The AEP algorithm for the fast computation of the distribution of the sum of dependent random variables

Philipp Arbenz [a], Paul Embrechts [a], Giovanni Puccetti [b,*]

[a] *Department of Mathematics, ETH Zurich, 8092 Zurich, Switzerland*

[b] *Department of Mathematics for Decisions, University of Firenze, 50134 Firenze, Italy*

Abstract

We propose a new algorithm to compute numerically the distribution function of the sum of d dependent, non-negative random variables with given joint distribution.

Key words: Convolution, Distribution functions

AMS 2000 Subject Classification: 62E17, 65C20

1. Motivations and preliminaries

In probability theory, the exact calculation of the distribution function of the sum of d dependent random variables $X_1, \dots, X_d$ is a rather onerous task. Even assuming the knowledge of the joint distribution H of the vector $(X_1, \dots, X_d)$, one often has to rely on tools like Monte Carlo and Quasi-Monte Carlo methods. All these techniques warrant considerable expertise and more importantly need to be tailored to the specific problem under study. In this paper, we introduce a numerical procedure, called the AEP algorithm, which accurately calculates

$$\mathbb{P}[X_1 + \cdots + X_d \leq s], \tag{1.1}$$

at a fixed real threshold s and only uses the joint distribution H without the need for any specific adaptation.

[*] Corresponding author. Address: Dipartimento di Matematica per le Decisioni, via Lombroso 6/17, 50134 Firenze, Italy. Tel. +39-055-4796824, fax +39-055-4796800.

Email addresses: `philipp.arbenz@math.ethz.ch` (Philipp Arbenz), `embrechts@math.ethz.ch` (Paul Embrechts), `giovanni.puccetti@dmd.unifi.it` (Giovanni Puccetti).

Problems like the computation of (1.1) arise especially in insurance or finance when one has to calculate an overall capital charge in order to offset the risk position $S_d = X_1 + \cdots + X_d$ deriving from a portfolio of d random losses with known joint distribution H. The minimum capital requirement associated to S_d is typically calculated as the Value-at-Risk (i.e. quantile) for the distribution of S_d, and this at some high level of probability. Therefore, the calculation of a VaR-based capital requirement is equivalent to the computation of the distribution of S_d (see (1.1)). For a internationally active bank, this latter task is required for example under the terms of the New Basel Capital Accord (Basel II); see Basel Committee on Banking Supervision (2006).

An area of applications in quantitative risk management where our algorithm may be particularly useful is stress-testing. In this context one often has information on the marginal distributions of the underlying risks but wants to stress-test the interdependence between these risks, a concept that enters here is that of copula. Especially in the context of the current (credit) crisis, flexibility of the copula used when linking marginal distributions to a joint distribution has no doubt gained importance. See for instance Embrechts (2009).

Though the examples treated in this paper are mainly illustrative, the dimension d (≤ 5), the marginal assumptions, as well as the dependence structure (Clayton and Gumbel copula) used are typical for risk management applications in insurance and finance. For more information on this type of questions, see for instance SCOR (2008), Aas et al. (2007) and Bürgi et al. (2008).

In the following, we will denote (row) vectors in bold-face, e.g. $\mathbf{1} = (1, \ldots, 1) \in \mathbb{R}^d$, $d > 1$. $\boldsymbol{e}_k$ represents the k-th vector of the canonical basis of $\mathbb{R}^d$ and $D = \{1, \ldots, d\}$. Given a vector $\boldsymbol{b} = (b_1, \ldots, b_d) \in \mathbb{R}^d$ and a real number h, $\mathscr{Q}(\boldsymbol{b}, h) \subset \mathbb{R}^d$ denotes the hypercube defined as

$$\mathscr{Q}(\boldsymbol{b}, h) = \begin{cases} \times_{k=1}^{d} (b_k, b_k + h] & \text{if } h > 0, \\ \times_{k=1}^{d} (b_k + h, b_k] & \text{if } h < 0. \end{cases} \tag{1.2}$$

For notational purposes, we set $\mathscr{Q}(\boldsymbol{b}, 0) = \emptyset$. On some probability space $(\Omega, \mathfrak{A}, \mathbb{P})$, let the random variables $X_1, \ldots, X_d$ have joint d-variate distribution H. H induces the probability measure V_H on $\mathbb{R}^d$ via

$$V_H \left[\times_{i=1}^{d} (-\infty, x_i] \right] = H(x_1, \ldots, x_d).$$

We identify with $\boldsymbol{i}_0, \ldots, \boldsymbol{i}_N$ all the 2^d vectors in $\{0,1\}^d$, i.e. $\boldsymbol{i}_0 = (0, \ldots, 0)$, $\boldsymbol{i}_k = \boldsymbol{e}_k, k \in D$, and so on, $\boldsymbol{i}_N = \mathbf{1} = (1, \ldots, 1)$, where $N = 2^d - 1$. $\#\boldsymbol{i} = \sum_{k=1}^{d} i_k$ denotes the number of 1 in the vector $\boldsymbol{i}$, e.g. $\#\boldsymbol{i}_0 = 0, \#\boldsymbol{i}_N = d$. The V_H-measure of a hypercube $\mathscr{Q}(\boldsymbol{b}, h), h > 0$ can be easily calculated as

$$V_H [\mathscr{Q}(\boldsymbol{b}, h)] = \mathbb{P}[X_k \in (b_k, b_k + h], k \in D] = \sum_{j=0}^{N} (-1)^{d - \#\boldsymbol{i}_j} H\big(\boldsymbol{b} + h\boldsymbol{i}_j\big). \tag{1.3}$$

The case $h < 0$ is analogous. If necessary, (1.3) can be also expressed in terms of the survival function $\overline{H} = 1 - H$. Moreover, $\mathscr{S}(\boldsymbol{b}, h) \subset \mathbb{R}^d$ denotes the d-dimensional simplex defined as

$$\mathscr{S}(\boldsymbol{b}, h) = \begin{cases} \left\{(x_1, \ldots, x_d) \in \mathbb{R}^d : x_k - b_k > 0, k \in D \text{ and } \sum_{k=1}^{d} (x_k - b_k) \leq h\right\}, & \text{if } h > 0, \\ \left\{(x_1, \ldots, x_d) \in \mathbb{R}^d : x_k - b_k \leq 0, k \in D \text{ and } \sum_{k=1}^{d} (x_k - b_k) > h\right\}, & \text{if } h < 0. \end{cases} \tag{1.4}$$

Again $\mathscr{S}(\boldsymbol{b}, 0) = \emptyset$. We finally denote by λ_d the Lebesgue measure on $\mathbb{R}^d$. For instance, the Lebesgue measure of the simplex $\mathscr{S}(\boldsymbol{b}, h)$ is given by

$$\lambda_d [\mathscr{S}(\boldsymbol{b}, h)] = \frac{|h|^d}{d!}. \tag{1.5}$$

2. Description of the AEP algorithm for $d = 2$

Throughout the paper, we assume the random variables $X_1, \ldots, X_d$ to be non-negative, i.e. $\mathbb{P}[X_k \le 0] = 0, k \in D$. The extension to random variables bounded from below is straightforward and will be illustrated below. We assume the knowledge of the joint distribution H of the vector $(X_1, \ldots, X_d)$ and define $S_d = X_1 + \cdots + X_d$. Our aim is then to calculate

$$\mathbb{P}[S_d \le s] = V_H[\mathscr{S}(\mathbf{0}, s)],$$

and this at a fixed positive threshold s.

Due to (1.3), it is very easy to compute the V_H-measure of hypercubes in $\mathbb{R}^d$. The idea behind the AEP algorithm is then to approximate the simplex $\mathscr{S}(\mathbf{0}, s)$ by hypercubes. Before going to the general case, we first briefly illustrate our method for dimension $d = 2$.

As illustrated in Figure 1, the V_H-measure of the simplex $\mathscr{S}_1^1 = \mathscr{S}(\mathbf{0}, s)$ can be proxied by the V_H-measure of the hypercube $\mathscr{Q}_1^1 = \mathscr{Q}(\mathbf{0}, \alpha s)$, with $\alpha \in [1/2, 1)$. The error committed by using this approximation can be expressed in terms of the measure of the three simplexes

$$\mathscr{S}_2^1 = \mathscr{S}((0, \alpha s), (1-\alpha)s), \ \mathscr{S}_2^2 = \mathscr{S}((\alpha s, 0), (1-\alpha)s) \text{ and } \mathscr{S}_2^3 = \mathscr{S}((\alpha s, \alpha s), (1-2\alpha)s).$$

Formally, we have

$$\mathscr{S}(\mathbf{0}, s) = \left(\mathscr{Q}_1^1 \cup \mathscr{S}_2^1 \cup \mathscr{S}_2^2\right) \setminus \mathscr{S}_2^3, \text{ for all } \alpha \in [1/2, 1). \tag{2.1}$$

Since $\alpha \in [1/2, 1)$, the sets $\mathscr{S}_2^1, \mathscr{S}_2^2$ and $\mathscr{Q}_1^1$ are pairwise disjoint. Note also that $\mathscr{S}_2^3 \subset \mathscr{Q}_1^1$. The V_H-measure of $\mathscr{S}(\mathbf{0}, s)$ can thus be written as

$$V_H[\mathscr{S}(\mathbf{0}, s)] = V_H\left[\mathscr{Q}_1^1\right] + V_H\left[\mathscr{S}_2^1\right] + V_H\left[\mathscr{S}_2^2\right] - V_H\left[\mathscr{S}_2^3\right].$$

With the notation $s_2^1 = s_2^2 = 1$ and $s_2^3 = -1$, we translate the equation above into

$$V_H[\mathscr{S}(\mathbf{0}, s)] = V_H\left[\mathscr{Q}_1^1\right] + \sum_{k=1}^{3} s_2^k V_H\left[\mathscr{S}_2^k\right]. \tag{2.2}$$

As a first approximation of $V_H[\mathscr{S}(\mathbf{0}, s)]$ we define the value

$$P_1(s) = V_H\left[\mathscr{Q}_1^1\right] = H(\alpha s, \alpha s) - H(0, \alpha s) - H(0, \alpha s) + H(0, 0).$$

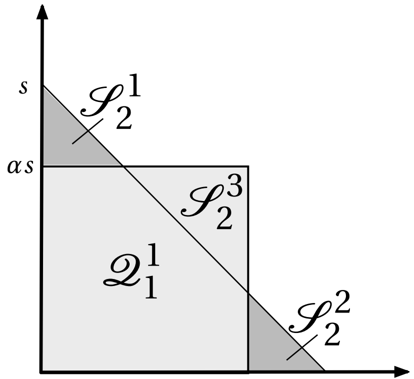

Fig. 1. Decomposition (2.1) of the two-dimensional simplex $\mathscr{S}((0,0), s)$.

Using (2.2), the error committed by considering $P_1(s)$ instead of $V_H[\mathscr{S}(\mathbf{0},s)]$ can be expressed in terms of the V_H-measure of the three simplexes $\mathscr{S}_2^k$ defined above, i.e.

$$V_H[\mathscr{S}(\mathbf{0},s)] - P_1(s) = \sum_{k=1}^{3} s_2^k V_H\left[\mathscr{S}_2^k\right]. \tag{2.3}$$

At this point, we can apply to each of the $\mathscr{S}_2^k$'s a decomposition analogous to the one given in (2.2) for $\mathscr{S}_1^1 = \mathscr{S}(\mathbf{0},s)$, in order to obtain a better approximation of their measures, and hence of the measure of $\mathscr{S}_1^1$. The only difference between the first and the following step is that we have to keep track whether the measure of a simplex has to be added to or subtracted from the next approximation $P_2(s)$ of $V_H[\mathscr{S}(\mathbf{0},s)]$. The value s_2^k, associated to each simplex $\mathscr{S}_2^k$, indicates whether the corresponding measure is to be added ($s_2^k = 1$) or subtracted ($s_2^k = -1$). The next approximation $P_2(s)$ will be defined such that the difference $V_H[\mathscr{S}(\mathbf{0},s)] - P_2(s)$ is the sum of the V_H-measures of a total of nine simplexes produced by the decompositions of the three $\mathscr{S}_2^k$'s. The nine simplexes are then passed as input to the third iteration and so on.

Before formally defining the algorithm in arbitrary dimension d, it is important to remark the following points:

- We will prove the set decomposition (2.1) to hold analogously in arbitrary dimension d for every choice of $\alpha \in [1/d, 1)$. Unfortunately, the corresponding simplexes $\mathscr{S}_n^k$ are in general not disjoint for $d > 2$. This will imply a more complicated formula for the general V_H-measure decomposition.
- Equation (2.2) depends on the choice of α. We will study in Section 4 which is the best value of α to use.

3. Description of the AEP algorithm for arbitrary d

Recall that in Section 1 we denoted with $\boldsymbol{i}_0, \ldots, \boldsymbol{i}_N$ all the 2^d vectors in $\{0,1\}^d$, and $N = 2^d - 1$. Let also $\alpha \in [1/d, 1)$. At the beginning of the n-th iteration ($n \in \mathbb{N}$) the algorithm receives as input N^{n-1} simplexes which we denote by $\mathscr{S}_n^k = \mathscr{S}(\boldsymbol{b}_n^k, h_n^k)$, for $k = 1, \ldots, N^{n-1}$. To each simplex is associated the value $s_n^k \in \{-1, 1\}$, which indicates whether the measure of the simplex has to be added ($s_n^k = 1$) or subtracted ($s_n^k = -1$) in order to compute an approximation of $V_H[\mathscr{S}(\mathbf{0},s)]$.

Each simplex $\mathscr{S}_n^k$ is then decomposed via one hypercube $\mathscr{Q}_n^k = \mathscr{Q}(\boldsymbol{b}_n^k, \alpha h_n^k)$ and N simplexes $\mathscr{S}_{n+1}^k = \mathscr{S}(\boldsymbol{b}_{n+1}^k, h_{n+1}^k)$. In fact, we prove in Appendix A the rather technical fact that the V_H-measure of each simplex $\mathscr{S}_n^k$ can be calculated as:

$$V_H\left[\mathscr{S}_n^k\right] = V_H\left[\mathscr{Q}_n^k\right] + \sum_{j=1}^{N} m^j V_H\left[\mathscr{S}_{n+1}^{Nk-N+j}\right], \tag{3.1}$$

where the sequences $\boldsymbol{b}_n^k, h_n^k$ and m^j are defined by their initial values $\boldsymbol{b}_1^1 = \mathbf{0}, h_1^1 = s$ and

$$\boldsymbol{b}_{n+1}^{Nk-N+j} = \boldsymbol{b}_n^k + \alpha h_n^k \boldsymbol{i}_j, \quad h_{n+1}^{Nk-N+j} = (1 - \#\boldsymbol{i}_j \alpha) h_n^k, \quad m^j = \begin{cases} (-1)^{1+\#\boldsymbol{i}_j} & \text{if } \#\boldsymbol{i}_j < 1/\alpha, \\ 0 & \text{if } \#\boldsymbol{i}_j = 1/\alpha, \\ (-1)^{d+1-\#\boldsymbol{i}_j} & \text{if } \#\boldsymbol{i}_j > 1/\alpha, \end{cases} \tag{3.2}$$

and this for all $j = 1, \ldots, N$ and $k = 1, \ldots, N^{n-1}$. At this point we note that changing the value $\boldsymbol{b}_1^1$ one can apply the algorithm to the case in which the random vector $(X_1, \ldots, X_d)$ assumes also negative values but is still bounded from below by $\boldsymbol{b}_1^1$.

We define the sequence $P_n(s)$ as the sum of the V_H-measures of the $\mathscr{Q}_n^k$, multiplied by the corresponding s_n^k, as

$$P_n(s) = P_{n-1}(s) + \sum_{k=1}^{N^{n-1}} s_n^k V_H\left[\mathscr{Q}_n^k\right] = \sum_{i=1}^{n}\sum_{k=1}^{N^{i-1}} s_i^k V_H\left[\mathscr{Q}_i^k\right], \tag{3.3}$$

where $P_0(s) = 0$ and the s_n^k are defined by $s_1^1 = 1$ and

$$s_{n+1}^{Nk-N+j} = s_n^k m^j, \text{ for all } j = 1, \ldots, N \text{ and } k = 1, \ldots, N^{n-1}. \tag{3.4}$$

We will see in the following that the sequence $P_n(s)$ converges to $V_H[\mathscr{S}(\mathbf{0}, s)]$, under weak assumptions on the distribution H. Moreover, note that $P_n(s)$ is straightforward to calculate via (1.3). The $(N^{n-1}) \times N = N^n$ simplexes $\mathscr{S}_{n+1}^k$, generated by (3.1) are then passed to the $(n+1)$-th iteration in order to approximate their V_H-measures with the measures of the hypercubes $\mathscr{Q}_{n+1}^k$.

As a first step, to show that $P_n(s)$ tends to $V_H[\mathscr{S}(\mathbf{0}, s)]$, we prove that the error committed by considering $P_n(s)$ instead of $V_H[\mathscr{S}(\mathbf{0}, s)]$ is given by the sum of the V_H-measures of the simplexes $\mathscr{S}_{n+1}^k$ passed to the $(n+1)$-th iteration, multiplied by the corresponding s_{n+1}^k.

Theorem 3.1 *With the notation introduced above, we have that*

$$V_H[\mathscr{S}(\mathbf{0}, s)] - P_n(s) = \sum_{k=1}^{N^n} s_{n+1}^k V_H\left[\mathscr{S}_{n+1}^k\right]. \tag{3.5}$$

Proof. We prove the theorem by induction on n. Note that (3.5) corresponds to (3.1) when $n = 1$. Now assume by induction that

$$V_H[\mathscr{S}(\mathbf{0}, s)] = P_{n-1}(s) + \sum_{k=1}^{N^{n-1}} s_n^k V_H\left[\mathscr{S}_n^k\right],$$

which, recalling (3.1), (3.3) and (3.4), yields

$$\begin{aligned} V_H[\mathscr{S}(\mathbf{0}, s)] &= P_{n-1}(s) + \sum_{k=1}^{N^{n-1}} s_n^k V_H\left[\mathscr{Q}_n^k\right] + \sum_{k=1}^{N^{n-1}} s_n^k \left(\sum_{j=1}^{N} m^j V_H\left[\mathscr{S}_{n+1}^{Nk-N+j}\right]\right) \\ &= P_n(s) + \sum_{k=1}^{N^{n-1}}\sum_{j=1}^{N} s_n^k m^j V_H\left[\mathscr{S}_{n+1}^{Nk-N+j}\right] \\ &= P_n(s) + \sum_{k=1}^{N^{n-1}}\sum_{j=1}^{N} s_{n+1}^{Nk-N+j} V_H\left[\mathscr{S}_{n+1}^{Nk-N+j}\right] = P_n(s) + \sum_{k=1}^{N^n} s_{n+1}^k V_H\left[\mathscr{S}_{n+1}^k\right]. \qquad \square \end{aligned}$$

We are now ready to give a sufficient condition for the convergence of the sequence $P_n(s)$ to $V_H[\mathscr{S}(\mathbf{0}, s)]$. The idea of the proof is that if the total Lebesgue measure of the new N simplexes $\mathscr{S}_{n+1}^{Nk-N+j}$, $j = 1, \ldots, N$, generated by the simplex $\mathscr{S}_n^k$, is smaller than the Lebesgue measure of $\mathscr{S}_n^k$ itself, then, by assuming continuity of H, the error (3.5) will go to zero. Let us define $e_n = \sum_{k=1}^{N^n} \lambda_d\left[\mathscr{S}_{n+1}^k\right]$ as the sum of the Lebesgue measure of the simplexes passed to iteration $n+1$. We call *volume factor* $f(\alpha)$ the ratio between the sum of the Lebesgue measure of the simplexes in two subsequent iterations, i.e. $f(\alpha) = e_n / e_{n-1}$. Recalling the formula (1.5) for the λ_d-measure of a simplex, we have that

$$\sum_{j=1}^{N} \lambda_d \left[\mathscr{S}_{n+1}^{Nk-N+j}\right] = \sum_{j=1}^{N} \frac{\left|\left(1-\#\boldsymbol{i}_j\alpha\right) h_n^k\right|^d}{d!} = \sum_{j=1}^{d} \binom{d}{j} \frac{|1-j\alpha|^d \left|h_n^k\right|^d}{d!}.$$

Observing that the N simplexes $\mathscr{S}_{n+1}^{Nk-N+j}$, $j = 1,\dots,N$, are generated by the simplex $\mathscr{S}_n^k$, we use the above equation to conclude that

$$f(\alpha) = \frac{e_n}{e_{n-1}} = \frac{\sum_{k=1}^{N^n} \lambda_d \left[\mathscr{S}_{n+1}^k\right]}{\sum_{k=1}^{N^{n-1}} \lambda_d \left[\mathscr{S}_n^k\right]} = \frac{\sum_{k=1}^{N^{n-1}} \sum_{j=1}^{N} \lambda_d \left[\mathscr{S}_{n+1}^{Nk-N+j}\right]}{\sum_{k=1}^{N^{n-1}} \lambda_d \left[\mathscr{S}_n^k\right]} = \frac{\sum_{k=1}^{N^{n-1}} \sum_{j=1}^{d} \binom{d}{j} \frac{|1-j\alpha|^d |h_n^k|^d}{d!}}{\sum_{k=1}^{N^{n-1}} \lambda_d \left[\mathscr{S}_n^k\right]}$$

$$= \frac{\frac{1}{d!} \sum_{k=1}^{N^{n-1}} \left|h_n^k\right|^d \sum_{j=1}^{d} \binom{d}{j} |1-j\alpha|^d}{\frac{1}{d!} \sum_{k=1}^{N^{n-1}} \left|h_n^k\right|^d} = \sum_{j=1}^{d} \binom{d}{j} |1-j\alpha|^d.$$

A sufficient condition for the convergence of the AEP algorithm can then be expressed in terms of the volume factor $f(\alpha)$. We first assume H to be continuous with a bounded density.

Theorem 3.2 *Assume that V_H has a bounded density v_H. If the volume factor satisfies $f(\alpha) < 1$, then*

$$\lim_{n\to\infty} P_n(s) = V_H\left[\mathscr{S}(\mathbf{0}, s)\right]. \tag{3.6}$$

Proof. Since V_H has a density v_H bounded by the positive constant $c > 0$, using (3.5) we have that

$$\begin{aligned} \left|V_H\left[\mathscr{S}(\mathbf{0}, s)\right] - P_n(s)\right| &= \left|\sum_{k=1}^{N^n} s_{n+1}^k V_H\left[\mathscr{S}_{n+1}^k\right]\right| = \left|\sum_{k=1}^{N^n} \int_{\mathscr{S}_{n+1}^k} s_{n+1}^k \, dH\right| \le \sum_{k=1}^{N^n} \left|\int_{\mathscr{S}_{n+1}^k} s_{n+1}^k c \, d\lambda_d\right| \\ &\le c \sum_{k=1}^{N^n} \int_{\mathscr{S}_{n+1}^k} \left|s_{n+1}^k\right| d\lambda_d = c \sum_{k=1}^{N^n} \int_{\mathscr{S}_{n+1}^k} d\lambda_d = c \sum_{k=1}^{N^n} \lambda_d\left[\mathscr{S}_{n+1}^k\right] = c e_n. \end{aligned}$$

We conclude by noting that, since $e_n > 0$ and $e_n / e_{n-1} = f(\alpha) < 1$ by assumption, e_n goes to zero exponentially in n. □

In order for (3.6) to hold, it is sufficient that v_H is bounded on $\bigcup_{k=1}^{N^n} \mathscr{S}_{n+1}^k$ for n large enough. Define the curve Γ_s as

$$\Gamma_s = \left\{(x_1,\dots,x_d) \in \mathbb{R}^d : \sum_{k=1}^{d} x_k = s\right\}. \tag{3.7}$$

The following theorem states that the L^1-distance from the curve Γ_s of each point in $\bigcup_{k=1}^{N^n} \mathscr{S}_{n+1}^k$ is bounded by $\gamma^n s$, for some $\gamma \in (0,1)$. This implies that this distance goes to zero in the limit as $n \to \infty$. For Theorem 3.2 to hold, it is then sufficient to require that H has a bounded density only in a neighborhood of Γ_s. We will discuss this assumption further in Section 8.

Theorem 3.3 *If $\boldsymbol{x} \in \bigcup_{k=1}^{N^n} \mathscr{S}_{n+1}^k$, then its L^1-distance from the curve Γ_s is bounded by $\gamma^n s$, with $\gamma = \max\{1-\alpha, |1-d\alpha|\} < 1$.*

Proof. We denote by $b_n^{k,r}$ and, respectively, i_j^r for $r \in D$ the d components of the vectors $\boldsymbol{b}_n^k$ and, respectively, $\boldsymbol{i}_j$. We prove by induction on n that

$$\sum_{r=1}^{d} b_n^{k,r} + h_n^k = s, \text{ for all } k = 1,\dots,N^{n-1} \text{ and } n \ge 1. \tag{3.8}$$

For $n = 1$, the statement is true since there is only one simplex with $\boldsymbol{b}_1^1 = \mathbf{0}$ and $h_1^1 = s$. Now assume the statement holds for n. By (3.2), we have that, for all $j = 1,\dots,N$ and $k = 1,\dots,N^{n-1}$,

$$\sum_{r=1}^{d} b_{n+1}^{Nk-N+j,r} + h_{n+1}^{Nk-N+j}$$

$$= \sum_{r=1}^{d} \left(b_n^{k,r} + \alpha h_n^k \boldsymbol{i}_j^r\right) + (1-\#\boldsymbol{i}_j\alpha)h_n^k = \sum_{r=1}^{d} b_n^{k,r} + \alpha h_n^k \sum_{r=1}^{d} \boldsymbol{i}_j^r + h_n^k - \alpha h_n^k \#\boldsymbol{i}_j =$$

$$\sum_{r=1}^{d} b_n^{k,r} + \alpha h_n^k \#\boldsymbol{i}_j + h_n^k - \alpha h_n^k \#\boldsymbol{i}_j = \sum_{r=1}^{d} b_n^{k,r} + h_n^k = s,$$

where the last inequality is the induction assumption. Due to (3.8), every simplex $\mathscr{S}_{n+1}^k$ generated by the AEP algorithm has the diagonal face lying on the curve Γ_s. As a consequence, the L^1-distance from Γ_s of each point in $\mathscr{S}_{n+1}^k$ is strictly smaller than the distance of the vector $\boldsymbol{b}_{n+1}^k$, which is $\left|h_{n+1}^k\right|$. For a fixed n and $k = 1,\dots N^{n-1}$, we have that $\left|h_{n+1}^{Nk-N+j}\right| \le \gamma \left|h_n^k\right|$ for all $j = 1,\dots,N$. Hence

$$\max_{k=1,\dots,N^n} \left|h_{n+1}^k\right| = \gamma^n h_1^1 = \gamma^n s, \text{ with } \gamma \in (0,1), \tag{3.9}$$

where equality holds as, for every $n \ge 1$, we have that $\left|h_{n+1}^{Nk-N+j}\right| = \gamma\left|h_n^k\right|$ for $j = 1$ or $j = N$. □

4. Choice of α

As already remarked, the AEP algorithm depends on the choice of the parameter α. It is important to note that, in general, an optimal choice of α would depend on the measure V_H. In the proof of Theorem 3.2, we proved that

$$|P_n(s) - V_H[\mathscr{S}(\mathbf{0}, s)]| \le Cf(\alpha)^n,$$

where C is a positive constant. Since we want to keep our algorithm independent of the choice of the distribution H, we suggest to use the α^* which minimizes $f(\alpha)$, i.e.

$$\alpha^* = \text{argmin}_{\alpha\in[\frac{1}{d},1)} f(\alpha) = \frac{2}{d+1}.$$

Some values of α^*, and the corresponding optimal volume factors $f(\alpha^*)$ are given in Table 1 for dimensions $d \le 7$.

d	α^*	$f(\alpha^*)$	d	α^*	$f(\alpha^*)$
2	$\frac{2}{3}$	$\frac{1}{3}$	5	$\frac{1}{3}$	$\frac{23}{27}$
3	$\frac{1}{2}$	$\frac{1}{2}$	6	$\frac{2}{7}$	>1
4	$\frac{2}{5}$	$\frac{83}{125}$	7	$\frac{1}{4}$	>1

Table 1
Values for α^* and $f(\alpha^*)$ for dimensions $d \le 7$.

We will show that using α^* has several desirable consequences. First of all, when $\alpha = \alpha^*$ and the dimension d is odd, in the measure decomposition (3.1), a number of $\binom{d}{(d+1)/2}$ simplexes have the corresponding coefficient m^j equal to zero and can therefore be neglected, increasing the computational efficiency of the algorithm. For example, in the decomposition of a three dimensional simplex, the algorithm generates only 4 new simplexes at every iteration with $\alpha = \alpha^*$, instead of the $2^d - 1 = 7$ generated with any other feasible value of α. Hence, for $\alpha = \alpha^*$, the number of new simplexes generated at each step is given by the function

$$f_S(d) = \begin{cases} 2^d - 1, & \text{if } d \text{ is even,} \\ 2^d - 1 - \binom{d}{(d+1)/2}, & \text{if } d \text{ is odd;} \end{cases} \tag{4.1}$$

see Section 5 for further details on this.

The choice $\alpha = \alpha^*$ will show to be convenient also because, as stated in the proof of Theorem 3.3, we have that

$$[0, +\infty)^d \cap \left(\bigcup_{k=1}^{N^{n-1}} \mathscr{S}_n^k \right) \subset \mathscr{S}(\mathbf{0}, (1+\gamma^n)s) \setminus \mathscr{S}(\mathbf{0}, (1-\gamma^n)s) \tag{4.2}$$

with $\gamma = \max\{1-\alpha, |1-d\alpha|\}$. It is straightforward to see that α^* also minimizes γ.

As illustrated in Table 1, Theorem 3.2 states the convergence of the sequence $P_n(s)$ when $d \le 5$. Various elements affect the speed at which $P_n(s)$ converges. First of all, it is in general always possible to put probability mass in a smooth way in a neighborhood of the curve Γ_s in order to seriously affect the convergence rate of $P_n(s)$. For the distributions of financial and actuarial interest used in Section 6, the algorithm performs very well; slow convergence is typically restricted to more pathological cases. We also have to consider that, for the same distribution H, it is in general required to compute the distribution of S_d at different thresholds s; see Section 6. Problems as described in Section 8 below may then occur only at a few points s.

A more relevant issue is the fact that the memory required by the algorithm to run the n-th iteration increases exponentially in n. At each iteration of the algorithm, every simplex $\mathscr{S}_n^k$ produces one hypercube and a number $f_S(d)$ of new simplexes to be passed to the following iteration; see (4.1). The computational effort in the $(n-1)$-th step thus increases as $O(f_S(d)^n)$. While the dimensions $d \le 5$ are manageable, as reported in Section 6, the numerical complexity for $d \ge 6$ increases considerably and quickly exhausts memory of a standard computer.

Finally, choosing $\alpha = \alpha^*$ also allows to increase the accuracy of the AEP algorithm and, under slightly stronger assumptions on H, will lead to convergence of AEP in higher dimensions, as we discuss in Section 5 below.

We now give some examples of the first step ($n = 1$) of the measure decomposition (3.1) obtained by choosing $\boldsymbol{b} = \mathbf{0}, s = 1, \alpha = \alpha^*$, for $d = 2, 3$:

– In the case $d = 2$, with $\alpha = 2/3$, we obtain (see Figure 2):

$$\begin{aligned} V_H[\mathscr{S}((0,0),1)] = V_H[\mathscr{Q}((0,0),2/3)] &+ V_H[\mathscr{S}((0,2/3),1/3)] \\ &+ V_H[\mathscr{S}((2/3,0),1/3)] - V_H[\mathscr{S}((2/3,2/3),-1/3)]. \end{aligned}$$

– In the case $d = 3$, with $\alpha = 1/2$, we obtain (see Figure 3):

$$\begin{aligned} V_H[\mathscr{S}((0,0,0),1)] &= V_H[\mathscr{Q}((0,0,0),1/2)] + V_H[\mathscr{S}((1/2,0,0),1/2)] \\ &+ V_H[\mathscr{S}((0,1/2,0),1/2)] + V_H[\mathscr{S}((0,0,1/2),1/2)] - V_H[(1/2,1/2,1/2),-1/2)]. \end{aligned}$$

5. An improvement of the numerical accuracy of the algorithm via *extrapolation*

In this section, we introduce a method to increase the accuracy of the AEP algorithm. This method is based on the choice $\alpha = \alpha^*$ as discussed in Section 4. To this end, we will make the stronger assumption that the joint distribution H has a twice continuously differentiable density v_H, with bounded derivatives. This will allow to approximate the density v_H by its linear Taylor expansion, providing a good estimate of the approximation error of AEP after a number of iterations.

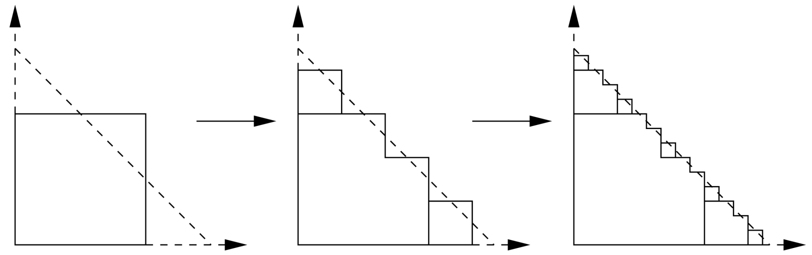

Fig. 2. The decomposition of a simplex by the AEP algorithm in the case $d = 2$.

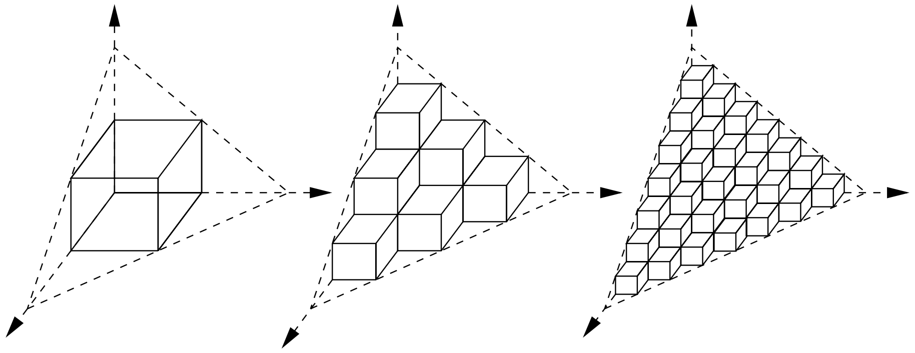

Fig. 3. The decomposition of a simplex by the AEP algorithm in the case $d = 3$.

We first need two simple integration results. Denoting by $\mathscr{S}_{d-1}$ a simplex in dimension $(d-1)$, for all $s > 0$, we have

$$\begin{aligned}\int_{\mathscr{S}(\mathbf{0},s)} x_d d\boldsymbol{x} &= \int_0^s \int_0^{s-x_d} \cdots \int_0^{s-\sum_{k=3}^d x_k} \int_0^{s-\sum_{k=2}^d x_k} x_d \otimes_{k=1}^d (\mathrm{d}x_k) \\ &= \int_0^s x_d \int_0^{s-x_d} \cdots \int_0^{s-\sum_{k=3}^d x_k} \int_0^{s-\sum_{k=2}^d x_k} \otimes_{k=1}^d (\mathrm{d}x_k) \\ &= \int_0^s x_d \lambda_{d-1} \left[\mathscr{S}_{d-1}(\mathbf{0}, s-x_d)\right] \mathrm{d}x_d = \int_0^s x_d \frac{(s-x_d)^{d-1}}{(d-1)!} \mathrm{d}x_d = \frac{s^{d+1}}{(d+1)!}.\end{aligned}$$

Analogously, for all $s > 0$, we have

$$\begin{aligned}\int_{\mathscr{Q}(\mathbf{0},\alpha s)} x_d d\boldsymbol{x} &= \int_0^{\alpha s} \int_0^{\alpha s} \cdots \int_0^{\alpha s} x_d \otimes_{k=1}^d (\mathrm{d}x_k) \\ &= \int_0^{\alpha s} x_d \int_0^{\alpha s} \cdots \int_0^{\alpha s} \otimes_{k=1}^d (\mathrm{d}x_k) = (\alpha s)^{d-1} \int_0^{\alpha s} x_d \mathrm{d}x_d = 1/2\,(\alpha s)^{d+1}.\end{aligned}$$

We now compute the V_H-measures of a hypercube and a simplex in the basic case in which the distribution H has a linear density, i.e. $v_H(\boldsymbol{b}+\boldsymbol{x}) = a + \sum_{k=1}^d c_k x_k$, for $\boldsymbol{x} \in \mathscr{S}(\mathbf{0}, s) \cup \mathscr{Q}(\mathbf{0}, \alpha s)$. For all $s > 0$, we obtain

$$V_H\left[\mathscr{S}(\boldsymbol{b},s)\right] = a\int_{\mathscr{S}(\mathbf{0},s)} \mathrm{d}\boldsymbol{x} + \sum_{k=1}^{d} c_k \int_{\mathscr{S}(\mathbf{0},s)} x_k \mathrm{d}\boldsymbol{x}$$
$$= a\frac{s^d}{d!} + \frac{s^{d+1}}{(d+1)!}\left(\sum_{k=1}^{d} c_k\right) = \frac{s^d}{d!}\left(a + \frac{s}{d+1}\sum_{k=1}^{d} c_k\right), \tag{5.1}$$
$$V_H\left(\mathscr{Q}(\boldsymbol{b},\alpha s)\right) = a\int_{\mathscr{Q}(\mathbf{0},\alpha s)} \mathrm{d}\boldsymbol{x} + \sum_{k=1}^{d} c_k \int_{\mathscr{Q}(\mathbf{0},\alpha s)} x_k \mathrm{d}\boldsymbol{x}$$
$$= a(\alpha s)^d + \frac{1}{2}\left(\sum_{k=1}^{d} c_k\right)(\alpha s)^{d+1} = (\alpha s)^d\left(a + \frac{1}{2}\alpha s\sum_{k=1}^{d} c_k\right). \tag{5.2}$$

Thus, for a linear density v_H, the ratio $V_H\left[\mathscr{S}(\boldsymbol{b},s)\right] / V_H\left[\mathscr{Q}(\boldsymbol{b},\alpha s)\right]$ can be made independent from the parameters $\boldsymbol{b}, s, a$ and from the c_k's by choosing $\alpha = \alpha^* = \frac{2}{d+1}$, for which we have

$$V_H\left[\mathscr{S}(\boldsymbol{b},s)\right] = \frac{(d+1)^d}{2^d\, d!} V_H\left[\mathscr{Q}(\boldsymbol{b},\alpha^* s)\right]. \tag{5.3}$$

With analogous computations, we obtain the same result for $s < 0$. The following theorem shows that (5.3) analogously holds for any sufficiently smooth density, in the limit as the number n of iterations of the AEP algorithm goes to infinity.

Theorem 5.1 *Assume that H has a twice continuously differentiable density v_H, with all partial derivatives of first and second order bounded by some constant D. Then, we have that*

$$\lim_{n\to+\infty} \max_{k=1,\dots,N^{n-1}} \left| V_H\left[\mathscr{S}(\boldsymbol{b}_n^k, h_n^k)\right] - \frac{(d+1)^d}{2^d\, d!} V_H\left[\mathscr{Q}(\boldsymbol{b}_n^k, \alpha^* h_n^k)\right]\right| = 0. \tag{5.4}$$

Proof. For a given b_n^k, we can use Taylor expansion to find some coefficients a and $c_k, k = 1,\dots,d$, depending on b_n^k, such that

$$v_H(\boldsymbol{b}_n^k + \boldsymbol{x}) = a + \sum_{k=1}^{d} c_k x_k + \sum_{|\beta|=2} R_\beta(\boldsymbol{x})\boldsymbol{x}^\beta \text{ for all } \boldsymbol{x} \in \mathscr{B}(\boldsymbol{b}_n^k), \tag{5.5}$$

where $\mathscr{B}(\boldsymbol{b}_n^k)$ is a ball in $\mathbb{R}^d$ centered at $\boldsymbol{b}_n^k$ such that $\mathscr{B}(\boldsymbol{b}_n^k) \supset \mathscr{S}(\boldsymbol{b}_n^k, h_n^k) \cup \mathscr{Q}(\boldsymbol{b}_n^k, \alpha^* h_n^k)$. Note that in equation (5.5) we used multi-index notation to indicate that the sum in the last equation extends over multi-indices $\beta \in \mathbb{N}^d$. Using the assumption on the partial derivatives of v_H, the remainder term $R_\beta(\boldsymbol{x})$ satisfies the inequality

$$\left|R_\beta(\boldsymbol{x})\right| \le \sup_{\boldsymbol{x}\in\mathscr{B}(\boldsymbol{b}_n^k)} \left|\frac{1}{\beta!}\frac{\partial^\beta v_H(\boldsymbol{x})}{\partial \boldsymbol{x}^\beta}\right| \le D, \tag{5.6}$$

for all β with $\left|\beta\right| = 2$. Using (5.5), and recalling the expressions (5.1) and (5.2) for a linear density and a positive h_n^k, we calculate that

$$\left| V_H\left[\mathscr{S}(\boldsymbol{b}_n^k, h_n^k)\right] - \frac{(d+1)^d}{2^d\, d!} V_H\left[\mathscr{Q}(\boldsymbol{b}_n^k, \alpha h_n^k)\right]\right| =$$
$$\left| \frac{\left(h_n^k\right)^d}{d!}\left(a + \frac{h_n^k}{d+1}\sum_{k=1}^{d} c_k\right) + \int_{\mathscr{S}(\mathbf{0},h_n^k)} \sum_{|\beta|=2} R_\beta(\boldsymbol{x})\boldsymbol{x}^\beta d\boldsymbol{x}\right.$$
$$\left. - \frac{(d+1)^d}{2^d\, d!}\left(\left(\alpha h_n^k\right)^d \left(a + \frac{1}{2}\alpha h_n^k \sum_{k=1}^{d} c_k\right) + \int_{\mathscr{Q}(\mathbf{0},\alpha h_n^k)} \sum_{|\beta|=2} R_\beta(\boldsymbol{x})\boldsymbol{x}^\beta d\boldsymbol{x}\right)\right|.$$

Choosing $\alpha = \alpha^*$, the previous expression simplifies to

$$\begin{aligned}
&\left| V_H\left[\mathscr{S}(\boldsymbol{b}_n^k, h_n^k)\right] - \frac{(d+1)^d}{2^d\, d!} V_H\left[\mathscr{Q}(\boldsymbol{b}_n^k, \alpha^* h_n^k)\right]\right| \\
&= \left| \int_{\mathscr{S}(\boldsymbol{0}, h_n^k)} \sum_{|\beta|=2} R_\beta(\boldsymbol{x})\boldsymbol{x}^\beta d\boldsymbol{x} - \frac{(d+1)^d}{2^d\, d!} \int_{\mathscr{Q}(\boldsymbol{0}, \alpha^* h_n^k)} \sum_{|\beta|=2} R_\beta(\boldsymbol{x})\boldsymbol{x}^\beta d\boldsymbol{x} \right| \\
&\leq \left| \sum_{|\beta|=2} \int_{\mathscr{S}(\boldsymbol{0}, h_n^k)} R_\beta(\boldsymbol{x})\boldsymbol{x}^\beta d\boldsymbol{x} \right| + \frac{(d+1)^d}{2^d\, d!} \left| \sum_{|\beta|=2} \int_{\mathscr{Q}(\boldsymbol{0}, \alpha^* h_n^k)} R_\beta(\boldsymbol{x})\boldsymbol{x}^\beta d\boldsymbol{x} \right| \\
&\leq D\left(\left| \sum_{|\beta|=2} \int_{\mathscr{S}(\boldsymbol{0}, h_n^k)} \boldsymbol{x}^\beta d\boldsymbol{x} \right| + \frac{(d+1)^d}{2^d\, d!} \left| \sum_{|\beta|=2} \int_{\mathscr{Q}(\boldsymbol{0}, \alpha^* h_n^k)} \boldsymbol{x}^\beta d\boldsymbol{x} \right| \right),
\end{aligned}$$

where the last inequality follows from (5.6). Using that

$$\begin{aligned}
\sum_{|\beta|=2} \int_{\mathscr{S}(\boldsymbol{0}, s)} \boldsymbol{x}^\beta d\boldsymbol{x} &= \sum_{i=1}^{d} \int_{\mathscr{S}(\boldsymbol{0}, s)} x_i^2 d\boldsymbol{x} + 2 \sum_{1\leq i<j\leq d} \int_{\mathscr{S}(\boldsymbol{0}, s)} x_i\, x_j d\boldsymbol{x} \\
&= \frac{2ds^{d+2}}{(d+2)!} + \frac{2d(d-1)s^{d+2}}{(d+2)!} = \frac{2d^2 s^{d+2}}{(d+2)!},
\end{aligned}$$

and

$$\begin{aligned}
\sum_{|\beta|=2} \int_{\mathscr{Q}(\boldsymbol{0}, \alpha s)} \boldsymbol{x}^\beta d\boldsymbol{x} &= \sum_{i=1}^{d} \int_{\mathscr{Q}(\boldsymbol{0}, \alpha s)} x_i^2 d\boldsymbol{x} + 2 \sum_{1\leq i<j\leq d} \int_{\mathscr{Q}(\boldsymbol{0}, \alpha s)} x_i\, x_j d\boldsymbol{x} \\
&= \frac{d\,(\alpha s)^{d+2}}{3} + \frac{2d(d-1)\,(\alpha s)^{d+2}}{4} = \frac{d(3d-1)\,(\alpha s)^{d+2}}{6},
\end{aligned}$$

we finally obtain

$$\left| V_H\left[\mathscr{S}(\boldsymbol{b}_n^k, h_n^k)\right] - \frac{(d+1)^d}{2^d\, d!} V_H\left[\mathscr{Q}(\boldsymbol{b}_n^k, \alpha^* h_n^k)\right]\right| \leq A\left|h_n^k\right|^{d+2}, \tag{5.7}$$

where A is a positive constant depending only on the dimension d and the distribution H. Note that in (5.7) we write h_n^k in absolute value in order to consider the completely analogous case in which h_n^k is negative. Recalling from (3.9) that $\max_{k=1,\ldots,N^{n-1}} \left|h_n^k\right| = \gamma^{n-1}s$, for $\gamma = |1 - d\alpha^*| = \frac{d-1}{d+1} < 1$, yields the theorem. □

Equation (5.4) gives a local estimator of the mass of the simplex $\mathscr{S}(\boldsymbol{b}_n^k, h_n^k)$ in terms of the volume of the corresponding hypercube $\mathscr{Q}(\boldsymbol{b}_n^k, h_n^k)$, which is straightforward to compute:

$$V_H\left[\mathscr{S}(\boldsymbol{b}_n^k, h_n^k)\right] \approx \frac{(d+1)^d}{2^d d!} V_H\left[\mathscr{Q}\left(\boldsymbol{b}_n^k, \frac{2h_n^k}{d+1}\right)\right]. \tag{5.8}$$

In the case that the density v_H is sufficiently smooth, it is then possible, after a number of iterations of AEP, to estimate the right side of (3.5) by using the approximation (5.8). This procedure defines the estimator $P_n^*(s)$ as

$$P_n^*(s) = P_{n-1}(s) + \frac{(d+1)^d}{2^d\, d!} \sum_{k=1}^{N^{n-1}} s_n^k V_H\left[\mathscr{Q}_n^k\right]. \tag{5.9}$$

In what follows, the use of $P_n^*(s)$ as an approximation of $V_H[\mathscr{S}(\boldsymbol{0}, s)]$ will be referred to as the *extrapolation* technique. The following theorem shows that $P_n^*(s)$ converges to $V_H\,[\mathscr{S}(\boldsymbol{0}, s)]$ faster, and in higher dimensions, than $P_n(s)$.

Theorem 5.2 *Under the assumptions of Theorem 5.1, and for $d \leq 8$, we have that*

$$\lim_{n\to+\infty} P_n^*(s) = V_H\left[\mathscr{S}(\mathbf{0},s)\right].$$

Proof. Using (3.5) and (5.7) in the definition (5.9) of $P_n^*(s)$, we obtain

$$\begin{aligned}
E^*(n) &= \left|V_H\left[\mathscr{S}(\mathbf{0},s)\right] - P_n^*(s)\right| \\
&= \left|V_H\left[\mathscr{S}(\mathbf{0},s)\right] - P_{n-1}(s) - \frac{(d+1)^d}{2^d d!}\sum_{k=1}^{N^{n-1}} s_n^k V_H\left[\mathscr{Q}_n^k\right]\right| \\
&= \left|\sum_{k=1}^{N^{n-1}} s_n^k V_H\left[\mathscr{S}_n^k\right] - \frac{(d+1)^d}{2^d d!}\sum_{k=1}^{N^{n-1}} s_n^k V_H\left[\mathscr{Q}_n^k\right]\right| \\
&\leq \sum_{k=1}^{N^{n-1}}\left|V_H\left[\mathscr{S}_n^k\right] - \frac{(d+1)^d}{2^d d!} V_H\left[\mathscr{Q}_n^k\right]\right| \leq A\sum_{k=1}^{N^{n-1}}\left|h_n^k\right|^{d+2} = Ae_{n-1}^*,
\end{aligned} \tag{5.10}$$

where, for the positive sequence $e_n^* = \sum_{k=1}^{N^n}\left|h_{n+1}^k\right|^{d+2}$, we have that

$$\begin{aligned}
\frac{e_n^*}{e_{n-1}^*} &= \frac{\sum_{k=1}^{N^{n-1}}\sum_{j=1}^{N}\left|h_{n+1}^{Nk-N+j}\right|^{d+2}}{\sum_{k=1}^{N^{n-1}}\left|h_n^k\right|^{d+2}} = \frac{\sum_{k=1}^{N^{n-1}}\sum_{j=1}^{d}\binom{d}{j}\left|1-j\alpha^*\right|^{d+2}\left|h_n^k\right|^{d+2}}{\sum_{k=1}^{N^{n-1}}\left|h_n^k\right|^{d+2}} \\
&= \frac{\sum_{k=1}^{N^{n-1}}\left|h_n^k\right|^{d+2}\sum_{j=1}^{d}\binom{d}{j}\left|1-j\alpha^*\right|^{d+2}}{\sum_{k=1}^{N^{n-1}}\left|h_n^k\right|^{d+2}} = \sum_{j=1}^{d}\binom{d}{j}\left|1-j\alpha^*\right|^{d+2}.
\end{aligned}$$

The theorem follows by noting that the factor $f_*(d)$, defined as

$$f_*(d) = \sum_{j=1}^{d}\binom{d}{j}\left|1-j\alpha^*\right|^{d+2}, \tag{5.11}$$

is lower than 1 for $d \leq 8$; see Table 2. In these dimensions, e_n^*, and hence $E^*(n)$, converge to zero. □

We remark that, due to Theorem 3.3, Theorem 5.2 remains valid also in the case H satisfies the extra smoothness conditions on its first and second derivatives only in a neighborhood of Γ_s. Moreover, under the assumptions of Theorem 5.1, it is possible to calculate an upper bound for the error $E^*(n)$ as a function of the number of evaluations performed by AEP. Indeed, (5.10) can be rewritten as

$$E^*(n) \leq Af_*(d)^n. \tag{5.12}$$

We now denote by $M(n)$ the total number of evaluations of the joint distribution H performed by AEP after the n-th iteration. $M(n)$ (as well as the computational time used) is proportional to the number of simplexes $f_S(d)^{n-1}$ passed to the n-th iteration. For all $n \geq 2$, we have that

$$M(n) = \sum_{k=0}^{n-1} 2^d f_S(d)^k = \frac{2^d}{f_S(d)-1}\left(f_S(d)^n - 1\right) \geq \left(\frac{2^d}{f_S(d)-1} - 1\right) f_S(d)^n = Bf_S(d)^n. \tag{5.13}$$

B is a positive constant depending only on the dimension d. Combining (5.12) and (5.13) gives

$$E(n) \leq A\left(\frac{M(n)}{B}\right)^{\frac{\ln f_*(d)}{\ln f_S(d)}}. \tag{5.14}$$

Then (5.14) provides an upper bound on the AEP approximation error $E^*(n)$ as a function of the number of evaluations performed. The polynomial rate of convergence $\frac{\ln f_*(d)}{\ln f_S(d)}$ of this bound depends only on the dimensionality d. In Table 2 we calculate this bound for dimensions $d \leq 8$. These numbers can be useful in order to compare the efficiency of AEP with respect to other algorithms, such as Monte Carlo methods (see Section 7 and Table 11).

d	2	3	4	5	6	7	8	9
$f_*(d)$	0.0370	0.1250	0.2339	0.3580	0.4982	0.6556	0.8314	>1
$f_S(d)$	3	4	15	21	63	92	255	385
$\frac{\ln f_*(d)}{\ln f_S(d)}$	-3	-1.5	-0.54	-0.34	-0.17	-0.09	-0.033	na

Table 2
Number $f_S(d)$ of new simplexes produced at each iteration, extrapolation error ratio $f_*(d)$ as defined in (5.11), and convergence rates of the AEP extrapolation error as a function of the number of evaluations performed by the algorithm. For $d = 9$ convergence of AEP is not assured (na).

6. Applications

In this section we test the AEP algorithm on some risk vectors $(X_1, \dots, X_d)$ of financial and actuarial interest. For illustrative reasons, we will provide the distribution H in terms of the marginal distributions F_{X_i} and copula C of the vector $(X_1, \dots, X_d)$. For the theory of copulas, we refer the reader to Nelsen (2006).

In Table 3 we consider a two-dimensional portfolio ($d = 2$) with Pareto marginals, i.e.

$$F_{X_i}(x) = \mathbb{P}[X_i \leq x] = 1 - (1+x)^{-\theta_i}, \; x \geq 0, \; i = 1, 2,$$

with tail parameters $\theta_1 = 0.9, \theta_2 = 1.8$. We couple these Pareto marginals via a Clayton copula $C = C^{Cl}_\delta$ with

$$C^{Cl}_\delta(u_1, \dots, u_d) = \left(u_1^{-\delta} + u_2^{-\delta} + \cdots + u_d^{-\delta} - d + 1\right)^{-1/\delta}, \; u_k \in [0,1], k \in D.$$

The parameter δ is set to 1.2. For the above described portfolio, we compute the approximation $P_n(s)$ (see (3.3)), at some given thresholds s and for different numbers of iterations n of the algorithm. The thresholds s are chosen in order to have estimates in the center as well as in the (heavy) tail of the distribution. For each n, we provide the computational time needed to obtain the estimate on a Apple MacBook (2.4 GHz Intel Core 2 Duo, 2 GB RAM). We also provide the estimates obtained by using the estimator $P^*_n(s)$ as defined in (5.9). For all iterations n and thresholds s, in Table 3, we provide the differences $P_n(s) - P_{14}(s)$ or $P^*_n(s) - P_{14}(s)$. This has been done in order to show the speed of convergence of the algorithm and the increase in accuracy due to extrapolation. For a two dimensional vector, we see that all iterations after the seventh leave the first eight decimal digits of the probability estimate unaltered, and this for all the thresholds. The estimate $P_7(s)$, obtained in 1/100-th of a second, could be already considered reasonably accurate. We also note that extrapolation allows to increase the accuracy of the estimates by two decimal digits on average, and this without increasing computational time.

In Tables 4($d = 3$) to 6($d = 5$) we perform the same analysis for different Clayton-Pareto models in which we progressively increase the number of random variables involved. In order to keep the computational time needed for a single estimate reasonable we were forced to take smaller number of iterations for the reference value, i.e. we choose $n = 12$ for $d = 3$, $n = 7$ for $d = 4$ and $n = 6$ for $d = 5$. AEP shows good convergence results for all dimensions d and thresholds s under study. In higher dimensions d, the extrapolation technique still seems to provide relevant extra accuracy. Memory constraints made estimates for $d = 6$ prohibitive. For dimensions $2 \leq d \leq 5$, Figure 5 shows that the average computational time needed by AEP to provide a single estimate increases exponentially in the number of iterations n.

Note that Tables 3-6 give information only about the convergence of the algorithm to a certain value, but do not say anything about the correctness of the limit, since we do not have analytical methods to compute $V_H[\mathscr{S}(\mathbf{0},s)]$ when the vector $(X_1,\ldots,X_d)$ has a general dependence structure C.

In practice, it possible to test the accuracy of AEP in particular cases when the X_i are independent or comonotonic. Some test cases are analyzed in Tables 7($d = 2$) to 9($d = 4$), where we still assume to have Pareto marginals, but coupled by a Gumbel copula $C = C_\gamma^{Gu}$ in which the parameter $\gamma \geq 1$ is allowed to vary. Formally, we have

$$C_\gamma^{Gu}(u_1,\ldots,u_d) = \exp\Big(-\big[(-\ln u_1)^\gamma + (-\ln u_2)^\gamma + \ldots (-\ln u_d)^\gamma\big]^{1/\gamma}\Big), \;\; u_k \in (0,1], k = 1,\ldots,d.$$

In the above mentioned tables, the multivariate model varies from independence ($\gamma = 1$), to comonotonicity ($\gamma = +\infty$). In these two extreme (with respect to the dependence parameter γ) cases we compare the analytical values for $V_H[\mathscr{S}(\mathbf{0},s)]$ with their AEP estimates. Tables 3-6 show that the extrapolated estimator $P_n^*(s)$ provides accurate estimates within a very reasonable computational time. A comparison with alternative methods is discussed in Section 7.

The possibility of computing the value $V_H[\mathscr{S}(\mathbf{0},s)]$ independently from AEP also allows us to test more specifically the effect of extrapolation. For this, we consider two- and three-dimensional vectors of independent Pareto marginals. Figure 4 shows the increase of accuracy due to extrapolation. Therefore, under a smooth model for H (see Theorem 5.1), the extrapolated estimator $P_n^*(s)$ is to be preferred to $P_n(s)$.

Of course, the AEP algorithm can be used to find estimates for the quantile function, i.e. for the inverse of the distribution of the sum S_d. Such quantiles are especially useful in finance and insurance, where they are generally referred to as Value-at-Risk (VaR) or return periods. In Table 10 we calculate, by numerical inversion, VaR at different quantile levels α for two different three-dimensional portfolios of risks. In order to calculate VaR values, we use root-finding algorithms like the bisection method.

We finally remark that the choice of the copula families (Clayton, Gumbel) and marginal distributions used in this section is purely illustrative and does not affect in any way the functioning of AEP algorithm. The same performances were reached for vectors showing negative dependence, as in the case of d Pareto marginals coupled by a Frank copula with negative parameter.

7. A comparison with Monte Carlo, Quasi-Monte Carlo and quadrature methods

For the estimation of $V_H[\mathscr{S}(\mathbf{0},s)]$, the main competitors of the AEP algorithm are probably Monte Carlo and quasi-Monte Carlo methods. Given M points $x_1,\ldots,x_M$ in $\mathscr{S}(\mathbf{0},s)$, it is possible to approximate $V_H[\mathscr{S}(\mathbf{0},s)]$ by the average of the density function v_H evaluated at those points, i.e.

$$V_H[\mathscr{S}(\mathbf{0},s)] = \int_{\mathscr{S}(\mathbf{0},s)} dH(\boldsymbol{x}) \simeq \frac{1}{M}\sum_{i=1}^{M} v_H(x_i). \tag{7.1}$$

If the x_i's are chosen to be (pseudo)randomly distributed, this is the *Monte Carlo* (MC) method. If the x_i's are chosen as elements of a low-discrepancy sequence, this is the *Quasi-Monte Carlo*

	$n = 14$	$n = 7$	$n = 7^*$	$n = 10$	$n = 10^*$	$n = 13$	$n = 13^*$
	(ref. value, 4.87 sec.)	(0.01 sec.)	(0.01 sec.)	(0.06 sec.)	(0.06 sec.)	(1.61 sec.)	(1.61 sec.)
$s = 1$	0.315835041363400	-4.46e-09	-1.45e-11	-6.12e-12	3.33e-15	2.05e-15	1.24e-14
$s = 100$	0.983690398912818	-3.10e-10	1.83e-09	-1.30e-12	-1.29e-14	2.88e-14	2.94e-14
$s = 10000$	0.999748719228269	-6.62e-08	-4.13e-08	-5.28e-12	6.49e-11	7.83e-14	8.08e-14
$s = 1000000$	0.999996018907898	-1.63e-09	-1.22e-09	-5.34e-11	-3.83e-11	-1.81e-13	1.06e-13

Table 3
Values for $P_n(s)$ and $P_n^*(s)$ (starred columns) for the sum of two Pareto distributions with parameters $\theta_1 = 0.9$ and $\theta_2 = 1.8$, coupled by a Clayton copula with parameter $\delta = 1.2$. For all $n < 14$, we give the difference from the reference value $P_{14}(s)$.

	$n = 12$	$n = 7$	$n = 7^*$	$n = 9$	$n = 9^*$	$n = 11$	$n = 11^*$
	(ref. value, 26.65 sec.)	(0.02 sec.)	(0.02 sec.)	(0.41 sec.)	(0.41 sec.)	(6.65 sec.)	(6.65 sec.)
$s = 1$	0.190859309168541	-2.28e-06	8.80e-07	-8.48e-08	3.36e-08	2.63e-09	1.84e-09
$s = 100$	0.983659546331932	-1.76e-05	1.14e-06	-6.51e-07	3.04e-07	-1.84e-08	1.45e-08
$s = 10000$	0.999748691148512	-1.70e-06	-1.10e-06	-3.69e-07	-2.21e-07	-4.66e-08	-1.18e-08
$s = 1000000$	0.999996018044029	-2.73e-08	-1.78e-08	-6.14e-09	-3.79e-09	-8.78e-10	-2.94e-10

Table 4
The same as Table 3 for the sum of three Pareto distributions with parameters $\theta_1 = 0.9$, $\theta_2 = 1.8$, $\theta_3 = 2.6$, coupled by a Clayton copula with parameter $\delta = 0.4$.

	$n = 7$	$n = 4$	$n = 4^*$	$n = 5$	$n = 5^*$	$n = 6$	$n = 6^*$
	(ref. value, 107.70 sec.)	(0.03 sec.)	(0.03 sec.)	(0.47 sec.)	(0.47 sec.)	(7.15 sec.)	(7.15 sec.)
$s = 10$	0.833447516734442	-6.31e-03	9.42e-05	-2.21e-03	3.71e-04	-6.04e-04	4.00e-04
$s = 100$	0.983412214152579	-1.61e-03	-4.95e-04	-7.14e-04	-1.54e-04	-2.45e-04	5.01e-05
$s = 1000$	0.997950264030106	-2.14e-04	-7.37e-05	-9.91e-05	-2.70e-05	-3.60e-05	3.68e-06
$s = 10000$	0.999742266243751	-2.69e-05	-9.30e-06	-1.25e-05	-3.42e-06	-4.54e-06	4.52e-07

Table 5
The same as Table 3 for the sum of four Pareto distributions with parameters $\theta_1 = 0.9$, $\theta_2 = 1.8$, $\theta_3 = 2.6$, $\theta_4 = 3.3$, coupled by a Clayton copula with parameter $\delta = 0.2$.

	$n = 6$	$n = 3$	$n = 3^*$	$n = 4$	$n = 4^*$	$n = 5$	$n = 5^*$
	(ref. value, 92.91 sec.)	(0.01 sec.)	(0.01 sec.)	(0.20 sec.)	(0.20 sec.)	(4.37 sec.)	(4.37 sec.)
$s = 10$	0.824132635126808	-3.12e-02	3.89e-03	-1.55e-02	5.66e-04	-7.77e-03	1.46e-04
$s = 100$	0.983253494805448	-5.30e-03	5.07e-05	-2.86e-03	-3.57e-04	-1.54e-03	-1.90e-04
$s = 1000$	0.997930730055234	-6.72e-04	-5.23e-06	-3.66e-04	-5.29e-05	-1.99e-04	-2.83e-05
$s = 10000$	0.999739803851201	-8.45e-05	-7.22e-07	-4.61e-05	-6.67e-06	-2.51e-05	-3.57e-06

Table 6
The same as Table 3 for the sum of five Pareto distributions with parameters $\theta_1 = 0.9$, $\theta_2 = 1.8$, $\theta_3 = 2.6$, $\theta_4 = 3.3$, $\theta_5 = 4$, coupled by a Clayton copula with parameter $\delta = 0.3$.

	$\gamma = 1$ (exact)	$\gamma = 1$	$\gamma = 1.25$	$\gamma = 1.5$	$\gamma = 1.75$	$\gamma = +\infty$	$\gamma = +\infty$ (exact)
$s = 1$	0.2862004	0.2862004	0.3280000	0.3527174	0.3682522	0.4108029	0.4108027
$s = 100$	0.9898913	0.9898913	0.9895957	0.9894472	0.9893640	0.9891761	0.9891761
$s = 1000$	0.9989990	0.9989990	0.9989857	0.9989798	0.9989766	0.9989700	0.9989700
$s = 10000$	0.9999000	0.9999000	0.9998995	0.9998993	0.9998992	0.9998990	0.9998990

Table 7
Values for $P_n^*(s)$ for the sum of two Pareto distributions with parameters $\theta_i = i, i = 1,2$, coupled by a Gumbel copula with parameter γ. The values in the first and in the last column are calculated analytically. The computational time for each estimate in this table is 0.53 sec. with $n = 12$.

	$\gamma = 1$ (exact)	$\gamma = 1$	$\gamma = 1.25$	$\gamma = 1.5$	$\gamma = 1.75$	$\gamma = +\infty$	$\gamma = +\infty$ (exact)
$s = 1$	0.1709337	0.1709337	0.2348582	0.2743918	0.2994054	0.3667285	0.3666755
$s = 100$	0.9898380	0.9898380	0.9893953	0.9891754	0.9890526	0.9887811	0.9887760
$s = 1000$	0.9989985	0.9989985	0.9989812	0.9989734	0.9989692	0.9989604	0.9989606
$s = 10000$	0.9999000	0.9999000	0.9998994	0.9998992	0.9998991	0.9998988	0.9998988

Table 8
The same as Table 7 for the sum of three Pareto distributions with parameters $\theta_i = i, i = 1,2,3$, coupled by a Gumbel copula with parameter γ. The computational time for each estimate in this table is 6.65 sec. with $n = 11$.

	$\gamma = 1$ (exact)	$\gamma = 1$	$\gamma = 1.25$	$\gamma = 1.5$	$\gamma = 1.75$	$\gamma = +\infty$	$\gamma = +\infty$ (exact)
$s = 1$	0.1040880	0.1040713	0.1762643	0.2244387	0.2555301	0.3387648	0.3390320
$s = 100$	0.9898032	0.9896608	0.9892592	0.9890502	0.9889268	0.9886415	0.9885287
$s = 1000$	0.9989981	0.9989732	0.9989652	0.9989616	0.9989595	0.9989743	0.9989558
$s = 10000$	0.9999000	0.9998973	0.9998973	0.9998973	0.9998973	0.9998973	0.9998987

Table 9
The same as Table 7 for the sum of four Pareto distributions with parameters $\theta_i = i, i = 1,2,3,4$ coupled by a Gumbel copula with parameter γ. The computational time for each estimate in this table is 7.15 sec. with $n = 6$.

(a)				**(b)**			
α	VaR_α	α	VaR_α	α	VaR_α	α	VaR_α
0.9	24.76	0.9999	3394.78	0.9	32.87	0.9999	112442.31
0.99	137.67	0.99999	17962.78	0.99	445.36	0.99999	1903698.40
0.999	700.20	0.999999	108190.96	0.999	6864.58	0.999999	32889360.00

Table 10
Value-at-Risk for: (a) a three dimensional portfolio with marginals F_1 = Exp(0.2), F_2 = Logn($\mu = -0.5, \sigma^2 = 9/2$), F_3 = Pareto(1.2) and a Gumbel copula with $\gamma = 1.3$. (b) a three dimensional portfolio with Pareto marginals with parameters $\theta_1 = 0.8$, $\theta_2 = 1$, $\theta_3 = 2$ and a Clayton copula with $\delta = 0.4$. The computation of all VaR estimates needs approximately 49 sec. with $n = 10$.

(QMC) method. A *low-discrepancy* sequence is a totally determinist sequence of vectors that generates representative samples from a uniform distribution on a given set. With respect to Monte Carlo methods, the advantage of using quasi-random sequences is that points cannot cluster coincidentally on some region of the set. Randomization of a low-discrepancy sequence however often improves performance; see L'Ecuyer and Lemieux (2000) on this.

Over the recent years, various methods and algorithms have been developed in order to reduce the variance of MC and QMC estimators and obtain probabilities of (rare) events with

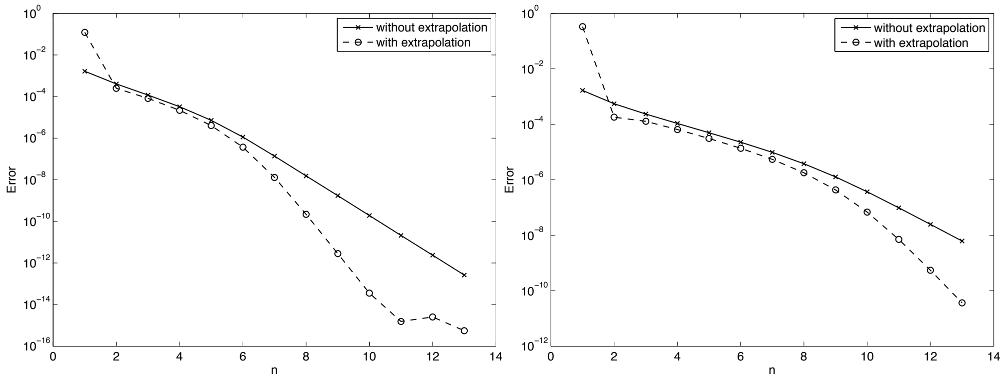

Fig. 4. Error committed by the AEP algorithm with and without the use of extrapolation technique for two test portfolios: two (left) and three (right) independent Pareto marginals with parameters $\theta_i = i, i = 1,2,3$.

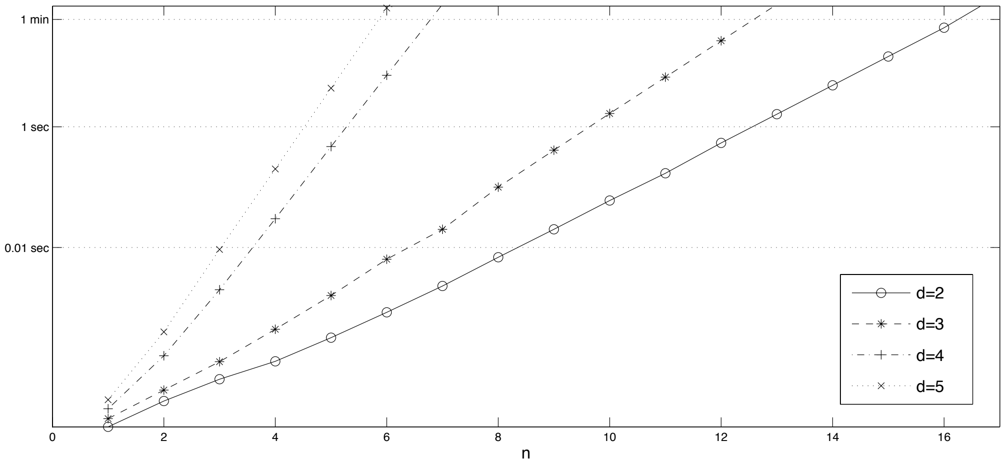

Fig. 5. AEP computational time (on a log-scale), as a function of the number of iterations n, for dimensions $2 \leq d \leq 5$.

a reasonable precision and effort. For details on the theory of *rare event simulation* within MC methods, we refer the reader to the monographs Asmussen and Glynn (2007), Glasserman (2004), and McLeish (2005, 2008). For an introduction to Quasi-Monte Carlo methods and recent improvements, we refer for instance to Niederreiter (1992) and L'Ecuyer and Lemieux (2000). A comprehensive overview of both methods is given in Weinzierl (2000).

Using Central Limit Theorem arguments, it is possible to show that traditional MC, using (pseudo) random numbers, has a convergence rate of $O(M^{-1/2})$, and this independently of the number of dimensions d. QMC can be much faster than MC with errors approaching $O(M^{-1})$ in optimal cases (see Morokoff (1998)), but the worst theoretic rate of convergence decreases with the dimension d as $O((\log M)^d M^{-1})$; see Niederreiter (1992). In applications to finance and insurance, it is more common to get results closer to the best rate of convergence if the density v_H is smooth, i.e. having a Lipschitz-continuous second derivative. In this case, it is possible to show that the convergence rate is at least $O((\log M)^d M^{-3/2})$; see Caflisch et al. (1997). In Table 11 we compare convergence rates of MC and QMC methods with respect to the AEP rates (depending on d) as provided in Section 5. We thus expect a well-designed

QMC algorithm to asymptotically perform better than AEP under a smooth probability model and for dimensions $d \geq 4$. In this table, we considered only the dimensions $d \leq 5$ which are manageable from a computational viewpoint; see Section 6 on this.

d	2	3	4	5
AEP (upper bound)	M^{-3}	$M^{-1.5}$	$M^{-0.54}$	$M^{-0.34}$
MC	$M^{-0.5}$	$M^{-0.5}$	$M^{-0.5}$	$M^{-0.5}$
QMC (best)	M^{-1}	M^{-1}	M^{-1}	M^{-1}
QMC (worst)	$M^{-1}(\log M)^2$	$M^{-1}(\log M)^3$	$M^{-1}(\log M)^4$	$M^{-1}(\log M)^5$

Table 11
Asymptotic convergence rates of the AEP, standard MC and QMC methods.

Don McLeish kindly adapted an algorithm using a randomized Korobov low discrepancy sequence to the portfolio leading to Table 3. The parameters for the sequence are those recommended in Gill and Lemieux (2007). The standard errors (s.e.) are obtained by independently randomizing ten (part (a) of the table) and fifty (part (b) of the table) sequences with 1 million terms each, corresponding to M =1e07 (a) and M =5e07 (b). The average CPU times are of course on a different machine (IBM Thinkpad 2.5 GHz Intel Core 2 Dual, 4 GB RAM). In Table 12, we provide the comparison between QMC and AEP extrapolated estimates. The results seem to be coherent with Table 11 above. For the same precision, AEP is much faster than QMC in the two-dimensional example, and slightly slower for $d = 4$. Recall that, in higher dimensions, programming a randomized Korobov rule is much more demanding than using AEP.

What is important to stress here is that in MC and randomized QMC methods similar to the one applied in Table 12, the final estimates contain a source of randomness. Contrary to this, the AEP algorithm is deterministic, being solely based on geometrical properties of a certain domain. Moreover, the accuracy of MC and QMC methods is generally lost for problems in which the density v_H is not smooth or cannot be given in closed form, and comes at the price of an adaptation of the sampling algorithm to the specific example under study. Recall that the AEP algorithm does not require the density of the distribution H in analytic form nor has to assume its smoothness overall its domain. Finally, the precision of MC methods depends on the threshold s at which $V_H[\mathscr{S}(\mathbf{0}, s)]$ is evaluated: estimates in the (far) tail of the distribution will be less accurate.

(a)	AEP estimate	QMC estimate	QMC s.e.	**(b)**	AEP estimate	QMC estimate	QMC s.e.
s	($n = 14$, 4.87 sec.)	(M =1e07, 6.6 sec.)		s	($n = 7$, 107.70 sec.)	(M =5e07, 95 sec.)	
10^0	0.315835041363413	0.3158345	2.7e-06	10^1	0.833826902853978	0.83380176	3.6e-06
10^2	0.983690398912470	0.98369106	1.0e-06	10^2	0.983565803484355	0.98362452	9.0e-07
10^4	0.999748719228038	0.99974872	1.5e-07	10^3	0.997972831330699	0.997997715	2.3e-07
10^6	0.999996018907752	0.999996	4.0e-08	10^4	0.999745113409911	0.999748680	5.0e-08

Table 12
AEP and QMC (using Korobov sequence) estimates for $V_H[\mathscr{S}(\mathbf{0}, s)]$ for the sum of: (a) two Pareto distributions with parameters $\theta_1 = 0.9$ and $\theta_2 = 1.8$, coupled by a Clayton copula with parameter $\delta = 1.2$; (b) four Pareto distributions with parameters $\theta_1 = 0.9$, $\theta_2 = 1.8$, $\theta_3 = 2.6$, $\theta_4 = 3.3$, coupled by a Clayton copula with parameter $\delta = 0.2$. Computational times are also provided.

The re-tailoring of the rule to be iterated, from example to example, is common also to other numerical techniques for the estimation of $V_H[\mathscr{S}(\mathbf{0},s)]$ such as *quadrature methods*; see Davis and Rabinowitz (1984) and Press et al. (2007) for a review. However, in the computation of multi-dimensional integrals as in (7.1), numerical quadrature rules are typically less efficient than MC and QMC since they must be applied iteratively.

We are of course aware that a well-designed quadrature rule or a specific quasi-random sequence might perform better than AEP in a specific example, both with respect to accuracy as well as computational effort. However, AEP provides very accurate estimates of the distribution of sums up to five dimensions in a reasonable time and this *without* the need to adapt to the probabilistic model under study. AEP can handle in a uniform way any joint distribution H, possibly in the form of its copula and marginals distributions. Because of its easiness to use and the very weak assumptions upon which it is based, AEP offers a competitive tool for the computation of the distribution function of a sum of up to five random variables. A web-based, user-friendly version has been programmed.

8. Final remarks

In this paper, we introduce the AEP algorithm in order to compute numerically the distribution function of the sum of d random variables $X_1,\dots,X_d$ with given joint distribution H. The algorithm is mainly based on two assumptions: the random variables X_i are bounded from below, and the distribution H has a bounded density in a neighborhood of the curve Γ_s defined in (3.7). Under this last assumption, the sum S_d has to be continuous at the threshold s where the distribution is calculated, i.e. $\mathbb{P}[S_d = s] = 0$. When $V_H[\Gamma_s] > 0$ instead, the algorithm may fail to converge. As an example, take two random variables X_1 and X_2 with $\mathbb{P}[X_1 = 1/2] = \mathbb{P}[X_2 = 1/2] = 1$. Then, $V_H[\mathscr{S}(\mathbf{0},1)] = 1$ but the sequence $P_n(1)$ alternates between 0 and 1. Similar examples for arbitrary dimension d can easily be constructed.

If H has at least a bounded density near Γ_s, then the convergence of the sequence $P_n(s)$ to the value $V_H[\mathscr{S}(\mathbf{0},s)]$ is guaranteed. As already remarked, the speed of convergence may vary depending on the probability mass of a neighborhood of Γ_s. Tools to increase the efficiency of the algorithm are therefore much needed in these latter cases.

Open problems

The AEP algorithm has been shown to converge when $d \le 5$ if the joint distribution H of the vector $(X_1,\dots,X_d)$ has a bounded density v_H. Under some extra smoothness assumptions for v_H, convergence holds when $d \le 8$. All these conditions can be weakened to hold only in a neighborhood of the curve Γ_s, and they are always satisfied in relevant financial/actuarial applications.

Of course, the algorithm may converge also in higher dimensions, but we were not able to give a proof of convergence in arbitrary dimensions. The main issue here is represented by the fact the sequences $P_n(s)$ and $P_n^*(s)$ are not monotonic, and this because the s_n^k's, as defined in (3.4), may be positive as well as negative. Recall also that, due to memory constraints, we were not able to run the algorithm for $d > 5$ and sufficiently large n. Moreover, we expect the AEP convergence rates to be better than their upper bounds given in Table 2.

Apart from the study of convergence of AEP in higher dimensions, in future research we will address also an extension of the algorithm to more general aggregating functions $\psi(X_1,\dots,X_d)$, and the study of an adaptive (i.e. depending on H) and more efficient (in terms of new simplexes produced at each iteration) decomposition of the simplexes.

Acknowledgments

The authors are grateful to Don McLeish for providing relevant comments on the paper and the example illustrated in Table 11. Giovanni Puccetti would like to thank RiskLab and the Forschungsinstitut für Mathematik (FIM) of the Department of Mathematics, ETH Zurich, for its financial support and kind hospitality. Philipp Arbenz likes to thank SCOR for financial support towards the final stages of writing this paper.

Appendix A. Proof of (3.1)

Recall that in Section 1 we denoted with $\boldsymbol{i}_0, \ldots, \boldsymbol{i}_N$ all the 2^d vectors in $\{0,1\}^d$, with $\boldsymbol{i}_0 = (0,\ldots,0)$, $\boldsymbol{i}_k = \boldsymbol{e}_k, k = 1,\ldots,d$, and $\boldsymbol{i}_N = \mathbf{1} = (1,\ldots,1)$, where $N = 2^d - 1$. Recall also that $\#\boldsymbol{i}$ denotes the number of 1s in the vector $\boldsymbol{i}$, e.g. $\#\boldsymbol{i}_0 = 0, \#\boldsymbol{i}_N = d$.

Theorem A.1 *For any $\boldsymbol{b} \in \mathbb{R}^d$, $h \in \mathbb{R}$ and $\alpha \in [1/d, 1)$, we have that*

$$V_H[\mathscr{S}(\boldsymbol{b}, h)] = V_H[\mathscr{Q}(\boldsymbol{b}, \alpha h)] + \sum_{j=1}^{N} m^j V_H\left[\mathscr{S}(\boldsymbol{b}^j, h^j)\right],$$

where, for all $j = 1,\ldots,N$,

$$\boldsymbol{b}^j = \boldsymbol{b} + \alpha h \boldsymbol{i}_j, \quad h^j = (1 - \#\boldsymbol{i}_j \alpha)h, \quad m^j = \begin{cases} (-1)^{1+\#\boldsymbol{i}_j} & \text{if } \ \#\boldsymbol{i}_j < 1/\alpha, \\ 0 & \text{if } \ \#\boldsymbol{i}_j = 1/\alpha, \\ (-1)^{d+1-\#\boldsymbol{i}_j} & \text{if } \ \#\boldsymbol{i}_j > 1/\alpha. \end{cases} \tag{A.1}$$

Note that (A.1) is equivalent to (3.1) under the notation introduced in Section 3. In order to prove the above theorem, we need some lemmas. In the following δ_{ij} denotes the Kronecker delta, i.e.

$$\delta_{ij} = \begin{cases} 0 & \text{if } i \neq j, \\ 1 & \text{if } i = j. \end{cases}$$

Lemma A.2 *Fix $i, j \in D$ with $i \neq j$. Then, for any $h, s \in \mathbb{R}$ with $hs \geq 0$ and $\boldsymbol{b} \in \mathbb{R}^d$, we have that*

$$\mathscr{S}(\boldsymbol{b} + h\boldsymbol{e}_i, s) \cap \mathscr{S}(\boldsymbol{b} + h\boldsymbol{e}_j, s) = \begin{cases} \mathscr{S}(\boldsymbol{b} + h\boldsymbol{e}_j + h\boldsymbol{e}_i, s - h) & \text{if } |h| < |s|, \\ \emptyset & \text{if } |h| \geq |s|. \end{cases}$$

Proof of $\subset$. First assume $0 < s \leq h$. By definition (1.4), for a vector $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b} + h\boldsymbol{e}_i, s)$ we have that

$$x_k > b_k + \delta_{ik} h, k \in D \quad \text{and} \quad \sum_{k=1}^{d} (x_k - b_k - \delta_{ik} h) \leq s,$$

from which it follows that

$$x_j \leq b_j + s - \sum_{k \neq j} (x_k - b_k - \delta_{ik} h) < b_j + s \leq b_j + h,$$

i.e. $\boldsymbol{x} \notin \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_j, s)$. Now assume that $0 < h < s$. For a vector $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_i, s) \cap \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_j, s)$ we have that

$$x_k - b_k > 0, k \in D \quad \text{with} \quad x_i > b_i + h \text{ and } x_j > b_j + h. \tag{A.2}$$

Again $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_i, s)$, therefore $\sum_{k=1}^d (x_k - (b_k + h\delta_{ik})) \le s$. Subtracting h from both sides of the last inequality, we obtain

$$\sum_{k=1}^d \big(x_k - (b_k + h\delta_{ik} + h\delta_{jk})\big) \le s - h. \tag{A.3}$$

Equations (A.2) and (A.3) show that $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_j + h\boldsymbol{e}_i, s-h)$. The case $h, s < 0$ is analogous.

Proof of $\supset$. If $0 < s \le h$, there is nothing to show. Suppose then $0 < h < s$. For any fixed $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_j + h\boldsymbol{e}_i, s-h)$, (A.3) holds with $x_k - (b_k + h\delta_{ik} + h\delta_{jk}) > 0, k \in D$. By adding $h\delta_{jk}$ in the sum on the left-hand side and h to the right-hand side of (A.3), we find that

$$\sum_{k=1}^d (x_k - (b_k + h\delta_{ik})) \le s. \tag{A.4}$$

Since $(x_k - (b_k + h\delta_{ik}))$ is still positive for all $k \in D$, (A.4) shows that $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_i, s)$. By a similar reasoning, we also have that $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}+h\boldsymbol{e}_j, s)$. The case $h, s < 0$ is analogous, the case $hs = 0$ is trivial. □

Lemma A.3 *For any $\boldsymbol{b} \in \mathbb{R}^d$, $h \in \mathbb{R}$ and $\alpha \in (0,1)$ we have that*

$$\mathscr{S}(\boldsymbol{b}, h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h) = \textstyle\bigcup_{k=1}^d \mathscr{S}(\boldsymbol{b} + \alpha h\boldsymbol{e}_k, h - \alpha h).$$

Proof of $\subset$. First assume $h > 0$. If $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}, h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)$ then $x_k > b_k, k \in D$ and $\sum_{k=1}^d (x_k - b_k) \le h$ while, by definition (1.2), there exists a $j \in D$ such that $x_j - b_j > \alpha h$. For this j, it is then possible to write

$$\sum_{k=1}^d x_k - (b_k + \delta_{jk}\alpha h) \le h - \alpha h \quad \text{with} \quad x_k - (b_k + \delta_{jk}\alpha h) > 0, k \in D, \tag{A.5}$$

which yields $\boldsymbol{x} \in \bigcup_{k=1}^d \mathscr{S}(\boldsymbol{b} + \alpha h\boldsymbol{e}_k, h - \alpha h)$.

Proof of $\supset$. Let $\boldsymbol{x} \in \bigcup_{k=1}^d \mathscr{S}(\boldsymbol{b} + \alpha h\boldsymbol{e}_k, h - \alpha h)$, meaning that there exists $j \in D$ for which $\boldsymbol{x}$ satisfies (A.5). It follows that $x_j > b_j + \alpha h$ (hence $\boldsymbol{x} \notin \mathscr{Q}(\boldsymbol{b}, \alpha h)$) and $\sum_{k=1}^d (x_k - b_k) \le h - \alpha h + \alpha h = h$. Noting that (A.5) also implies $x_k > b_k, k \in D$, we finally obtain that $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b}, h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)$. The case $h < 0$ is analogous, while the case $h = 0$ is trivial. □

Lemma A.4 *For any $\boldsymbol{b} \in \mathbb{R}^d$, $h \in \mathbb{R}$ and $\alpha \in [1/d, 1)$ we have that*

$$\mathscr{Q}(\boldsymbol{b}, \alpha h) \setminus \mathscr{S}(\boldsymbol{b}, h) = \mathscr{S}(\boldsymbol{b} + \alpha h\mathbf{1}, h - \alpha dh) \cap \mathscr{Q}(\boldsymbol{b}, \alpha h).$$

Proof of $\subset$. If $\alpha = 1/d$, the lemma is straightforward. Choose then $\alpha \in (1/d, 1)$ and assume $h > 0$. If $\boldsymbol{x} \in \mathscr{Q}(\boldsymbol{b}, \alpha h) \setminus \mathscr{S}(\boldsymbol{b}, h)$, then $x_k > b_k$ for all $k \in D$. Since $\boldsymbol{x} \notin \mathscr{S}(\boldsymbol{b}, h)$, it follows that $\sum_{i=1}^d (x_i - b_i) > h$. Since $x_k \le b_k + \alpha h$ for all $k \in D$, we can write

$$\sum_{k=1}^d (x_k - b_k - \alpha h) > h - \alpha dh \text{ with } x_k - b_k - \alpha h \le 0 \text{ for all } k \in D. \tag{A.6}$$

As $h - d\alpha h = h(1 - d\alpha) < 0$ we conclude that $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b} + \alpha h\mathbf{1}, h - \alpha dh)$ and hence, by assumption, $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b} + \alpha h\mathbf{1}, h - \alpha dh) \cap \mathscr{Q}(\boldsymbol{b}, \alpha h)$.

Proof of $\supset$. Let $\boldsymbol{x} \in \mathscr{S}(\boldsymbol{b} + \alpha h\mathbf{1}, h - \alpha dh) \cap \mathscr{Q}(\boldsymbol{b}, \alpha h)$. Due to $h - \alpha dh < 0$, it follows that (A.6) holds, implying that $\sum_{k=1}^d (x_k - b_k) > h$, i.e. $\boldsymbol{x} \notin \mathscr{S}(\boldsymbol{b}, h)$. The case $h < 0$ is analogous, while the case $h = 0$ is trivial. □

We are now ready to prove the main result in this appendix.

Proof of Theorem A.1. The case $h = 0$ is trivial. Suppose then that $h \neq 0$. From the general property of two sets A, B that $B = (A \cup (B \setminus A)) \setminus (A \setminus B)$, $(A \setminus B) \subset A \cup (B \setminus A)$, and $A \cap (B \setminus A) = \emptyset$, it follows that

$$V_H\left[\mathscr{S}(\boldsymbol{b}, h)\right] = V_H\left[\mathscr{Q}(\boldsymbol{b}, \alpha h)\right] + V_H\left[\mathscr{S}(\boldsymbol{b}, h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)\right] - V_H\left[\mathscr{Q}(\boldsymbol{b}, \alpha h) \setminus \mathscr{S}(\boldsymbol{b}, h)\right]. \tag{A.7}$$

Using the notation $\mathscr{S}^k = \mathscr{S}(\boldsymbol{b} + \alpha h \boldsymbol{e}_k, h - \alpha h)$, Lemma A.3 implies for the second summand in (A.7) that

$$V_H\left[\mathscr{S}(\boldsymbol{b}, h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)\right] = V_H\left[\bigcup_{k=1}^{d} \mathscr{S}^k\right] = \sum_{k=1}^{d} (-1)^{k+1} \sum_{I \subset D, |I| = k} V_H\left[\bigcap_{i \in I} \mathscr{S}^i\right]. \tag{A.8}$$

Fix $I \subset D$, with $I = \{n_1, \ldots, n_k\}$, using iteratively Lemma A.2 yields

$$\bigcap_{i \in I} \mathscr{S}(\boldsymbol{b} + \alpha h \boldsymbol{e}_{n_i}, h - \alpha h) = \begin{cases} \mathscr{S}(\boldsymbol{b} + \alpha h \sum_{j=1}^{k} \boldsymbol{e}_{n_j}, h(1 - k\alpha)) & \text{if } k\alpha < 1, \\ \emptyset & \text{if } k\alpha \geq 1. \end{cases}$$

Substituting this last expression in (A.8) implies

$$\begin{aligned} V_H\left[\mathscr{S}(\boldsymbol{b}, h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)\right] &= \sum_{\substack{k \in D, \\ k\alpha < 1}} (-1)^{k+1} \sum_{\substack{\boldsymbol{i}_r \in \{0,1\}^d, \\ \#\boldsymbol{i}_r = k}} V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \boldsymbol{i}_r, h(1 - k\alpha))\right] \\ &= \sum_{\substack{\boldsymbol{i} \in \{0,1\}^d, \\ 0 < \#\boldsymbol{i} < 1/\alpha}} (-1)^{\#\boldsymbol{i}+1} V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \boldsymbol{i}, h(1 - \#\boldsymbol{i}\alpha))\right]. \end{aligned} \tag{A.9}$$

Using Lemma A.4 for the third summand in (A.7), we can also write that

$$\begin{aligned} V_H\left[\mathscr{Q}(\boldsymbol{b}, \alpha h) \setminus \mathscr{S}(\boldsymbol{b}, h)\right] &= V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \mathbf{1}, h - \alpha d h) \cap \mathscr{Q}(\boldsymbol{b}, \alpha h)\right] \\ &= V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \mathbf{1}, h - \alpha d h)\right] - V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \mathbf{1}, h - \alpha d h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)\right]. \end{aligned} \tag{A.10}$$

Note that if $\alpha = 1/d$ the quantity in (A.10) is zero. We can hence assume that $\alpha \neq 1/d$. Observing that $\mathscr{Q}(\boldsymbol{b}, \alpha h) = \mathscr{Q}(\boldsymbol{b} + \alpha h \mathbf{1}, -\alpha h)$, and defining $\hat{\boldsymbol{b}} = \boldsymbol{b} + \alpha h \mathbf{1}$, $\hat{\alpha} = -\alpha/(1 - \alpha d) > 1/d$ and $\hat{h} = h(1 - \alpha d)$ we can write

$$V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \mathbf{1}, h - \alpha d h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)\right] = V_H\left[\mathscr{S}(\hat{\boldsymbol{b}}, \hat{h}) \setminus \mathscr{Q}(\hat{\boldsymbol{b}}, \hat{\alpha}\hat{h})\right].$$

Note that the right-hand side of the previous equation is empty if $\hat{\alpha} \geq 1$, i.e. $\alpha \in (1/d, 1/(d-1)]$. At this point, equation (A.9) yields

$$\begin{aligned} V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \mathbf{1}, h - \alpha d h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)\right] &= \sum_{\substack{\boldsymbol{i} \in \{0,1\}^d, \\ 0 < \#\boldsymbol{i} < 1/\hat{\alpha}}} (-1)^{\#\boldsymbol{i}+1} V_H\left[\mathscr{S}(\hat{\boldsymbol{b}} + \hat{\alpha}\hat{h}\boldsymbol{i}, \hat{h}(1 - \#\boldsymbol{i}\hat{\alpha}))\right] \\ &= \sum_{\substack{\boldsymbol{i} \in \{0,1\}^d, \\ 0 < \#\boldsymbol{i} < d - 1/\alpha}} (-1)^{\#\boldsymbol{i}+1} V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h(\mathbf{1} - \boldsymbol{i}), h(1 - \alpha(d - \#\boldsymbol{i})))\right]. \end{aligned}$$

Substituting $\hat{\boldsymbol{\imath}} = \mathbf{1} - \boldsymbol{i}$ ($\#\hat{\boldsymbol{\imath}} = d - \#\boldsymbol{i}$) in the previous equation, we can equivalently write

$$V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \mathbf{1}, h - \alpha d h) \setminus \mathscr{Q}(\boldsymbol{b}, \alpha h)\right] = \sum_{\substack{\hat{\boldsymbol{\imath}} \in \{0,1\}^d, \\ 1/\alpha < \#\hat{\boldsymbol{\imath}} < d}} (-1)^{d - \#\hat{\boldsymbol{\imath}}+1} V_H\left[\mathscr{S}(\boldsymbol{b} + \alpha h \hat{\boldsymbol{\imath}}, h(1 - \#\hat{\boldsymbol{\imath}}\alpha))\right]. \tag{A.11}$$

Coherently with what remarked above, this last equation is null in the above mentioned case in which $\hat{\alpha} \geq 1$. Recalling (A.10), and noting that

$$\mathscr{S}(\boldsymbol{b}+\alpha h\mathbf{1}, h-\alpha d h) = \mathscr{S}(\boldsymbol{b}+\alpha h\boldsymbol{i}_N, h(1-\#\boldsymbol{i}_N\alpha)),$$

we obtain

$$\begin{aligned}
&V_H\left[\mathscr{Q}(\boldsymbol{b},h)\setminus\mathscr{S}(\boldsymbol{b},\alpha h)\right]\\
&= V_H\left[\mathscr{S}(\boldsymbol{b}+\alpha h\boldsymbol{i}_N, h(1-\#\boldsymbol{i}_N\alpha))\right] - \sum_{\substack{\hat{\boldsymbol{i}}\in\{0,1\}^d,\\ 1/\alpha<\#\hat{\boldsymbol{i}}<d}} (-1)^{d-\#\hat{\boldsymbol{i}}+1} V_H\left[\mathscr{S}(\boldsymbol{b}+\alpha h\hat{\boldsymbol{i}}, h(1-\#\hat{\boldsymbol{i}}\alpha))\right]\\
&\qquad = \sum_{\substack{\hat{\boldsymbol{i}}\in\{0,1\}^d,\\ 1/\alpha<\#\hat{\boldsymbol{i}}\leq d}} (-1)^{d-\#\hat{\boldsymbol{i}}} V_H\left[\mathscr{S}(\boldsymbol{b}+\alpha h\hat{\boldsymbol{i}}, h(1-\#\hat{\boldsymbol{i}}\alpha))\right].
\end{aligned}\tag{A.12}$$

Finally, recalling the definitions in (A.1), we substitute equations (A.9) and (A.12) into (A.7) to obtain

$$\begin{aligned}
V_H\left[\mathscr{S}(\boldsymbol{b},h)\right] &= V_H\left[\mathscr{Q}(\boldsymbol{b},\alpha h)\right] + \sum_{\substack{\boldsymbol{i}\in\{0,1\}^d,\\ 0<\#\boldsymbol{i}<1/\alpha}} (-1)^{\#\boldsymbol{i}+1} V_H\left[\mathscr{S}(\boldsymbol{b}+\alpha h\boldsymbol{i}, h(1-\#\boldsymbol{i}\alpha))\right]\\
&\quad - \sum_{\substack{\hat{\boldsymbol{i}}\in\{0,1\}^d,\\ 1/\alpha<\#\hat{\boldsymbol{i}}\leq d}} (-1)^{d-\#\hat{\boldsymbol{i}}} V_H\left[\mathscr{S}(\boldsymbol{b}+\alpha h\hat{\boldsymbol{i}}, h(1-\#\hat{\boldsymbol{i}}\alpha))\right]\\
&\qquad = V_H\left[\mathscr{Q}(\boldsymbol{b},\alpha h)\right] + \sum_{j=1}^{N} m^j V_H\left[\mathscr{S}(\boldsymbol{b}^j, h^j)\right]. \qquad \square
\end{aligned}$$

References

Aas, K., X. K. Dimakos, and A. Øksendal (2007). Risk capital aggregation. *Risk Management 9*(2), 82–107.

Asmussen, S. and P. W. Glynn (2007). *Stochastic Simulation: Algorithms and Analysis*, Volume 57. New York: Springer.

Basel Committee on Banking Supervision (2006). *International Convergence of Capital Measurement and Capital Standards.* Basel: Bank for International Settlements.

Bürgi, R., M. Dacorogna, and R. Iles (2008). Risk aggregation, dependence structure and diversification benefit. In D. Rösch and H. Scheule (Eds.), *Stress-testing for Financial Institutions. Applications, Regulations and Techniques.* London: Risk Books.

Caflisch, R. E., W. Morokoff, and A. Owen (1997). Valuation of mortgage-backed securities using brownian bridges to reduce effective dimension. *Journal of Computational Finance 1*(1), 27–46.

Davis, P. J. and P. Rabinowitz (1984). *Methods of Numerical Integration.* Orlando, FL: Academic Press. Second edition.

Embrechts, P. (2009). Copulas: a personal view. Journal of Risk and Insurance, to appear.

Gill, H. S. and C. Lemieux (2007). Searching for extensible Korobov rules. *J. Complexity 23*(4-6), 603–613.

Glasserman, P. (2004). *Monte Carlo Methods in Financial Engineering.* New York: Springer-Verlag.

L'Ecuyer, P. and C. Lemieux (2000). Variance reduction via lattice rules. *Management Science 46*(9), 1214–1235.

McLeish, D. L. (2005). *Monte Carlo Simulation and Finance*. Hoboken, NJ: John Wiley & Sons.

McLeish, D. L. (2008). Bounded relative error importance sampling and rare event simulation. Preprint, University of Waterloo.

Morokoff, W. J. (1998). Generating quasi-random paths for stochastic processes. *SIAM Rev. 40*(4), 765–788.

Nelsen, R. B. (2006). *An Introduction to Copulas*. New York: Springer. Second edition.

Niederreiter, H. (1992). *Random Number Generation and Quasi-Monte Carlo Methods*, Volume 63 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Philadelphia: SIAM.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery (2007). *Numerical Recipes: the Art of Scientific Computing*. Cambridge: Cambridge University Press. Third Edition.

SCOR (2008). *From Principle Based Risk Management to Solvency Requirements*. Swiss Solvency Test Documentation, SCOR, Switzerland.

Weinzierl, S. (2000). Introduction to Monte Carlo Methods. Eprint arXiv:hep-ph/0006269.