Sampling Nested Archimedean Copulas

Alexander J. McNeil*
Maxwell Institute for the Mathematical Sciences &
Department of Actuarial Mathematics and Statistics
Heriot-Watt University
Edinburgh EH14 4AS, Scotland
A.J.McNeil@hw.ac.uk
Tel. +44 131 451 3230 Fax +44 131 451 3327

12th January 2007

Abstract

We give algorithms for sampling from non-exchangeable Archimedean copulas created by the nesting of Archimedean copula generators, where in the most general algorithm the generators may be nested to an arbitrary depth. These algorithms are based on representations of these copulas using Laplace transforms. Precise instructions are given for the case when all generators are taken from the Gumbel parametric family or the Clayton family; the Gumbel case in particular proves very easy to simulate.

AMS 2000 MSC: 00A72, 11K45, 60E10, 62E15, 65C05

Keywords: Archimedean copulas, Laplace transforms, stochastic simulation

1 Introduction

Copulas are multivariate distribution functions with standard uniform marginal distributions and Archimedean copulas are a prominent class of copulas with a common method of construction involving one-dimensional generator functions; standard textbook references are Joe [5] and Nelsen [9]. There has been recent interest in multivariate Archimedean copulas, partly prompted by new applications in financial modelling; see Schönbucher [11] and Cherubini et al. [1]. In these applications Archimedean copulas are used as parsimonious parametric models for the dependence structure of random

*I thank Christoph Hummel of Converium for motivating this problem in the context of modelling dependent insurance losses. My thanks also go to Martin Pfister and Johanna Nešlehová for many useful discussions.

vectors, particularly in situations where data are not available to estimate more complex multivariate distributions. Because the resulting models are frequently used in a Monte Carlo context, it is very important that we are able to sample Archimedean copulas and this paper is a contribution to that subject.

A bivariate Archimedean copula is constructed according to

$$C(u_1, u_2) = \psi(\psi^{-1}(u_1) + \psi^{-1}(u_2)), \tag{1}$$

where $\psi : [0, \infty) \to [0, 1]$ is any continuous, decreasing, convex function satisfying $\psi(0) = 1$ and $\psi(\infty) := \lim_{t\to\infty} \psi(t) = 0$. The function ψ is known as the copula generator and ψ^{-1} is its inverse, defined in general by $\psi^{-1}(u) = \inf\{t : \psi(t) \leq u\}$ where, by convention, $\inf \emptyset = \infty$. A comprehensive reference for bivariate Archimedean copulas is Nelsen [9].

One way of extending the bivariate copula to higher dimensions is to use the so-called exchangeable construction

$$C(u_1, \ldots, u_d) = \psi\left(\psi^{-1}(u_1) + \ldots + \psi^{-1}(u_d)\right). \tag{2}$$

Kimberling [6] showed that this gives a copula in any dimension d if and only if the generator ψ is a completely monotonic function, so that it satisfies

$$(-1)^k \frac{d^k}{dt^k}\psi(t) \geq 0, \quad k \in \mathbb{N}, \quad t > 0. \tag{3}$$

We refer to completely monotonic generators as LT-Archimedean copula generators, where LT stands for Laplace transform. This is because the class of completely monotonic functions ψ on $[0, \infty)$ with $\psi(0) = 1$ coincides with the class of Laplace-Stieltjes transforms (henceforth simply Laplace transforms) of distribution functions on $[0, \infty)$ according to Bernstein's Theorem (1928) (see Feller [4] or Widder [14]).

The representation of Archimedean copulas using Laplace transforms leads to a very useful way of simulating such copulas, which was recognized by Marshall and Olkin [7]. The textbook of Joe [5] contains much information on the link between Archimedean copulas and Laplace transforms; see also McNeil et al. [8]. A recent reference on sampling Archimedean copulas is Whelan [13], who discusses methods based on Laplace transforms as well as other integral representations.

Random vectors with the distribution (2) have a highly specialized form of dependence structure: they are exchangeable or, in other words, distributionally invariant under permutations. Moreover they are models for positively dependent random vectors in the sense that the bivariate margins are positive quadrant dependent (PQD) (see Nelsen [9], page 151), which implies that pairwise rank correlations are non-negative. There has been interest in constructing more flexible multivariate Archimedean copulas and

Joe [5] discusses Archimedean copulas constructed by the nesting of generators. These models continue to have bivariate Archimedean margins that are PQD but they do allow different degrees of positive dependence in different bivariate margins.

A three-dimensional example can be constructed using two LT-Archimedean generators, ψ_1 and ψ_2, according to

$$C(u_1, u_2, u_3) = \psi_1 \left(\psi_1^{-1}(u_1) + \psi_1^{-1} \circ \psi_2(\psi_2^{-1}(u_2) + \psi_2^{-1}(u_3)) \right). \tag{4}$$

Under the additional condition that the first derivative of $\psi_1^{-1} \circ \psi_2$ is completely monotonic (a condition that will later make sense) the construction (4) yields a proper distribution function.

Random sampling from (4) can be based on a representation using Laplace transforms given in Joe [5], but this does not seem to be a known result. We explain how sampling is performed and then go on to consider higher-dimensional constructions. We give algorithms for sampling from the four-dimensional construction

$$\begin{aligned} &C(u_1, u_2, u_3, u_4) = \\ &\quad \psi_1 \left(\psi_1^{-1} \circ \psi_2(\psi_2^{-1}(u_1) + \psi_2^{-1}(u_2)) + \psi_1^{-1} \circ \psi_3(\psi_3^{-1}(u_3) + \psi_3^{-1}(u_4)) \right) \end{aligned} \tag{5}$$

and a d-dimensional construction defined iteratively for $d \geq 2$ by

$$\begin{aligned} &C_d(u_1, \ldots, u_{d+1}; \psi_1, \ldots, \psi_d) = \\ &\quad \psi_1 \left(\psi_1^{-1}(u_1) + \psi_1^{-1} \left(C_{d-1}(u_2, \ldots, u_{d+1}; \psi_2, \ldots, \psi_d) \right) \right), \end{aligned} \tag{6}$$

where in this notation $C_1(u_1, u_2; \psi)$ refers to the bivariate copula (1) and where $C_2(u_1, u_2, u_3; \psi_1, \psi_2)$ is the copula (4). We note again that further conditions apply to the LT-Archimedean generators in order for the constructions (5) and (6) to yield valid multivariate distribution functions. A reader who understands how sampling is achieved for (4), (5) and (6) will see how the ideas may be adapted to even more general Archimedean copulas with nested generators.

The paper is structured as follows. In the next section we review the generation of random vectors with exchangeable Archimedean copulas using the Laplace transform method. In Section 3 we discuss the construction (4), present simulation algorithms and give parametric examples. In Section 4 we treat the d-dimensional construction (6) and give a theoretical result to justify both a recursive and a fully explicit algorithm. Examples are given in Section 5.

2 The Exchangeable Case

Let ψ be a completely monotonic function on $[0, \infty)$ with $\psi(0) = 1$ and $\lim_{t \to \infty} \psi(t) = 0$ and let G be the distribution function on $[0, \infty)$ with

Laplace-Stieltjes transform given by ψ so that

$$\psi(t) = \int_0^\infty e^{-tx} dG(x),\ t \geq 0. \tag{7}$$

Note that the requirement that $\lim_{t\to\infty} \psi(t) = 0$ means that the distribution may not have point mass at zero (so that $G(0) = 0$) but otherwise there are no constraints on the distribution function G; in particular, it may be discrete or continuous.

Marshall and Olkin showed explicitly how to construct a random vector $(U_1, \ldots, U_d)$ with distribution function given by the copula (2). Let $F(u) := \exp(-\psi^{-1}(u))$ and observe that $F^v(u)$ is a valid univariate distribution function on $[0, 1]$ for any $v > 0$. If V is a random variable with distribution function G and $U_1, \ldots, U_d$ are conditionally independent given V with conditional distribution function given by $P(U_i \leq u \mid V = v) = F^v(u)$ then $(U_1, \ldots, U_d)$ has the required distribution function, as is seen by observing that

$$\begin{aligned}
P\left(U_1 \leq u_1, \ldots, U_d \leq u_d\right) &= \int_0^\infty P\left(U_1 \leq u_1, \ldots, U_d \leq u_d \mid V = v\right) dG(v) \\
&= \int_0^\infty \prod_{i=1}^d F^v(u_i) dG(v) \\
&= \int_0^\infty e^{-v\left(\psi^{-1}(u_1)+\cdots+\psi^{-1}(u_d)\right)} dG(v) \\
&= \psi\left(\psi^{-1}(u_1) + \cdots + \psi^{-1}(u_d)\right).
\end{aligned}$$

In summary, the algorithm for sampling this copula has the following steps.

Algorithm 1.

1. Generate a variate V with distribution function G with LT ψ.

2. Generate independent uniform variates $X_1, \ldots, X_d$.

3. Return $(U_1, \ldots, U_d) = (\psi(-\ln(X_1)/V), \ldots, \psi(-\ln(X_d)/V))$.

To apply this method in practice we either begin with a particular distribution function G that we can sample and for which we can calculate the Laplace transform, or we begin with a particular completely monotonic generator ψ and search for the distribution function G for which ψ is the Laplace transform. In this paper we adopt the latter approach, although we note that it is often difficult to find the distribution function G corresponding to a given ψ and, even when G can be found, we still have the practical problem of sampling the corresponding distribution. Clearly a generic method to sample distributions with given Laplace transforms would be desirable, but we are not aware of a viable general solution and we are forced to consider Archimedean copulas on a case by case basis. Algorithms proposed by Devroye [2, 3] can help in certain cases, particularly the case where G is the distribution function of a discrete random variable.

In this paper we restrict attention to two of the more common one-parameter families of copulas, these being the Gumbel and Clayton copula families as given in Table 1. The variables V to be sampled have respectively positive stable and gamma distributions.

	Gumbel	Clayton
Copula Summary		
$\psi(t;\theta)$	$\exp(-t^{1/\theta})$	$(1+t)^{-1/\theta}$
Parameter space	$\theta \geq 1$	$\theta > 0$
$\psi^{-1}(u;\theta)$	$(-\ln u)^{\theta}$	$u^{-\theta}-1$
Distribution of V	$\mathrm{St}(1/\theta, 1, (\cos(\pi/(2\theta)))^{\theta}, 0)$	$\mathrm{Ga}(1/\theta, 1)$
Inner Copula Summary		
$\psi_2^{(1)}(t; v, \theta_1, \theta_2)$	$\exp(-vt^{\theta_1/\theta_2})$	$\exp\left(v - v(1+t)^{\theta_1/\theta_2}\right)$
equivalent generator	$\exp(-t^{1/\alpha})$	$\exp\left(v - (t+v^{\alpha})^{1/\alpha}\right)$

Table 1: Copula generators and inner copula generators for the Gumbel and Clayton parametric families. For details of the stable distribution to be sampled in the Gumbel case, see Appendix A.

3 The Trivariate Nested Case

Assume that ψ_1 and ψ_2 are LT-Archimedean and let G_1 be the distribution function with Laplace transform ψ_1. We note first that (4) may be represented as

$$C(u_1, u_2, u_3) = \int_0^{\infty} e^{-v\psi_1^{-1}(u_1)} e^{-v\psi_1^{-1}\circ\psi_2(\psi_2^{-1}(u_2)+\psi_2^{-1}(u_3))} dG_1(v). \qquad (8)$$

Following Joe [5] we will show that when the derivative of $\psi_1^{-1} \circ \psi_2$ is completely monotonic, the integral representation can be interpreted as a mixture of distributions.

Let $F_i(u) := \exp(-\psi_i^{-1}(u))$ for $i = 1, 2$ and note that the functions $F_i^v(u)$ are valid univariate distribution functions for any $v > 0$. The first term in the integrand is $F_1^v(u_1)$ and the second may be written as

$$\begin{aligned} H(u_2, u_3; v) &:= F_1^v(\psi_2(\psi_2^{-1}(u_2) + \psi_2^{-1}(u_3))) \\ &= \psi_2^{(1)}(\psi_2^{(1)\,-1}(F_1^v(u_2); v) + \psi_2^{(1)\,-1}(F_1^v(u_3); v); v) \end{aligned} \tag{9}$$

where the function $\psi_2^{(1)}(\cdot\,; v)$ is given by

$$\psi_2^{(1)}(\cdot\,; v) = F_1^v(\psi_2(\cdot)) = \exp(-v\psi_1^{-1} \circ \psi_2(\cdot)). \tag{10}$$

Under our assumption that the derivative of $\psi_1^{-1} \circ \psi_2$ is completely monotonic, the function $\psi_2^{(1)}(\cdot\,; v)$ in (10) is an LT-Archimedean copula generator so that (9) is a bivariate distribution function on $[0, 1]^2$ with marginal distributions $F^v(u)$. The complete monotonicity of the generator $\psi_2^{(1)}(\cdot\,; v)$ follows from a result in Feller [4] (see also Nelsen [9], page 123), which says that if f is completely monotonic and g is a positive function with a completely monotonic derivative then $f \circ g$ is completely monotonic. This is applied here by taking $f(t)$ to be $\exp(-vt)$ and g to be $\psi_1^{-1} \circ \psi_2$.

We can rewrite (8) as

$$C_2(u_1, u_2, u_3; \psi_1, \psi_2) = \int_0^\infty F_1^v(u_1) C_1(F_1^v(u_2), F_1^v(u_3); \psi_2^{(1)}(\cdot\,; v)) dG_1(v), \tag{11}$$

using the notation of (6). This represents the copula as a mixture distribution and reveals how it may be sampled.

Algorithm 2.

1. Generate a variate V with distribution function G_1 with LT ψ_1.

2. Generate a uniform variate X_1.

3. Generate (X_2, X_3) from the bivariate Archimedean copula with generator $\psi_2^{(1)}(\cdot\,; V) = \exp(-V\psi_1^{-1} \circ \psi_2(\cdot))$ using Algorithm 1.

4. Return (U_1, U_2, U_3) where $U_i = \psi_1(-\ln(X_i)/V)$ for $i = 1, 2, 3$.

Assuming we can accomplish step (1) for the generator ψ_1 the next practical concern is whether we can accomplish step (3) for the generator $\psi_2^{(1)}$. We will refer to this step as sampling the inner copula. In the absence of a generic method for generators we are again forced to consider copulas on a case by case basis.

The first issue is what generators ψ_1 and ψ_2 we can mix while ensuring that the derivative of $\psi_1^{-1} \circ \psi_2$ is completely monotonic. This has again been considered in some detail in Joe [5], who looks in particular at the case when we take generators $\psi_1(\cdot) = \psi(\cdot\,; \theta_1)$ and $\psi_2(\cdot) = \psi(\cdot\,; \theta_2)$ in the same parametric family $\psi(\cdot\,; \theta)$ for some parameter values θ_1 and θ_2. For many

LT-Archimedean generators, including the Gumbel and Clayton generators in Table 1, it may be verified that the function $\psi_1^{-1} \circ \psi_2$ has a completely monotonic derivative if and only if $\theta_2 \geq \theta_1$.

We note that this does have an implication for the kind of dependence structure that the copula (4) can represent. Suppose (U_1, U_2, U_3) is a random vector with this distribution. Then the pairs (U_1, U_2) and (U_1, U_3) have bivariate marginal distributions that are Archimedean copulas with generator $\psi(\cdot\,; \theta_1)$ and the pair (U_2, U_3) is distributed according to an Archimedean copula with generator $\psi(\cdot\,; \theta_2)$. The requirement that $\theta_1 < \theta_2$ means that the pair (U_2, U_3) is more concordant than the pairs (U_1, U_2) and (U_1, U_3), so that the rank correlation (Spearman or Kendall) between the inner pair (U_2, U_3) is higher than that between the other pairs; see Nelsen [9] for more details.

The generator $\psi_2^{(1)}$ of the inner copula depends in general on three parameters: v, the realization of the random draw in the first step of the algorithm, θ_1 and θ_2. For the two copula families in Table 1 we give the inner copula generators obtained from (10).

We note however that some simplification is possible. First, we observe that in both cases the dependence of the inner copula on θ_1 and θ_2 is through the ratio $\alpha = \theta_2/\theta_1 > 1$, so that we could parameterize these generators in terms of α. Second we note that for any generator ψ and any $k > 0$, the generator $\tilde{\psi}(t) := \psi(kt)$ yields exactly the same copula when used in Archimedean constructions of the kind (2), (4) and (6).

In the case of the Gumbel copula, the inner copula generator may be replaced by the Gumbel generator $\exp(-t^{1/\alpha})$. This means that Algorithm 2 is particularly easy in the Gumbel case; in step (3) we generate (X_2, X_3) from a Gumbel copula with generator $\psi(\cdot, \theta_2/\theta_1)$. The realization of V actually plays no role in the inner copula generation for this copula, which is not generally the case.

In the Clayton case we can also give an equivalent generator which is closely related to the Gumbel generator. In fact, if $g(x; \alpha)$ denotes the density of the stable distribution with Laplace transform $\psi(t; \alpha) = \exp(-t^{1/\alpha})$ then the equivalent generator is the Laplace transform of the tilted stable density

$$f(x; v, \alpha) = \frac{e^{-v^\alpha x} g(x; \alpha)}{\psi(v^\alpha; \alpha)}. \tag{12}$$

This can be sampled in principle using a rejection method with g as envelope, although we note that this may be slow (because of a high rejection probability) for some values of v and α.

Algorithm 2 is a recursive algorithm in the sense that the three-dimensional copula can be sampled if we have already implemented a method for the simple two-dimensional exchangeable copula (and the required inner copula generator). It is of course possible to give a fully explicit algorithm for simulating the three-dimensional copula. This may be based on further development of the mixture representation (11). By using the argument of Section 2 we have that

$$C_1(u_2, u_3; \psi_2^{(1)}(\cdot\,; v)) = \int_0^\infty e^{-v_2 \psi_2^{(1)\ -1}(u_2;v)} e^{-v_2 \psi_2^{(1)\ -1}(u_3;v)} dG_2(v_2; v)$$

where $G_2(\cdot\,; v)$ is the distribution function with Laplace transform $\psi_2^{(1)}(t; v)$. Moreover, since $\psi_2^{(1)\ -1}(F_1^v(u); v) = \psi_2^{-1}(u)$ we can rewrite (11) as

$$C_2(u_1, u_2, u_3; \psi_1, \psi_2) = \int_0^\infty \int_0^\infty F_1^{v_1}(u_1) F_2^{v_2}(u_2) F_2^{v_2}(u_3) dG_2(v_2; v_1) dG(v_1). \tag{13}$$

Both the mixture representation (13) and the representation (11) may be found in slightly different form in [5] (see pages 87–88) but their implications for stochastic simulation are not discussed. Representation (13) yields the following explicit algorithm

Algorithm 3.

1. Generate a variate V_1 with distribution function G_1 with LT ψ_1.
2. Generate a variate V_2 with distribution function $G_2(v; V_1)$ with LT $\psi_2^{(1)}(\cdot; V_1) = \exp(-V_1 \psi_1^{-1} \circ \psi_2(\cdot))$
3. Generate independent uniform variates X_1, X_2, X_3.
4. Return (U_1, U_2, U_3) where $U_i = \psi_i(-\ln(X_i)/V_i)$ for $i = 1, 2$ and $U_3 = \psi_2(-\ln(X_3)/V_2)$.

We end this section by giving a recursive algorithm for the 4-dimensional copula (5), where we assume that ψ_1, ψ_2 and ψ_3 are LT-Archimedean and both $\psi_1^{-1} \circ \psi_2$ and $\psi_1^{-1} \circ \psi_3$ are functions with completely monotonic derivatives. The justification of the algorithm follows easily from the kind of arguments employed in this section.

Algorithm 4.

1. Generate a variate V with distribution function G_1 with Laplace transform ψ_1.
2. Generate (X_1, X_2) from the bivariate Archimedean copula with generator $\psi_2^{(1)}(\cdot\,; V) = \exp(-V \psi_1^{-1} \circ \psi_2(\cdot))$ using Algorithm 1.
3. Generate (X_3, X_4) from the bivariate Archimedean copula with generator $\psi_3^{(1)}(\cdot\,; V) = \exp(-V \psi_1^{-1} \circ \psi_2(\cdot))$ using Algorithm 1.
4. Return (U_1, U_2, U_3, U_4) where $U_i = \psi_1(-\ln(X_i)/V)$ for $i = 1, \ldots, 4$.

4 The General Nested Case

In this section we show how the ideas of the previous section may be applied to sample copulas where generators are nested arbitrarily deeply, such as the copula in (6).

Throughout the section we will assume that $\psi_1, \ldots, \psi_d$ are LT-Archimedean copula generators and that $\psi_k^{-1} \circ \psi_{k+1}$ have completely monotonic derivatives for $k = 1, \ldots, d-1$. Note that this implies that $\psi_k^{-1} \circ \psi_j$ have completely monotonic derivatives for any $j > k$, as may be seen from the following kind of argument.

Lemma 4.1. *If $\psi_1^{-1} \circ \psi_2$ and $\psi_2^{-1} \circ \psi_3$ have completely monotonic derivatives then so does $\psi_1^{-1} \circ \psi_3$.*

Proof. Let $f := \psi_1^{-1} \circ \psi_2$, $g := \psi_2^{-1} \circ \psi_3$ and $h := \psi_1^{-1} \circ \psi_3$, so that $h = f \circ g$. The derivative is $h'(t) = f'(g(t))g'(t)$ on $t > 0$ and two lemmas of Feller [4] (see also Nelsen [9], page 123) guarantee that h' is completely monotonic: $f' \circ g$ is completely monotonic, as it is a completely monotonic function of a function with a completely monotonic derivative, and h' is completely monotonic because it is a product of completely monotonic functions. □

Now suppose that we define for $k = 1, \ldots, d-1$, $j > k$ and $v > 0$ the copula generators

$$\psi_j^{(k)}(\cdot\,; v) := \exp(-v\psi_k^{-1} \circ \psi_j(\cdot))$$

Clearly, by the argument used in Section 3, these are LT-Archimedean generators because they are completely monotonic. The role of these functions in this section is that they will be the generators of recursively nested inner copulas indexed by k, which is a consequence of the following property.

Lemma 4.2. *For $1 \leq k \leq d-2$, $k+1 < j \leq d$ and $v, \tilde{v} > 0$ we have*

$$\exp(-v\psi_{k+1}^{(k)\,-1}(\psi_j^{(k)}(\cdot\,; \tilde{v}); \tilde{v})) = \psi_j^{(k+1)}(\cdot\,; v).$$

Proof. This follows easily by observing that

$$\psi_{k+1}^{(k)\,-1}(\psi_j^{(k)}(\cdot\,; \tilde{v}); \tilde{v}) = \psi_{k+1}^{-1} \circ \psi_j(\cdot).$$

□

The following lemma on the manipulation of nested copula functions will also be useful.

Lemma 4.3. *Let $d \geq 1$ be given and assume that $\psi_1, \ldots, \psi_d$ are LT-Archimedean generators. Let $F : [0,1] \to [0,1]$ be a strictly increasing continuous function such that $\tilde{\psi}_k = F \circ \psi_k$, $k = 1, \ldots, d$ are also LT-Archimedean generators. Then the function C_d defined in (6) satisfies*

$$F(C_d(u_1, \ldots, u_{d+1}; \psi_1, \ldots, \psi_d)) = C_d(F(u_1), \ldots, F(u_{d+1}); \tilde{\psi}_1, \ldots, \tilde{\psi}_d).$$

Proof. Using the recursive definition (6), we have

$$\begin{aligned}
&F(C_d(u_1,\dots,u_{d+1};\psi_1,\dots,\psi_d))\\
&\quad= F\circ\psi_1\left(\psi_1^{-1}\circ F^{-1}(F(u_1))+\psi_1^{-1}\circ F^{-1}(F(C_{d-1}(u_2,\dots,u_{d+1};\psi_2,\dots,\psi_d)))\right)\\
&\quad= \tilde\psi_1\left(\tilde\psi_1^{-1}(F(u_1))+\tilde\psi_1^{-1}\left(F(C_{d-1}(u_2,\dots,u_{d+1};\psi_2,\dots,\psi_d))\right)\right)
\end{aligned}$$

If the identity

$$F(C_{d-1}(u_2,\dots,u_{d+1};\psi_2,\dots,\psi_d)) = C_{d-1}(F(u_2),\dots,F(u_{d+1});\tilde\psi_2,\dots,\tilde\psi_d)$$

can be assumed to hold, then the lemma follows again from (6). We proceed in this way by a process of backward induction, which ends by noting that

$$F(C_1(u_d,u_{d+1};\psi_d)) = C_1(F(u_d),F(u_{d+1});\tilde\psi_d).$$

□

Our main result gives two representations for nested Archimedean copulas, which will both yield simulation algorithms.

Theorem 4.4. *If $\psi_1,\dots,\psi_d$ are LT-Archimedean generators and $\psi_k^{-1}\circ\psi_{k+1}$ have completely monotonic derivatives for $k=1,\dots,d-1$ then the function $C_d(u_1,\dots,u_{d+1};\psi_1,\dots,\psi_d)$ defined in (6) is a copula and has the mixture representations*

$$\begin{aligned}
&C_d(u_1,\dots,u_{d+1};\psi_1,\dots,\psi_d)\\
&\quad=\int_0^\infty F_1^{v_1}(u_1)C_{d-1}\left(F_1^{v_1}(u_2),\dots,F_1^{v_1}(u_{d+1});\psi_2^{(1)}(\cdot\,;v_1),\dots,\psi_d^{(1)}(\cdot\,;v_1)\right)dG_1(v_1)\\
&\quad=\int_0^\infty\cdots\int_0^\infty F_1^{v_1}(u_1)\cdots F_d^{v_d}(u_d)F_d^{v_d}(u_{d+1})dG_d(v_d;v_{d-1})\cdots dG_2(v_2;v_1)dG_1(v_1),
\end{aligned}$$

where G_1 has LT ψ_1, $F_k(u)=\exp(-\psi_k^{-1}(u))$ for $k=1,\dots,d$ and $G_k(v;v_{k-1})$ is the distribution function with LT $\psi_k^{(k-1)}(\cdot\,;v_{k-1})$ for $k=2,\dots,d$.

Proof. If $C_d(u_1,\dots,u_{d+1};\psi_1,\dots,\psi_d)$ is a valid multivariate distribution function for any d under the conditions of the theorem, then it is obviously a copula, since the marginal distributions are standard uniform. We show that C_d is a valid multivariate df by deriving the mixture representations.

It follows from (6) and the complete monotonicity of ψ_1^{-1} that we can write

$$C_d(u_1,\dots,u_{d+1};\psi_1,\dots,\psi_d)=\int_0^\infty e^{-v_1\psi_1^{-1}(u_1)}e^{-v_1\psi_1^{-1}(C_{d-1}(u_2,\dots,u_{d+1};\psi_2,\dots,\psi_d))}dG_1(v_1),$$

where G_1 is a distribution function on $[0,\infty)$ with LT ψ_1. The first term in the product is $F_1^{v_1}(u_1)$, which is clearly a continuous and strictly increasing univariate distribution function on $[0,1]$. Moreover, we can use Lemma 4.3 to show that the second term is

$$C_{d-1}(F_1^{v_1}(u_2),\ldots,F_1^{v_1}(u_{d+1});\psi_2^{(1)}(\cdot\,;v_1),\ldots,\psi_d^{(1)}(\cdot\,;v_1))$$

If $C_{d-1}(u_2,\ldots,u_{d+1};\psi_2^{(1)}(\cdot\,;v_1),\ldots,\psi_d^{(1)}(\cdot\,;v_1))$ is a copula then this is a valid multivariate distribution function (by the Theorem of Sklar [12]; see also Nelsen [9], page 41) and $C_d(u_1,\ldots,u_{d+1};\psi_1,\ldots,\psi_d)$ will have the first mixture representation as asserted and be a multivariate distribution function.

Thus the problem is pushed back to showing that $C_{d-1}(u_2,\ldots,u_{d+1};\psi_2^{(1)}(\cdot\,;v_1),\ldots,\psi_d^{(1)}(\cdot\,;v_1))$ is a multivariate distribution function and hence a copula. In general, by repeating the above argument we will have, for $k=2,\ldots,d-1$, the sequence of representations

$$C_{d-k+1}(u_k,\ldots,u_{d+1};\psi_k^{(k-1)}(\cdot\,;v_{k-1}),\ldots,\psi_d^{(k-1)}(\cdot\,;v_{k-1}))=\int_0^\infty F_k^{v_k}(u_k;v_{k-1})$$
$$C_{d-k}\left(F_k^{v_k}(u_{k+1};v_{k-1}),\ldots,F_k^{v_k}(u_{d+1};v_{k-1});\psi_{k+1}^{(k)}(\cdot;v_k),\ldots,\psi_d^{(k)}(\cdot;v_k)\right)dG_k(v_k;v_{k-1}),$$

where $F_k(u;v_{k-1})=\exp(-\psi_k^{(k-1)\ -1}(u;v_{k-1}))$. At each step $k=2,\ldots,d-1$, if we can assume that

$$C_{d-k}\left(u_{k+1},\ldots,u_{d+1};\psi_{k+1}^{(k)}(\cdot\,;v_k),\ldots,\psi_d^{(k)}(\cdot\,;v_k)\right)$$

is a copula, the so-called kth inner copula, then we have the asserted mixture representation for the previous inner copula

$$C_{d-k+1}(u_k,\ldots,u_{d+1};\psi_k^{(k-1)}(\cdot\,;v_{k-1}),\ldots,\psi_d^{(k-1)}(\cdot\,;v_{k-1})).$$

The form of the generators of the inner copulas follows from Lemma 4.2. The inductive argument is anchored when we arrive at the ultimate inner copula $C_1(u_d,u_{d+1};\psi_d^{(d-1)}(\cdot\,;v_{d-1}))$. This is certainly a copula because its generator is completely monotonic and the argument of Section 2 shows that it has the representation

$$C_1(u_d,u_{d+1};\psi_d^{(d-1)}(\cdot\,;v_{d-1}))=\int_0^\infty F_d^{v_d}(u_d;v_{d-1})F_d^{v_d}(u_{d+1};v_{d-1})dG_d(v_d;v_{d-1}).$$

The second mixture representation is obtained by simplifying the sequence of nested representations and noting that for $k\geq 2$

$$F_k^{v_k}\left(F_{k-1}^{v_{k-1}}\left(\cdots F_2^{v_2}\left(F_1^{v_1}(u);v_1\right)\cdots;v_{k-2}\right);v_{k-1}\right)=F_k^{v_k}(u).$$

□

A recursive algorithm for generating random samples from the copula $C_d(u_1, \ldots, u_{d+1}; \psi_1, \ldots, \psi_d)$ based on the first mixture representation in Theorem 4.4 has the following steps.

Algorithm 5.

1. Generate a variate V with distribution function G_1 with LT ψ_1.
2. Generate X_1 from a standard uniform distribution.
3. Generate $(X_2, \ldots, X_{d+1})$ from the nested Archimedean copula $C_{d-1}(u_2, \ldots, u_{d+1}; \psi_2^{(1)}(\cdot\,; V), \ldots, \psi_d^{(1)}(\cdot\,; V))$.
4. Return $(U_1, \ldots, U_{d+1})$ where $U_i = \psi_1(-\ln(X_i)/V)$, $i = 1, \ldots, d+1$.

A fully explicit algorithm based on the second mixture representation has the following steps.

Algorithm 6.

1. Generate a variate V_1 with distribution function G_1 with LT ψ_1.
2. For $k = 2, \ldots d$, generate variates V_k with distribution functions $G_k(v; V_{k-1})$ with LTs $\psi_k^{(k-1)}(\cdot\,; V_{k-1}) = \exp(-V_{k-1}\psi_{k-1}^{-1} \circ \psi_k(\cdot))$
3. Generate independent uniform variates $X_1, \ldots, X_{d+1}$.
4. Return $(U_1, \ldots, U_{d+1})$ where $U_i = \psi_i(-\ln(X_i)/V_i)$ for $i = 1, \ldots, d$ and $U_{d+1} = \psi_d(-\ln(X_{d+1})/V_d)$.

Consider the nested Gumbel copula where the generators $\psi(\cdot\,; \theta_i)$, $i = 1, \ldots, d$ have the parametric form given in Table 1. The conditions of Theorem 4.4 are met if $1 \leq \theta_1 \leq \cdots \leq \theta_d$. In this case the recursive algorithm is an attractive way to proceed and can be implemented in any software that allows recursive definition of functions. The (first) inner copula simplifies to

$$C_{d-1}(u_2, \ldots, u_{d+1}; \psi(\cdot\,; \theta_2/\theta_1), \ldots, \psi(\cdot\,; \theta_d/\theta_1))$$

so that the generators do not involve V_1. In other words the inner copula is again Gumbel of dimension one less and with new parameters that are related in a simple way to the original parameters. An example of an appropriate recursive algorithm in the S language is given in Appendix B.

For the nested Clayton copula the conditions of Theorem 4.4 are met if $0 < \theta_1 \leq \cdots \leq \theta_d$. We could base a simulation algorithm on either mixture representation (the recursive or the explicit) but in both cases the practical problem of efficiently generating variates from exponentially tilted positive stable distributions will arise. The first inner copula simplifies slightly to

$$C_{d-1}(u_1, \ldots, u_{d+1}; \tilde{\psi}(\cdot\,; V_1, \theta_2/\theta_1), \ldots, \tilde{\psi}(\cdot\,; V_1, \theta_d/\theta_1))$$

where $\tilde{\psi}(\cdot\,; v, \alpha)$ is the LT-Archimedean generator which is the Laplace transform of the tilted positive stable density in (12). Note that the generators involve V_1, the value of the gamma-distributed mixing variable from the first step of the algorithm. A representation for the first inner copula in terms of the second inner copula (and in general for the kth inner copula in terms of the $k-1$ inner copula) will involve mixing over tilted stable densities. In terms of Algorithm 6 the sequence of variates $V_2, \ldots, V_d$ will all be from the tilted stable family.

5 Examples

In Figure 1 we show 3000 points from a 4-dimensional Clayton copula with structure (5) where the generators have parameters $\theta_1 = 1$, $\theta_2 = 3$ and $\theta_3 = 8$. These were generated using Algorithm 4. Note that we essentially obtain a 2-group model: the first group consists of (U_1, U_2) and these are distributed according to a bivariate Clayton copula with parameter θ_2; the second group consists of (U_3, U_4) and these are distributed according to a bivariate Clayton copula with parameter θ_3; the between-group dependence of any pair is described by a bivariate Clayton copula with parameter θ_1. The positive dependence within the first group is weaker than that within the second group and the between group dependence is weaker still. The Kendall's rank correlations ρ_τ in this model may be easily calculated using the formula $\rho_\tau = \theta/(\theta+2)$ (see Nelsen [9]) and take the values 0.6 and 0.8 within groups and 1/3 between groups. (Empirical estimates of these correlations from the simulated data provide a useful confirmation that implementation is accurate.)

In Figure 2 we show 3000 points from a 7-dimensional Gumbel copula with structure (6) where the generators have parameters $\theta_i = i,\ i = 1, \ldots, 6$. These data were generated using Algorithm 5 and the code in Appendix B. Here we obtain a multivariate uniform distribution in which dependence get stronger as we move along the sequence $U_1, \ldots, U_7$; more accurately, U_1 is independent of all subsequent values, U_2 has moderate dependence with all subsequent values, U_3 has a stronger level of dependence with all subsequent values, and so on, U_6 and U_7 being most strongly dependent. The Kendall's rank correlation values may be easily calculated using the formula $\rho_\tau = 1 - 1/\theta$ and the values are respectively $0, 0.5, 2/3, 0.75, 0.8$ and $5/6$.

The S code for the Gumbel case is very fast and has been included in the library QRMlib; see `http://www.math.ethz.ch/~mcneil/book/QRMlib.html`.

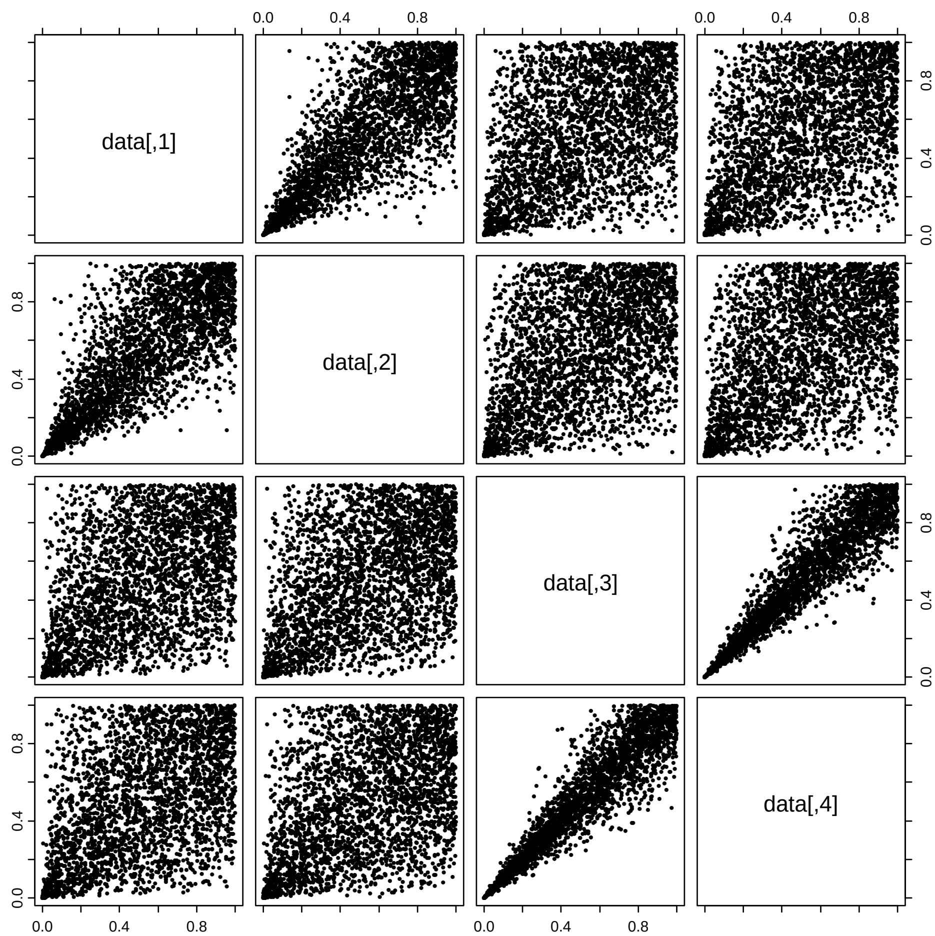

Figure 1: 3000 points from a 4-dimensional Clayton copula with structure (5). The generators have parameters $\theta_1 = 1$, $\theta_2 = 3$ and $\theta_3 = 8$.

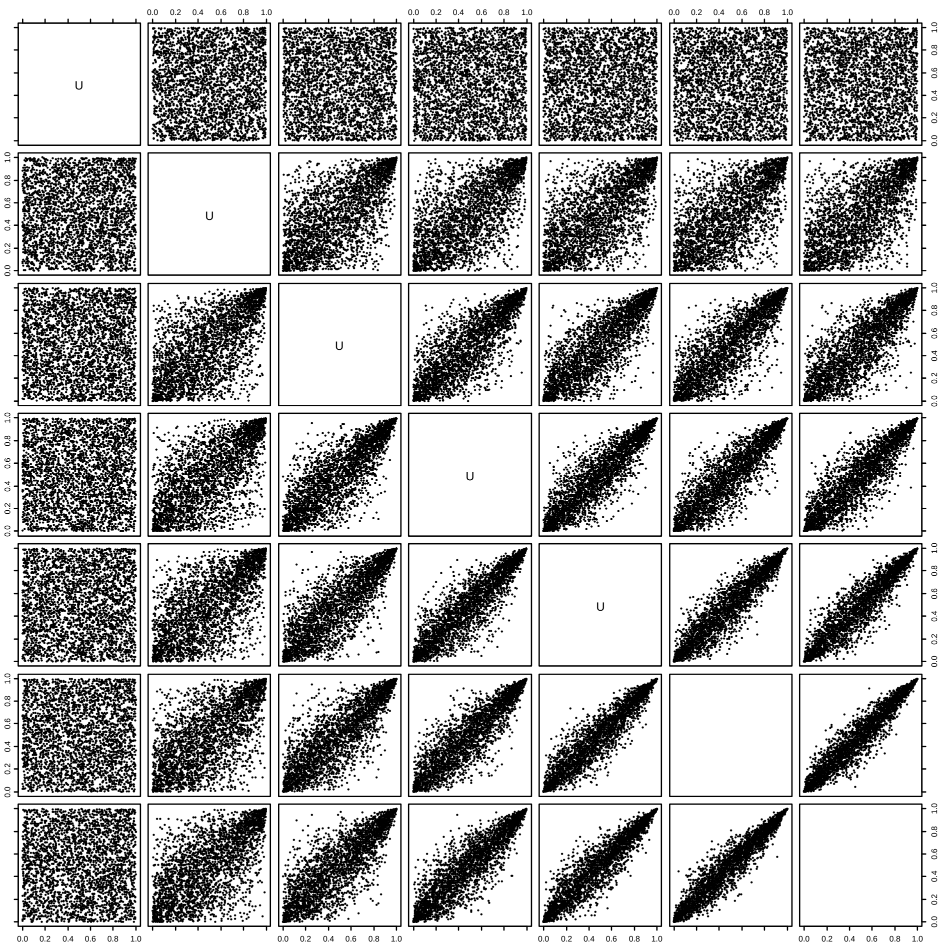

Figure 2: 3000 points from a 7-dimensional Gumbel copula with structure (6). The generators have parameters $\theta_i = i,\ i = 1, \ldots, 6$.

A Simulating Positive Stable Variates

The random variable X has an α-stable distribution, written $X \sim \mathrm{St}(\alpha, \beta, \gamma, \delta)$, if its characteristic function is

$$\psi(t) = E\exp(itX) = \begin{cases} \exp\left(-\gamma^\alpha |t|^\alpha \left(1 - i\beta \mathrm{sign}(t) \tan\frac{\pi\alpha}{2}\right) + i\delta t\right) & \alpha \neq 1, \\ \exp\left(-\gamma |t| \left(1 + i\beta \mathrm{sign}(t) \frac{2}{\pi} \ln|t|\right) + i\delta t\right) & \alpha = 1, \end{cases} \tag{14}$$

where $\alpha \in (0, 2]$, $\beta \in [-1, 1]$, $\gamma > 0$ and $\delta \in \mathbb{R}$. Note that there are various alternative parameterizations of the stable distributions and we use a parameterization of Nolan (see [10], Definition 1.8). The case $X \sim \mathrm{St}(\alpha, 1, \gamma, 0)$ for $\alpha < 1$ gives a distribution on the positive half-axis, which we refer to as a positive stable distribution. A simulation algorithm for a standardized variate $Z \sim \mathrm{St}(\alpha, \beta, 1, 0)$ is given in Nolan (see [10], Theorem 1.19). In the case $\alpha \neq 1$ then $X = \delta + \gamma Z$ has a $\mathrm{St}(\alpha, \beta, \gamma, \delta)$ distribution.

B S Code

```
rcopula.Gumbel2Gp <- function(n = 1000, gpsizes =c(2,2), theta =c(2,3,5))
{
  Y <- rstable(n,1/theta[1])*(cos(pi/(2*theta[1])))^theta[1]
  innerU1 <- rcopula.gumbel(n,theta[2]/theta[1],gpsizes[1])
  innerU2 <- rcopula.gumbel(n,theta[3]/theta[1],gpsizes[2])
  U <- cbind(innerU1,innerU2)
  Y <- matrix(Y, nrow = n, ncol = sum(gpsizes))
  out <- exp( - ( - log(U)/Y)^(1/theta[1]))
  out
}

########################################################################

rcopula.GumbelNested <- function(n, theta)
{
  d <- length(theta)+1
  if (d==2)
    out <- rcopula.gumbel(n,theta,d)
  else if (d>2){
    Y <- rstable(n,1/theta[1])*(cos(pi/(2*theta[1])))^theta[1]
    U <- runif(n)
    innerU <- rcopula.GumbelNested(n,theta[-1]/theta[1])
    U <- cbind(U,innerU)
    Y <- matrix(Y, nrow = n, ncol = d)
    out <- exp( - ( - log(U)/Y)^(1/theta[1]))
  }
```

```
  out
}

#########################################################################

rstable <- function(n,alpha,beta=1){
  t0 <- atan(beta*tan((pi * alpha)/2))/alpha
  Theta <- pi * (runif(n)-0.5)
  W <-  - log(runif(n))
  term1 <- sin(alpha*(t0+Theta))/(cos(alpha*t0)*cos(Theta))^(1/alpha)
  term2 <- ((cos(alpha*t0+(alpha-1)*Theta))/W)^((1-alpha)/alpha)
  term1*term2
}

#########################################################################
rAC <- function(name, n,d,theta) {
    illegalpar <- switch(name,
                         clayton=(theta<0),
                         gumbel=(theta<1))
    if (illegalpar) stop("Illegal parameter value")
    independence <- switch(name,
                           clayton=(theta==0),
                           gumbel=(theta==1))
    U <- runif(n*d)
    U <- matrix(U,nrow=n,ncol=d)
    if (independence) return(U)
    Y <- switch(name,
       clayton=rgamma(n,1/theta),
       gumbel=rstable(n,1/theta)*(cos(pi/(2*theta)))^theta)
    Y <- matrix(Y, nrow=n,ncol=d)
    psi <- switch(name,
      clayton=function(t,theta){(1+t)^(-1/theta)},
      gumbel=function(t,theta){exp(-t^(1/theta))})
    psi(-log(U)/Y,theta)
  }

#######################################################################

rcopula.gumbel <- function(n, theta, d){
  rAC("gumbel",n,d,theta)
}
```

References

[1] U. Cherubini, E. Luciano, and W. Vecchiato. *Copula Methods in Finance.* Wiley, Chichester, 2004.

[2] L. Devroye. On random variate generation when only moments or Fourier coefficients are known. *Mathematics and Computers in Simulation*, 31:71–89, 1989.

[3] L. Devroye. Algorithms for generating discrete random variables with a given generating function or a given moment sequence. *SIAM Journal on Scientific and Statistical Computing*, 12:107–126, 1999.

[4] W. Feller. *An Introduction to Probability Theory and Its Applications: Volume II.* Wiley, New York, 1971.

[5] Harry Joe. *Multivariate Models and Dependence Concepts.* Chapman & Hall, London, 1997.

[6] C.H. Kimberling. A probabilistic interpretation of complete monotonicity. *Aequationes Math.*, 10:152–164, 1974.

[7] A.W. Marshall and I. Olkin. Families of multivariate distributions. *J. Amer. Statist. Assoc.*, 83:834–841, 1988.

[8] A.J. McNeil, R. Frey, and P. Embrechts. *Quantitative Risk Management: Concepts, Techniques and Tools.* Princeton University Press, Princeton, 2005.

[9] R.B. Nelsen. *An Introduction to Copulas.* Springer, New York, 1999.

[10] J.P. Nolan. Stable distributions. Unpublished book manuscript, 2002.

[11] P.J. Schönbucher. *Credit Derivatives Pricing Models.* Wiley, 2003.

[12] B. Schweizer and A. Sklar. *Probabilistic Metric Spaces.* North-Holland/Elsevier, New York, 1983.

[13] N. Whelan. Sampling from Archimedean copulas. *Quantitative Finance*, 4(3):339–352, 2004.

[14] D.V. Widder. *The Laplace Transform.* Princeton University Press, Princeton, 1946.