Proc. of the European Safety and Reliability Conference (ESREL 2026)
Edited by José C. Matos, Paulo B. Lourenço, Michael Beer, Daniel Oliveira,
Edoardo Patelli, Hélder S. Sousa, Rui A. Silva, Monica Santamaria and Chiara Turco
© 2026 ESREL 2026 Organizers. *Published by* Research Publishing, Singapore.
ISBN: 978-981-94-6718-1 | doi: 10.3850/ESREL2026061419_esrel26-p26138-cd

Recent experience with the use of Chimera for vine copula modelling and future challenges

Oswaldo Morales-Nápoles
Hydraulic Engineering, Delft University of Technology, The Netherlands.
E-mail: o.moralesnapoles@tudelft.nl

Vine copulas are sequences of trees used to represent graphically multivariate probability distributions constructed from bivariate components. Chimera is the name of the catalogue developed at TU Delft, containing all possible tree sequences with up to 8 nodes. The catalogue is presented in a database consisting of more than 663 million matrices. In this paper, we succinctly review the development of Chimera, illustrate recent research using it for simulation studies, and highlight future research directions, also involving simulation studies. We conclude by discussing its potential significance for advancing risk and reliability analysis.

Keywords: vine copula, Chimera, risk, reliability.

1. Introduction

One of the requisites for a PhD graduation at the TU Delft is that the candidate should submit several propositions related to the research to be defended. These propositions should be defendable and, more importantly, opposable. One of the propositions proposed for defence in 2010 was: "The construction of a large data base with all possible labelled regular vines on n nodes for a sufficiently large n that includes all known graphical properties is, at this stage, as important for applications as algorithms for generating them". This database, which was named Chimera (see Fig. 1), was finally presented in Morales-Nápoles et al. (2023). Since 2010, the belief that Chimera constitutes an essential tool to continue the study of vine-copulas has not only remained unchanged but has been further reinforced.

Some of the main motivations to build Chimera are: (i) the number of regular vines grows extremely quickly with the number of nodes (variables), and the availability of substantial number of regular vines for practitioners and researchers is advantageous, (ii) The graphical properties of regular-vines and their relation to the multivariate distributions they realize remains underinvestigated, and Chimera can be used to addressing this gap systematically, (iii) heuristics, algorithms, and hypotheses from either simplified and non-simplified vine copulas can be "experimentally" tested using Chimera. These three arguments are not mutually exclusive nor exhaustive. They provide context regarding the conception and potential of Chimera.



Fig. 1. Logo of Chimera.

Throughout the remainder of this paper, the three points outlined above will be addressed. The paper starts with some basic concepts to make the reading self-standing. Simulation studies recently conducted at the TU Delft using Chimera are illustrated and future research directions discussed.

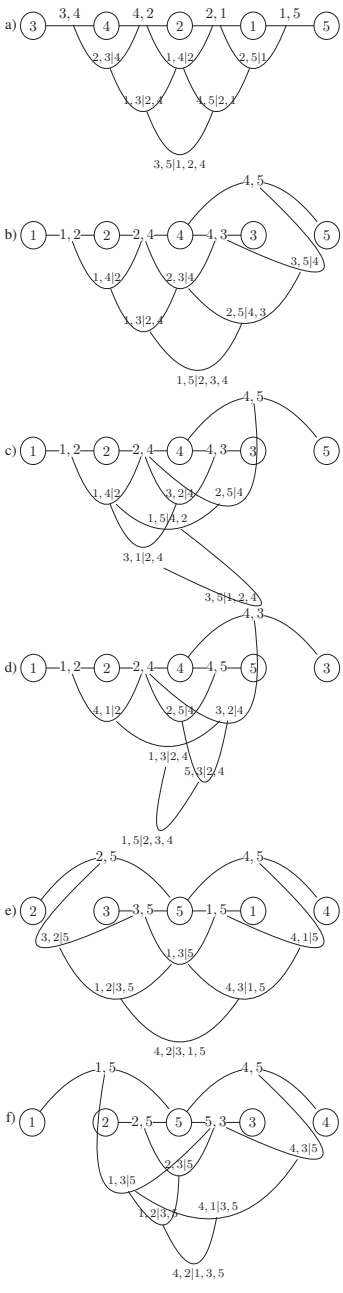


Fig. 2. Six examples of regular vines for each equivalence class; a) corresponds to a D-vine; T_1 is isomorphic for b), c) and d); c) and d) are tree-equivalent but in different equivalence classes, T_1 is isomorphic for e) and f); f) is a C-vine.

2. Basic concepts

In this section basic concepts are introduced in an informal way. For a formal treatment, the reader is referred to Joe (2014) or Czado (2019). A *tree* is a connected graph with no cycles. A *vine* is a sequence of trees $T_1, \ldots, T_{n-1}$ such that the edges of T_i become nodes of T_{i+1} for $i = 1, \ldots, n-1$. If additionally, all edges in T_i connected as nodes in T_{i+1} are adjacent to a common node in T_i then this is a *regular vine*. For an edge in a vine, the *constraint set* is the set of nodes that the edge is incident to, and which constrain future edge selection. The *conditioned set* is the intersection (pair) of nodes of two constraint sets, and the *conditioning set* is the remainder of the constraint set that provides the context for their conditional dependence. In Fig. 2, six regular vines are presented. The conditioned set is presented to the left of the vertical line and the conditioning set to the right

When $T_1, \ldots, T_{n-1}$ are *isomorphic*, that is, of the same type of tree structure in different regular vines they are *tree-equivalent*. When vines can be transformed into one another by permutation they are in the same *equivalence class*. Fig. 2 presents the six non-equivalent classes of regular vines on 5 variables. Notice that 2a) corresponds to a D-vine; 2c) and 2d) are tree-equivalent but in different equivalence classes, and 2f) is a C-vine. The number of regular vines per equivalence class is presented in Table 1.

Table 1. Number of regular vines per equivalence class.

Equivalence class	Number of regular vines
a)	60
b)	120
c)	60
d)	120
e)	60
f)	60

A *vine copula* is a regular vine in which each edge is associated with a bi-variate copula, representing the dependence between the two variables

in the conditioned set, conditional on the variables in the conditioning set. Hence, it is a way to build a multivariate probability distribution using pairwise dependencies. Regular vines are represented by a matrix. A *regular vine matrix* is an upper-triangular matrix that encodes the structure of a regular vine, specifying the order of variables and the sequence of conditional dependencies in the vine. The diagonal elements $m_{i,i}$ represent the variables in a particular order while upper-triangular elements $m_{i,j}$, $(i < j)$ determine the conditioning and conditioned sets of the pair copulas in the vine. For example, Matrix M below is a vine copula matrix for the regular vine in Fig.2c). Where, $(m_{2,2}, m_{1,2}) = (4, 5)$; $(m_{3,3}, m_{1,3}) = (2, 4)$; $(m_{4,4}, m_{1,4}) = (1, 2)$ and $(m_{5,5}, m_{1,5}) = (3, 4)$ represent T_1. $(m_{3,3}, m_{2,3}|m_{1,3}) = (2, 5|4)$; $(m_{4,4}, m_{2,4}|m_{1,4}) = (1, 4|2)$; and $(m_{5,5}, m_{2,5}|m_{1,5}) = (3, 2|4)$ represent T_2. $(m_{4,4}, m_{3,4}|m_{2,4}, m_{1,4}) = (1, 5|4, 2)$ and $(m_{5,5}, m_{3,5}|m_{2,5}, m_{1,5}) = (3, 1|2, 4)$ constitute T_3 and T_4 has as single edge $(m_{5,5}, m_{4,5}|m_{3,5}, m_{2,5}, m_{1,5}) = (3, 5|1, 2, 4)$.

$$M = \begin{pmatrix} 5 & 5 & 4 & 2 & 4 \\ & 4 & 5 & 4 & 2 \\ & & 2 & 5 & 1 \\ & & & 1 & 5 \\ & & & & 3 \end{pmatrix} \tag{1}$$

Finally, it is important to remark that, in general, the copulas attached to regular vines are parametrized by rank correlations and the conditional copulas are assumed to be constant for all values of the conditioning variables. That is:

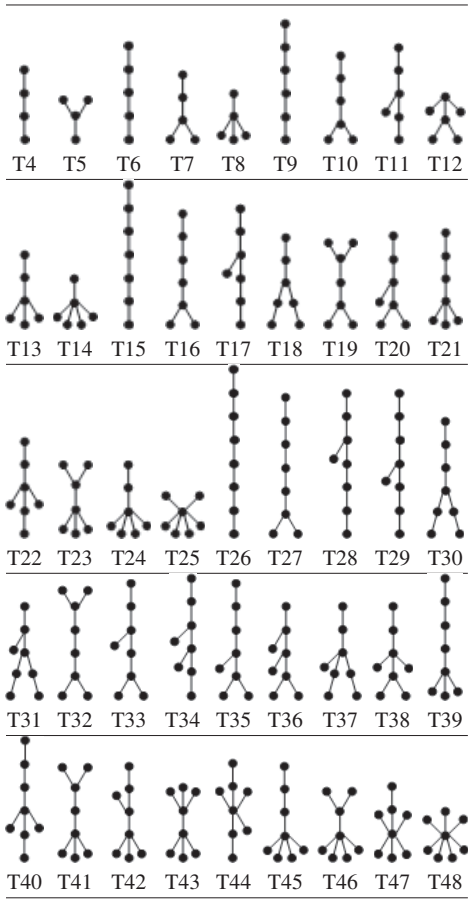
$$C_{F_{X|\mathbf{y}},\, F_{Z|\mathbf{y}}}\big(F_{X|\mathbf{y}}(x),\, F_{Z|\mathbf{y}}(z) \mid \mathbf{y}\big) = C_{F_{X|\mathbf{y}},\, F_{Z|\mathbf{y}}}\big(F_{X|\mathbf{y}}(x),\, F_{Z|\mathbf{y}}(z)\big) \tag{2}$$

When vine copulas are treated under the assumption (or decision according to Cooke and Bedford (2025)) above, they are referred to as *simplified vine copulas* or simply vine copulas as stated earlier. When the assumption is relaxed, they are called *non-simplified vine copulas*.

3. Recent research

3.1. *Chimera*

Table 2. Non-isomorphic trees on 4, 5, 6, 7 and 8 nodes and their labeling in Chimera.



As mentioned earlier, Chimera (Morales-Nápoles et al. (2023)) is a catalogue that presents all 24 4×4; 480 5×5; 23,040 6×6; 2,580,480 7×7; and 660,602,880 8×8 matrices representing regular vines on 8 nodes. Matrices are presented as data structures for R, Matlab and Python. They are classified according to their tree equivalent class using the notation indicated in Table 2 for non-isomorphic trees. For example, all vines like in Fig. 2a) would be classified in Chimera as T6+T4; 2b) as T7+T4; 2c) and 2d) as T7+T5 (they are tree

equivalent but in different equivalence classes); 2e) as T8+T4; and 2f) as T8+T5. With the use of Chimera and High-performance computing (using the DelftBlue supercomputer in our case), we can systematically investigate different properties of vine copulas. This will be illustrated next.

3.2. *Simulation with AIC*

A criterion often used when selecting a vine copula to fit to data is Akaike's information criterion.

$$AIC_j = -2\sum_{k=1}^{N} \ln\left(\mathcal{L}\left(\hat{\theta}\left(\mathcal{B}(\mathcal{V}_j); u_k\right)\right)\right) + 2K \tag{3}$$

Where $\mathcal{L}\left(\hat{\theta}\left(\mathcal{B}(\mathcal{V}_j); u_k\right)\right)$ is the likelihood contribution of the model specified by the regular vine $\mathcal{V}$ with copula specification $\mathcal{B}$ and parameters $\hat{\theta}$. One of the most widely used estimation methods for vine copulas is the stepwise semi-parametric estimator. Roughly, all marginal parameters are handled separately, then the parameters of the vine copula are estimated in every tree conditioning on the parameters of the preceding levels of the structure. For a particular data set on up to eight variables, with the use of Chimera, we can fit all possible vine copulas using the stepwise semi parametric estimator (Aas et al., 2009). The best possible vine copula would be the one with lowest (most negative) AIC value. Where $\mathcal{L}\left(\hat{\theta}\left(\mathcal{B}(\mathcal{V}_j); u_k\right)\right)$ is the likelihood contribution of the model specified by the regular vine $\mathcal{V}_j$ with copula specification $\mathcal{B}$ and parameters $\hat{\theta}$. One of the most widely used estimation methods for vine copulas is the stepwise semi-parametric estimator. Roughly, all marginal parameters are handled separately, then the parameters of the vine copula are estimated in every tree conditioning on the parameters of the preceding levels of the structure. For a particular data set on up to eight variables, with the use of Chimera, we can fit all possible vine copulas using the stepwise semi parametric estimator (Aas et al., 2009). The best possible vine copula would be the one with lowest (most negative) AIC value.

The computational costs of fitting all possible vine copulas to data (especially above 7 nodes) are considerable. Several algorithms have been proposed to speed up the fitting procedure. The most common one is known as Dißmann's algorithm (Dißmann et al., 2013). This algorithm is based on a maximum spanning tree (MST) search. The first tree in the vine copula is selected based on the strongest pairwise dependencies as judged by rank correlations. Subsequent trees are also selected based on a maximum spanning tree search also based on (partial) rank correlations. Naturally, Dißmann's algorithm is much faster computationally than a brute force approach. One possible simulation experiment to perform with the aid of Chimera is:

(1) Generate a sample of 300 or 1000 observations from the vine represented by matrix M above, with Clayton copulas for Spearman's rank correlation equal to 0.8 attached to every level of the vine.
(2) Fix the regular vine, fix the choice of copula (to Clayton) and estimate the parameters $\hat{\theta}$ Compute AIC for this vine copula (AIC_T).
(3) Fit all possible 480 vine copulas using the stepwise semi-parametric estimator and select from the 480 vine copulas the one with smallest AIC. Call this the brute force solution $AIC_{BF} = \min\{AIC_1, \ldots, AIC_{480}\}$
(4) Estimate another vine copula through Dißmann's algorithm (AIC_A).
(5) Compute $\Delta_{BF} = |AIC_T - AIC_{BF}|$ and $\Delta_A = |AIC_T - AIC_A|$. Naturally, the closer Δ_{BF} and Δ_A are to zero, the better the goodness of fit would be expected for the model.
(6) Repeat steps 1 to 5, say 100 times.

The results of implementing the simulation experiment above are presented in Figure 3. All calculations are performed using Coblenz (2021).

The 90th percentile of Δ_{BF} with 300 samples is 46, while for Δ_A, the same percentile corresponds to 616. Also, 93% of the times, the regular vine corresponding to M is selected as the one with smallest AIC among the 480 possible regular vine structures. For 1000 samples the 90th percentile of Δ_{BF} is 113 while the corresponding one for Δ_A is 2144. In the case of simulations with 1000 samples, all 100 times M is selected as the one with smallest AIC among the 480

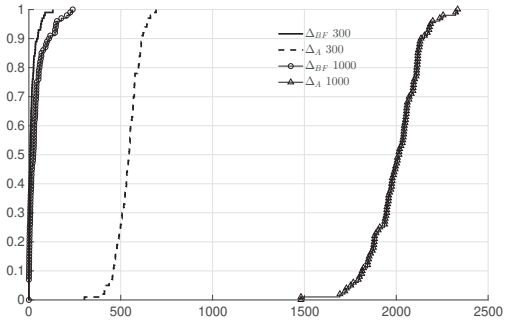


Fig. 3. Empirical distribution of Δ_{BF} and Δ_A for 300 and 1000 samples. Spearman's rank correlation equal to 0.8 with Clayton copulas along the vine-copula.

possible regular vine structures. Clearly, a brute force approach when asymmetries in the bivariate margins and high correlations are observed in the data is strongly preferred to the algorithm.

In Morales-Nápoles et al. (2026), a large-scale study with over 1.4 billion vine copulas fit to synthetic and real data like the one briefly presented in this section is described. The study is performed for 3 sample sizes, 30, 300 and 1000, three different levels of correlation, low, medium and high (0.2, 0.6 and 0.8 Spearman's coefficient) for Gaussian, Clayton and Gumbel copulas.

3.3. *Hypothesis testing*

One drawback of the approach introduced in previous section is that it does not perform a formal hypothesis test. For vine copulas, one of the most used hypotheses tests is the so-called Vuong test (Vuong, 1989). This test is based on the likelihood ratio. The unadjusted test is based on the statistic ν.

$$\nu = \frac{N^{0.5}\mathcal{LR}}{\hat{\omega}^2} \tag{4}$$

where, $\mathcal{LR} = \sum_{k=1}^{N} \log \frac{\mathcal{L}\left(\hat{\theta}(\mathcal{B}(\mathcal{V}_s);u_k)\right)}{\mathcal{L}\left(\hat{\theta}(\mathcal{B}(\mathcal{V}_j);u_k)\right)}$, and,

$$\hat{\omega}^2 = \frac{1}{N}\sum_{k=1}^{N}\left(\log\frac{\mathcal{L}\left(\hat{\theta}(\mathcal{B}(\mathcal{V}_s);u_k)\right)}{\mathcal{L}\left(\hat{\theta}(\mathcal{B}(\mathcal{V}_j);u_k)\right)}\right)^2 - \left(\frac{1}{N}\sum_{k=1}^{N}\log\frac{\mathcal{L}\left(\hat{\theta}(\mathcal{B}(\mathcal{V}_s);u_k)\right)}{\mathcal{L}\left(\hat{\theta}(\mathcal{B}(\mathcal{V}_j);u_k)\right)}\right)^2$$

The null hypothesis is that the models are equivalent. $\mathcal{H}_0 : \mathcal{B}(\mathcal{V}_s) \equiv \mathcal{B}(\mathcal{V}_j)$. One chooses a value c from the standard normal distribution for some significance level. If the value of the statistic is higher than c then one rejects the null hypothesis that the models are equivalent in favor of $\mathcal{B}(\mathcal{V}_s)$ being better than $\mathcal{B}(\mathcal{V}_j)$. If ν is smaller than $-c$ then one rejects the null hypothesis in favor of $\mathcal{B}(\mathcal{V}_j)$ being better than $\mathcal{B}(\mathcal{V}_s)$. Finally, if $\nu < c$, then one cannot discriminate between the two competing models given the data. The outcome of the test is typically coded as 1, 2 or 0 respectively depending on the cases above.

Table 3. Summary of results of 100 simulations of Vuong's test (Brute Force) with 300 and 1000 samples from the vine represented by matrix M above, with Clayton copulas for Spearman's rank correlation equal to 0.8 attached to every level of the vine.

$\mathcal{H}_0 : \mathcal{B}(\mathcal{V}_s) \equiv \mathcal{B}(\mathcal{V}_j)$	300 samples	1000 samples	M in 300 samples	M in 1000 samples
0	19	58	-	-
1	40	42	38	41
2	13		9	-
3	18		18	-
4	3		3	-
5	1		1	-
6	5		5	-
104	1		1	-

Suppose that in the simulation explained in the previous section, we apply Vuong's test, instead of computing AIC differences. The results are summarized in Table 3 for the brute force procedure. As may be seen in Table 3, for 300 samples, 19 times out of a 100 Vuong's test could not identify

a model as equivalent to the truth. When it does identify one model as equivalent, 38 times out of 40 it correctly identifies the model described by M. When 2 models are identified as equivalent, then 9 times out of 13 M is correctly identified along with another model. Other values in columns two and four can be interpreted similarly. Interestingly, in one of the hundred simulations, 104 vine copulas out of 480 were identified as equivalent to the "truth" which may have been caused by a "bad" sample. Notice also that with 300 samples, in 81 simulations Vuong's test will identify at least one model as equivalent to the true model and in 75 of these M is correctly identified.

When moving to 1000 samples, 42 times out of a hundred, the test identifies one model (out of 480 possible) as equivalent to the true, of which 41 correspond to M.

These results together with those from previous section suggest that Vuong's test (assuming strictly non-nested models) may be too restrictive for model identification, while a brute force procedure with minimum AIC is likely to capture the properties of the multivariate model.

4. Applications in Risk and Reliability

Three different concepts frequently used in applications related to risk and reliability are briefly discussed: *limit state*, *conditional quantiles* and *conditional expectations*. These are discussed in the context of the true model, the brute force solution, Dißmann's algorithm and a "valid" vine copula according to Vong's test which is not the brute force solution. The concepts briefly treated in this section have important implications in risk and reliability analysis. See for example Mares-Nasarre et al. (2024) and Paprotny et al. (2025) for recent applications where these concepts show relevance.

We use 300 samples from matrix M. The brute force procedure, finds matrix M to be a valid model. The copula families associated with M are presented in C_M. Additionally to M, M_2 is found to be a valid matrix according to Vuong's test. The copula families are given by C_2. M_2 is in fact vine d) in Figure 2. Dißmann's algorithm outputs the vine copula matrix M_D and copula families C_D below. M_D is in the equivalence class b) in Figure 2. According to Vuong's test, the fit would not be equivalent to the true model. The parameters estimated for each case are also shown.

$$C_M = \begin{pmatrix} \text{Clayton} & \text{Clayton} & \text{Clayton} & \text{Clayton} \\ & \text{Clayton} & \text{Clayton} & \text{Clayton} \\ & & \text{Clayton} & \text{surjoe} \\ & & & \text{Clayton} \end{pmatrix}$$

$$\hat{\theta}_M = \begin{pmatrix} 3.1385 & 3.3721 & 3.3012 & 3.1521 \\ & 3.0189 & 3.0310 & 2.9493 \\ & & 3.3701 & 4.1413 \\ & & & 3.7939 \end{pmatrix}$$

$$M_2 = \begin{pmatrix} 5 & 5 & 4 & 4 & 2 \\ & 4 & 5 & 2 & 4 \\ & & 2 & 5 & 3 \\ & & & 3 & 5 \\ & & & & 1 \end{pmatrix}$$

$$C_2 = \begin{pmatrix} \text{Clayton} & \text{Clayton} & \text{Clayton} & \text{Clayton} \\ & \text{Clayton} & \text{Clayton} & \text{Clayton} \\ & & \text{surjoe} & \text{surjoe} \\ & & & \text{surjoe} \end{pmatrix}$$

$$\hat{\theta}_2 = \begin{pmatrix} 3.1385 & 3.2721 & 3.1521 & 3.3012 \\ & 3.0189 & 2.9493 & 3.0310 \\ & & 8.9479 & 4.1413 \\ & & & 1.3621 \end{pmatrix}$$

$$M_D = \begin{pmatrix} 5 & 5 & 3 & 1 & 3 \\ & 3 & 5 & 3 & 5 \\ & & 1 & 5 & 1 \\ & & & 4 & 4 \\ & & & & 2 \end{pmatrix}$$

$$C_D = \begin{pmatrix} \text{Plackett} & \text{Clayton} & \text{SurJoe} & \text{SurJoe} \\ & \text{SurJoe} & \text{amhaq} & \text{SurJoe} \\ & & \text{Ind} & \text{amhaq} \\ & & & \text{Frank} \end{pmatrix}$$

$$\hat{\theta}_D = \begin{pmatrix} 1.3389e3 & 4.3684 & 7.5096 & 7.5856 \\ & 1.2643 & -1.0000 & 1.3369 \\ & & 0 & -0.9999 \\ & & & 4.8828 \end{pmatrix}$$

4.1. *Limit state*

A limit state is an integral over a failure region. In order to illustrate this idea we compute $P_1 = P(X_1 \leq x_{1,.05} \cap \ldots \cap X_5 \leq x_{5,.05})$, $P_2 = P(X_1 \leq x_{1,.05} \cup \ldots \cup X_5 \leq x_{5,.05})$, $P_3 = P(X_1 > x_{1,.95} \cap \ldots \cap X_5 > x_{5,.95})$ and

$P_4 = P(X_1 > x_{1,.95} \cup \ldots \cup X_5 > x_{5,.95})$ for the cases of interest. 1 000 000 samples are used for each case.

Table 4. P_1, P_2, P_3 and P_4 for the cases of interest.

Cases	P_1	P_2	P_3	P_4
M, Clayton, 3.1819	0.03446	0.06290	0.00341	0.15218
$M, C_M, \hat{\theta}_M$	0.03467	0.06279	0.00334	0.15110
$M_2, C_2, \hat{\theta}_2$	0.03489	0.06283	0.00329	0.15213
$M_D, C_D, \hat{\theta}_D$	0.03489	0.06618	0.00062	0.15472

According to Table 4 the only probability severely underestimated (by a factor 5.3 approximately) is P_3 for the Dißmann's solution. All other probabilities in the example would be more or less well approximated by the different models.

4.2. *Conditional distribution*

In applications, for prediction, the expectation of such conditional distributions is required. Other type of inference, very common in applications, requires extrapolation of exceedance probabilities in the order of 10^{-3} or lower. Figure 4 shows $P(X_1 > x_1 | X_2 > x_{2,.95}, \ldots, X_5 > x_{5,.95})$ for all four cases under consideration. Notice that since this is a hypothetical example, the non-conditional margins are $U(0,1)$

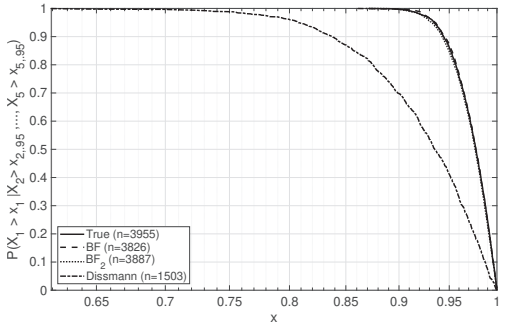


Fig. 4. $P(X_1 > x_1 | X_2 > x_{2,.95}, \ldots, X_5 > x_{5,.95})$

Clearly, for the case presented in Figure 4, the conditional expectation and conditional quantiles will be badly estimated by the Dißmann's solution. This is in line with the observations from previous section. This effect may be further amplified depending on the actual marginal distributions of the variables of interest.

5. Final comments and conclusion

This paper presents briefly, recent experience at TU Delft with the use of Chimera in simulation experiments to better understand multidimensional distributions constructed with simplified vine copulas and its significance for applications in risk and reliability.

Simplified vine copulas are attractive for applications in science and engineering (in risk and reliability in particular) because of their flexibility and their relatively lower parametric nature. Based on the results of large scale simulation studies and results briefly discussed in this paper we recommend the use of a brute force approach for fitting simplified vines to data using Chimera for up to 8 variables. Furthermore, this paper advocates the use of Chimera to better understand vine copulas including algorithms for estimating non-simplified vine copulas. There is still much to be learned about vine copulas and Chimera is a valuable tool to tackle this learning process.

Acknowledgement

The author is grateful with Marcel 't Hart and Özge Şahin for discussions around the subjects treated in this paper.

References

Aas, K., C. Czado, A. Frigessi, and H. Bakken (2009). Pair-copula constructions of multiple dependence. *Insurance: Mathematics and Economics 44*(2), 182–198.

Coblenz, M. (2021). Matvines: A vine copula package for matlab. *SoftwareX 14*, 100700.

Cooke, R. and T. Bedford (2025). Identifiability and the simplifying assumption. Working Paper, Delft University of Technology.

Czado, C. (2019). *Analyzing dependent data with vine copulas. A Practical Guide With R*. Springer Cham.

Dißmann, J., E. Brechmann, C. Czado, and D. Kurowicka (2013). Selecting and estimating regular vine copulae and application to financial returns. *Computational Statistics & Data Analysis 59*, 52–69.

Joe, H. (2014). *Dependence modeling with copulas*. CRC Press.

Mares-Nasarre, P., A. Muscalus, K. Haas, and O. Morales-Nápoles (2024). The probabilistic dependence of ship-induced waves is preserved spatially and temporally in the savannah river (usa). *Scientific Reports 14*, 6155–6184.

Morales-Nápoles, O., M. Rajabi-Bahaabadi, G. A. Torres-Alves, and C. M. P. 't Hart (2023). Chimera: An atlas of regular vines on up to 8 nodes. *Scientific Data 10*(1).

Morales-Nápoles, O., C. M. P. 't Hart, G. A. Torres-Alves, E. Perrone, and E. Ragno (2026). Brute-force and maximum weight spanning tree algorithm fitting of regular vines on up to 8 nodes. Working Paper, Delft University of Technology.

Paprotny, D., C. M. P. 't Hart, and O. Morales-Nápoles (2025). Evolution of flood protection levels and flood vulnerability in europe since 1950 estimated with vine-copula models. *Natural Hazards 121*, 6155–6184.

Vuong, Q. H. (1989). Likelihood ratio tests for model selection and non-nested hypotheses. *Econometrica 57*(2), 307–333.