mathcal{P}_{\boldsymbol{X}}(t_1, \dots, t_n) = E\left[\prod_{i=1}^n (1 - r_i^\Theta + r_i^\Theta t_i^{b_i})\right] = \int_0^\infty \prod_{i=1}^n (1 - r_i^\theta + r_i^\theta t_i^{b_i})\,\mathrm{d}F_\Theta(\theta).$$

This model has applications for instance in life insurance, where death benefits are usually known in advance, or for insurance-linked securities in situations where investors recover their initial investment unless a trigger event occurs before the maturity date.

We note that the underlying dependence structure in this model is an Archimedean copula, see, for instance, [Marshall and Olkin, 1988], Section 4.7.5.2 of [Denuit et al., 2006] or Section 7.4 of [McNeil et al., 2015] for the frailty construction of Archimedean copulas using common mixtures.

We consider the case where Θ is a discrete rv with support $\mathbb{N}_1$. Following the computational strategy from [Cossette et al., 2018], we select a threshold value $\theta^* = F_\Theta^{-1}(1-\varepsilon)$ for a small $\varepsilon > 0$ and we have

$$\mathcal{P}_S(t) = \sum_{\theta=1}^{\theta^*} \Pr(\Theta = \theta) \prod_{i=1}^n (1 - r_i^\theta + r_i^\theta t^{b_i}). \tag{33}$$

Note that when the components of the random vector are independent, the rv S is known as the generalized Poisson-binomial distribution [Zhang et al., 2018]. In the case of (33), we have a generalized Poisson-binomial distribution where the dependence structure is an Archimedean copula.

The OGF of the sequence of expected allocations for risk X_1 is

$$\mathcal{P}_S^{[1]}(t) = \sum_{\theta=1}^{\theta^*} \Pr(\Theta = \theta) r_1^{\theta} t^{b_1} \prod_{i=2}^{n} (1 - r_i^{\theta} + r_i^{\theta} t^{b_i}).$$

Example 5.3. *We consider a portfolio of $n = 6$ risks, with Θ following a shifted geometric rv with pmf $f_\Theta(k) = (1-\alpha)\alpha^{k-1}$, for $k \in \mathbb{N}_1$, with $\alpha = 0.5$. It follows that the underlying dependence structure is an Ali-Mikhail-Haq copula. Following [Cossette et al., 2018], we select a threshold $\varepsilon = 10^{-10}$, such that $\theta^* = 34$. The indemnity payments are the same as in Table 2. We present the validation curve, the pmf for the conditional means of risk X_3, and the pmf of S in Figure 8 for $\alpha \in \{0, 0.1, 0.5, 0.8, 0.95\}$. Increasing the dependence parameter has the effect of increasing the probability of zero contributions and of full ($X_3 = b_3$) contributions. For other allocation values, the expected allocations tend to cluster around the same value of 6 since increasing the dependence also increases the probability of mutual occurrence. Indeed, the probabilities for the outcomes $X_1 = 1, X_4 = 4$ and $X_5 = 5$ become more likely (resp, 0.006, 0.007, 0.011, 0.02 and 0.032 for $\alpha = 0$, 0.1, 0.5, 0.8 and 0.95), so more diversification occurs when the total costs are divisible by 10, as α increases.*

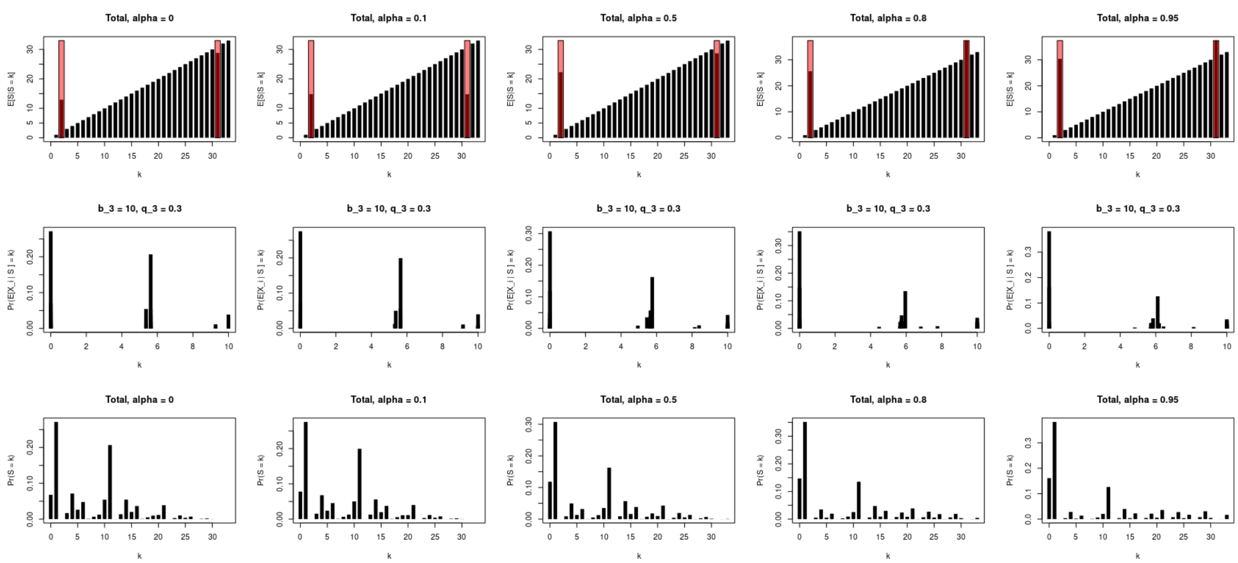

Figure 8: Pool of six participants. **À refaire en tikz quand ça sera final**

Next, we add 69 participants to the pool to investigate the effect of reducing the sparsity of the possible expected allocations. We present the validation curve, the values of $\Pr(E[X_3|S] = k)$ for $k \in \{0, \ldots, 441\}$ and $\alpha = 0, 0.1, 0.5, 0.8, 0.95$ in Figure 9. Note that the density function of S does not always converge to a normal distribution, hence central limit theorems do not apply. Indeed, the common mixture representation of the Ali-Mikhail-Haq copula generates multiple nodes for the pdf of S in this example. However, the OGF method with the FFT algorithm lets us extract the exact values of the pdf of expected allocations with ease. As we increase the dependence parameter,

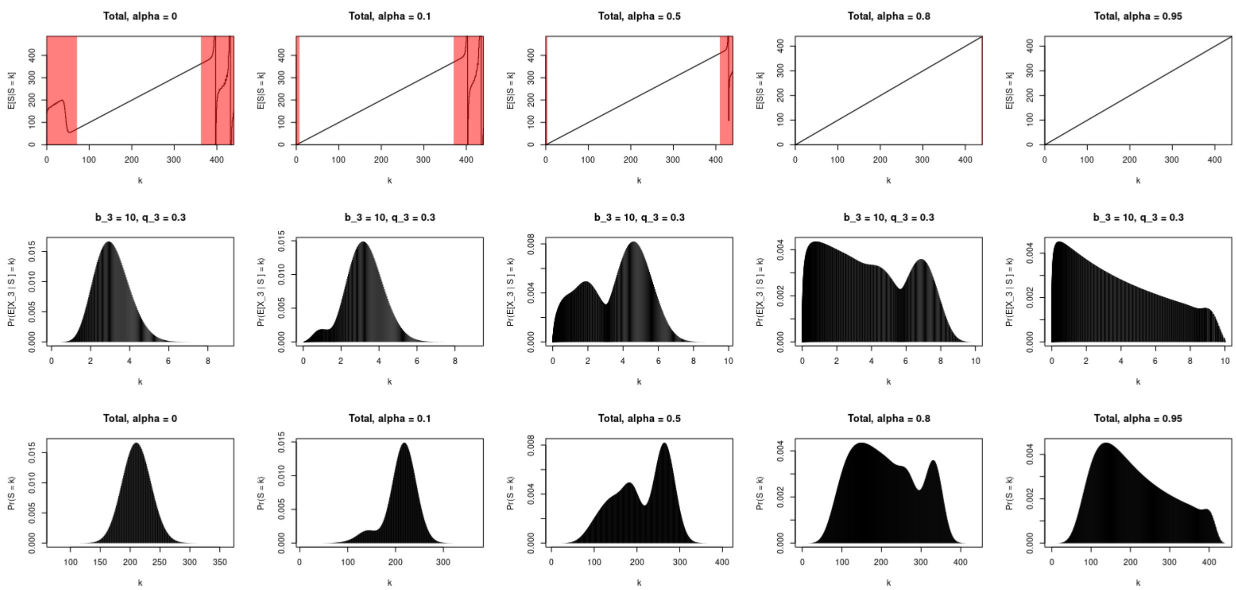

Figure 9: Pool of 75 participants. **À refaire en tikz quand ça sera final**

the probability mass of S and $E[X_3|S]$ is less concentrated around its means, thus the tail of the distributions have non-zero mass, so there are no numerical issues in the validation curve.

$\triangle$

6 Discussion

We proposed a transform method to compute the expected allocation, which has valuable applications in peer-to-peer insurance and capital allocation problems. The method simplifies solutions to some capital allocation problems and enables FFT-based algorithms for fast computations.

Future research could involve developing methods to quantify or correct aliasing errors for heavy-tailed distributions. In Section 3.2, we use a very large truncation point ($k_{max} = 2^{20}$). As computer processors continue to perform faster computations, it is convenient to increase the truncation point; however, it may also be convenient to provide methods that reduce this error source for efficiency's sake. The authors of [Grubel and Hermesmeier, 1999] quantify the aliasing error related to using the FFT algorithm to compute the pmf of compound distributions and also propose a tilting procedure to reduce this error. Developing a similar theory for the OGFs of expected allocations and expected cumulative allocations will increase these methods' efficiency.

Further research involves investigating the implications of this method in the continuous case. Letting $\mathcal{L}_{X_1,\ldots,X_n}$ denote the multivariate Laplace-Stieltjes transform of the vector $(X_1, \ldots, X_n)$, one can show that

$$-\frac{\partial}{\partial t_1}\mathcal{L}_{X_1,\ldots,X_n}(t_1,\ldots,t_n)\bigg|_{t_1=\cdots=t_n=t}$$

is the Laplace transform of $E\left[X_1 \times 1_{\{S=s\}}\right]$. One could use this formulation to obtain new closed-form expressions for expected allocations, compute expected allocations through numerical inversion of Laplace transforms, or develop asymptotic properties of expected allocations. We note that the Laplace transform of size-biased rvs is explored in, for instance, [Furman et al., 2020].

Another research topic involves the allocation of tail variance. In [Furman and Landsman, 2006], the authors introduce the tail variance, defined by

$$TV_\kappa(X) = Var(X|X \ > F_X^{-1}(\kappa)),$$

with $\kappa \in (0,1)$, and propose allocations via the tail covariance allocation rule,

$$TCov_\kappa(X_1|S) = Cov(X_1, S|S > F_S^{-1}(\kappa)) = \sum_{j=1}^{n} Cov(X_1, X_j|S > F_S^{-1}(\kappa)).$$

One can obtain efficient algorithms to compute the desired expectations once again. We have

$$E\left[X_1 X_j t^S\right] = \left\{ t_1 t_j \frac{\partial^2}{\partial t_1 \partial t_j} \mathcal{P}_{X_1,\ldots,X_n}(t_1,\ldots,t_n) \right\}\Big|_{t_1=\cdots=t_n=t}$$

for $j \in \{1,\ldots,n\} \setminus \{1\}$. The OGF for expected allocations for the second factorial moment is

$$E\left[X_1(X_1-1)t^S\right] = \left\{ t_1^2 \frac{\partial^2}{\partial t_1^2} \mathcal{P}_{X_1,S_{-1}}(t_1,t_2) \right\}\Big|_{t_1=t_2=t},$$

one can generalize the latter formula to kth factorial moments by taking subsequent derivatives. It follows that $E\left[X_1 X_j | S > k\right]$ and $E\left[X_1^2 | S > k\right]$ can be computed with

$$E\left[X_1 X_j \times 1_{\{S \leq k\}}\right] = [t^k] \left\{ \frac{E\left[X_1 X_j t^S\right]}{1-t} \right\}$$

and

$$E\left[X_1^2 \times 1_{\{S \leq k\}}\right] = [t^k] \left\{ \frac{E\left[X_1(X_1-1)t^S\right] + \mathcal{P}_S^{[1]}(t)}{1-t} \right\}.$$

A R code for the numerical applications

A.1 Large portfolio

```
set.seed(10112021)
n_participants <- 10000
kmax <- 2^13
lam <- list()
fc <- list()
mu <- list()

lambdas <- rexp(n_participants, 10)
rs <- sample(1:6, n_participants, replace = TRUE)
qs <- runif(n_participants, 0.4, 0.5)

# Assign parameters
```

```
for(i in 1:n_participants) {
  lam[[i]] <- lambdas[i]
  fci <- dnbinom(0:(kmax-2), rs[i], qs[i])
  fc[[i]] <- c(fci, 1 - sum(fci))
}

dft_fx <- list()
phic <- list()
cm <- list()

for(i in 1:n_participants) {
  dft_fx[[i]] <- exp(lam[[i]] * (fft(fc[[i]]) - 1))
  phic[[i]] <- fft(c(1:(kmax-1) * fc[[i]][-1], 0))
}

dft_fs <- Reduce("*", dft_fx)
fs <- Re(fft(dft_fs, inverse = TRUE))/kmax
e1 <- exp(-2i*pi*(0:(kmax-1))/kmax)

for(i in 1:n_participants) {
  dft_mu <- e1 * phic[[i]] * lam[[i]] * dft_fs
  mu[[i]] <- Re(fft(dft_mu, inverse = TRUE))/kmax
  cm[[i]] <- mu[[i]]/fs
}

cm_tot <- Reduce("+", cm)
cm_tot[1 + seq(4000, 5000, 100)] # Validation
```

A.2 Heavy tailed portfolio

```
library(actuar)
n <- 3

xmax <- 2^15
kmax <- 2^20
alphas <- seq(1.3, 1.9, 0.3)
lambdas <- 10 * (alphas - 1)

phis <- rep(1, kmax)
cm3 <- list()

for(i in 1:n) {
  fx <- discretize(ppareto(x, alphas[i], lambdas[i]), 0, xmax - 1,
                   method = "unbiased", lev = levpareto(x, alphas[i], lambdas[i]))
  phix <- fft(c(fx, rep(0, kmax - xmax)))
  phis <- phis * phix
}

fs3 <- Re(fft(phis, inverse = TRUE))/kmax

for(i in 1:n) {
  fx <- discretize(ppareto(x, alphas[i], lambdas[i]), 0, xmax - 1,
                   method = "unbiased", lev = levpareto(x, alphas[i], lambdas[i]))
  phix <- fft(c(fx, rep(0, kmax - xmax)))
  phi_deriv_x1 <- fft(c((1:(xmax - 1)) * fx[-1], rep(0, kmax - xmax + 1)))
  agf <- phis / phix * phi_deriv_x1 * exp(-2i*pi*(0:(kmax-1))/kmax)
  cm3[[i]] <- (Re(fft(agf, inverse = TRUE))/kmax / fs3)[1:xmax]
}
```

A.3 Small portfolio

```
n_participants <- 4
kmax <- 2^6
cmax <- 4
lam <- list(0.08, 0.08, 0.1, 0.1)
fc <- list(c(0, 0.1, 0.2, 0.4, 0.3, rep(0, kmax - cmax - 1)),
           c(0, 0.15, 0.25, 0.3, 0.3, rep(0, kmax - cmax - 1)),
           c(0, 0.1, 0.2, 0.3, 0.4, rep(0, kmax - cmax - 1)),
           c(0, 0.15, 0.25, 0.3, 0.3, rep(0, kmax - cmax - 1)))

dft_fx <- list()
phic <- list()
mu <- list()
conditional_mean <- list()

for(i in 1:n_participants) {
  dft_fx[[i]] <- exp(lam[[i]] * (fft(fc[[i]]) - 1))
  phic[[i]] <- fft(c(1:cmax * fc[[i]][2:(cmax + 1)], rep(0, kmax - cmax)))
}

dft_fs <- Reduce("*", dft_fx)
fs <- Re(fft(dft_fs, inverse = TRUE))/kmax
e1 <- exp(-2i*pi*(0:(kmax-1))/kmax)

for(i in 1:n_participants) {
  dft_mu <- e1 * phic[[i]] * lam[[i]] * dft_fs
  mu[[i]] <- Re(fft(dft_mu, inverse = TRUE))/kmax
  conditional_mean[[i]] <- mu[[i]]/fs
}

sapply(conditional_mean, "[[", 2) # Validation
conditional_mean_total <- Reduce("+", conditional_mean)
conditional_mean_total[1 + 1:10] # Validation
```

A.4 Archimedean copula example

```
set.seed(20220314)
n <- 6
# bi <- sample(1:10, n, replace = TRUE)
bi <- c(1, 3, 10, 4, 5, 10)
qi <- c(0.1, 0.15, 0.2, 0.25, 0.3)

kmax <- sum(bi) + 1
fft1 <- exp(-2i * pi * (0:(kmax - 1))/kmax)

alph <- 0
eps_theta <- 1e-10
theta_max <- max(2, floor(log(eps_theta)/log(alph)) + 1)
f_theta <- alph**(1:theta_max - 1) * (1 - alph)

LST_inv_geom <- function(u) log((1 - alph)/u + alph)
fft1 <- exp(-2i * pi * (0:(kmax - 1))/kmax)

qi <- runif(n)
ri <- exp(-LST_inv_geom(qi))

fgp_S <- function(s) {
```

```
  marginals <- apply(sapply(1:n, function(k) 1 - ri[k]^(1:theta_max) + ri[k]^(1:theta_max) * s^bi[k
    ]), 1, prod)
  sum(f_theta * marginals)
}

fgp_S <- Vectorize(fgp_S)
phis <- fgp_S(fft1)

fs <- (Re(fft(phis, inverse = TRUE))/kmax)

fgp_alloc_i <- function(s, i) {
  marginals <- bi[i] * ri[i]^(1:theta_max) * s^bi[i] * apply(sapply((1:n)[-i], function(k) 1 - ri[k
    ]^(1:theta_max) + ri[k]^(1:theta_max) * s^bi[k]), 1, prod)
  sum(f_theta * marginals)
}

fgp_alloc_i <- Vectorize(fgp_alloc_i)
phi_alloc_1 <- fgp_alloc_i(fft1, 1)
conditional_mean_1 <- (Re(fft(phi_alloc_1, inverse = TRUE))/kmax/fs)
round(conditional_mean_1, 3)
plot(conditional_mean_1, type = 's')
```

References

[Arratia et al., 2019] Arratia, R., Goldstein, L., and Kochman, F. (2019). Size bias for one and all. *Probability Surveys*, 16:1–61.

[Bargès et al., 2009] Bargès, M., Cossette, H., and Marceau, E. (2009). TVaR-based capital allocation with copulas. *Insurance: Mathematics and Economics*, 45(3):348–361.

[Chen, 1975] Chen, L. H. (1975). Poisson approximation for dependent trials. *The Annals of Probability*, 3(3):534–545.

[Cooley and Tukey, 1965] Cooley, J. W. and Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, 19(90):297–301.

[Cormen et al., 2009] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to Algorithms*. MIT Press.

[Cossette et al., 2018] Cossette, H., Marceau, E., Mtalai, I., and Veilleux, D. (2018). Dependent risk models with Archimedean copulas: A computational strategy based on common mixtures and applications. *Insurance: Mathematics and Economics*, 78:53–71.

[Denuit, 2019] Denuit, M. (2019). Size-Biased Transform and Conditional Mean Risk Sharing, with Application to P2p Insurance and Tontines. *ASTIN Bulletin: The Journal of the IAA*, 49(3):591–617.

[Denuit and Dhaene, 2012] Denuit, M. and Dhaene, J. (2012). Convex order and comonotonic conditional mean risk sharing. *Insurance: Mathematics and Economics*, 51(2):265–270.

[Denuit et al., 2006] Denuit, M., Dhaene, J., Goovaerts, M., and Kaas, R. (2006). *Actuarial Theory for Dependent Risks Measures, Orders and Models*. Wiley.

[Denuit et al., 2021a] Denuit, M., Dhaene, J., and Robert, C. Y. (2021a). Risk-sharing rules and their properties, with applications to peer-to-peer insurance. LIDAM Discussion Paper ISBA.

[Denuit et al., 2021b] Denuit, M., Hieber, P., and Robert, C. Y. (2021b). Mortality credits within large survivor funds. Technical Report 2021038, Université catholique de Louvain, Institute of Statistics, Biostatistics and Actuarial Sciences (ISBA).

[Denuit and Robert, 2020] Denuit, M. and Robert, C. Y. (2020). Large-loss behavior of conditional mean risk sharing. *ASTIN Bulletin*, 50(3):1093–1122.

[Denuit and Robert, 2021a] Denuit, M. and Robert, C. Y. (2021a). Conditional Tail Expectation Decomposition and Conditional Mean Risk Sharing for Dependent and Conditionally Independent Losses. *Methodology and Computing in Applied Probability*.

[Denuit and Robert, 2021b] Denuit, M. and Robert, C. Y. (2021b). From risk sharing to pure premium for a large number of heterogeneous losses. *Insurance: Mathematics and Economics*, 96:116–126.

[Dickson, 2017] Dickson, D. C. M. (2017). *Insurance Risk and Ruin*. International Series on Actuarial Science. Cambridge University Press, Cambridge, United Kingdom, second edition edition.

[Embrechts and Frei, 2009] Embrechts, P. and Frei, M. (2009). Panjer recursion versus FFT for compound distributions. *Mathematical Methods of Operations Research*, 69(3):497–508.

[Embrechts et al., 1993] Embrechts, P., Grübel, R., and Pitts, S. M. (1993). Some applications of the fast Fourier transform algorithm in insurance mathematics This paper is dedicated to Professor W. S. Jewell on the occasion of his 60th birthday. *Statistica Neerlandica*, 47(1):59–75.

[Furman et al., 2020] Furman, E., Hackmann, D., and Kuznetsov, A. (2020). On log-normal convolutions: An analytical–numerical method with applications to economic capital determination. *Insurance: Mathematics and Economics*, 90:120–134.

[Furman and Landsman, 2005] Furman, E. and Landsman, Z. (2005). Risk capital decomposition for a multivariate dependent gamma portfolio. *Insurance: Mathematics and Economics*, 37(3):635–649.

[Furman and Landsman, 2006] Furman, E. and Landsman, Z. (2006). Tail Variance Premium with Applications for Elliptical Portfolio of Risks. *ASTIN Bulletin*, 36(2):433–462.

[Furman and Landsman, 2008] Furman, E. and Landsman, Z. (2008). Economic Capital Allocations for Non-negative Portfolios of Dependent Risks. *ASTIN Bulletin: The Journal of the IAA*, 38(2):601–619.

[Graham et al., 1994] Graham, R. L., Knuth, D. E., and Patashnik, O. (1994). *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley, Reading, Mass, 2nd ed edition.

[Grimmett and Stirzaker, 2020] Grimmett, G. and Stirzaker, D. (2020). *Probability and Random Processes*. Oxford University Press.

[Grubel and Hermesmeier, 1999] Grubel, R. and Hermesmeier, R. (1999). Computation of compound distributions i: Aliasing errors and exponential tilting. *ASTIN Bulletin: The Journal of the IAA*, 29(2):197–214.

[Johnson et al., 1997] Johnson, N. L., Kotz, S., and Balakrishnan, N. (1997). *Discrete Multivariate Distributions*. Wiley Series in Probability and Statistics. Wiley.

[Klugman et al., 2013] Klugman, S. A., Panjer, H. H., and Willmot, G. E. (2013). *Loss Models: Further Topics*. John Wiley & Sons.

[Klugman et al., 2018] Klugman, S. A., Panjer, H. H., and Willmot, G. E. (2018). *Loss Models: From Data to Decisions*. Wiley Series in Probability and Statistics. Society of Actuaries ; John Wiley and Sons, Inc, [Schaumburg, IL] : Hoboken, NJ, fifth edition edition.

[Lindskog and McNeil, 2003] Lindskog, F. and McNeil, A. J. (2003). Common Poisson shock models: Applications to insurance and credit risk modelling. *ASTIN Bulletin*, 33(2):209–238.

[Mahamunulu, 1967] Mahamunulu, D. M. (1967). A note on regression in the multivariate Poisson distribution. *Journal of the American Statistical Association*, 62(317):251–258.

[Marceau, 2013] Marceau, É. (2013). *Modélisation et évaluation quantitative des risques en actuariat*. Springer Paris, Paris.

[Marshall and Olkin, 1988] Marshall, A. W. and Olkin, I. (1988). Families of Multivariate Distributions. *Journal of the American Statistical Association*, 83(403):834–841.

[Mathai and Moschopoulos, 1991] Mathai, A. M. and Moschopoulos, P. G. (1991). On a multivariate gamma. *Journal of Multivariate Analysis*, 39(1):135–153.

[McNeil et al., 2015] McNeil, A. J., Frey, R., and Embrechts, P. (2015). *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton Series in Finance. Princeton University Press, Princeton, NJ, revised edition edition.

[Panjer, 1981] Panjer, H. H. (1981). Recursive evaluation of a family of compound distributions. *ASTIN Bulletin*, 12(1):22–26.

[Panjer and Willmot, 1992] Panjer, H. H. and Willmot, G. E. (1992). *Insurance Risk Models*. Society of Actuaries.

[Sedgewick and Flajolet, 2013] Sedgewick, R. and Flajolet, P. (2013). *An Introduction to the Analysis of Algorithms*. Addison-Wesley, Upper Saddle River, NJ, 2nd ed edition.

[Tasche, 1999] Tasche, D. (1999). Risk contributions and performance measurement. *Report of the Lehrstuhl für mathematische Statistik, TU München*.

[Tasche, 2007] Tasche, D. (2007). Capital allocation to business units and sub-portfolios: the euler principle. *arXiv preprint arXiv:0708.2542*.

[Teicher, 1954] Teicher, H. (1954). On the multivariate Poisson distribution. *Scandinavian Actuarial Journal*, 1954(1):1–9.

[Wang, 1998] Wang, S. S. (1998). Aggregation of correlated risk portfolios: Models and algorithms. *Proceedings of the Casualty Actuarial Society*, page 92.

[Weiß and Aleksandrov, 2022] Weiß, C. H. and Aleksandrov, B. (2022). Computing (Bivariate) Poisson Moments Using Stein–Chen Identities. *The American Statistician*, 76(1):10–15.

[Wilf, 2006] Wilf, H. S. (2006). *Generatingfunctionology*. A K Peters, Wellesley, Mass, 3rd ed edition.

[Zhang et al., 2018] Zhang, M., Hong, Y., and Balakrishnan, N. (2018). The generalized Poisson-binomial distribution and the computation of its distribution function. *Journal of Statistical Computation and Simulation*, 88(8):1515–1527.