# A Survey of
# Estimation of Distribution Algorithms
# Based on Copulas

Yasser González-Fernández, Marta Soto
Institute of Cybernetics, Mathematics and Physics, Cuba.
Email: {ygf, mrosa}@icimaf.cu.

**Abstract**

The use of probabilistic models based on copulas in Estimation of Distribution Algorithms (EDAs) has been identified as an emerging research trend on these algorithms for continuous domains. By using copulas, the effect of the dependence structure and the margins in a joint distribution can be represented separately. Consequently, EDAs based on copulas inherit these characteristics and are able to build flexible search distributions. This paper presents a survey of the EDAs based on copulas that have been proposed in the literature between 2007 and 2012. We also identify different aspects that, in our opinion, should be considered in order to attain a deeper understanding of EDAs based on copulas.

## 1 Introduction

Estimation of Distribution Algorithms (EDAs) [44, 51] are evolutionary optimization methods characterized by the explicit use of probabilistic models. These algorithms explore the search space by iteratively estimating and sampling a probability distribution (search distribution) built from promising solutions. In the recent years, several EDAs that build the search distributions through probabilistic models based on copulas have been proposed in the literature. As evidence of this increasing popularity, the use of copulas in EDAs has been identified as an emerging approach for the solution of real-valued optimization problems [34].

The use of copula-based models in continuous EDAs places these algorithms in an advantageous position in comparison with other EDAs that rely on the assumption of a particular multivariate distribution, such as the multivariate normal distribution. By means of copulas [39, 52], any multivariate distribution can be decomposed into the marginal distributions and a copula that determines the dependence structure between the variables. Moreover, by means of copula-based factorizations, it is possible to construct multivariate distributions that combine a rich variety of dependence patterns and it also provide us with a potential mechanism to tackle the curse of dimensionality. EDAs based on copulas inherit these characteristics and are able to build flexible search distributions.

The papers is organized as follow. Section 2 gives an overview on different aspects of copula theory that are relevant to this work. Section 3 presents the general procedure of an EDA, while a description of the copula-based EDAs that have been proposed in the literature is presented in Section 4. Finally, Section 5 gives the conclusions.

## 2 Overview on Copula Theory

The notion of copulas separates the effect of the dependence structure and the margins in a joint distribution. This definition is supported by Sklar's Theorem [63, 64], which is defined as follows. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a vector of continuous random variables with joint density function $f$ and joint cumulative distribution function $F$, both defined on $\mathbb{R}^n$. Also let $\mathbf{x} = (x_1, \ldots, x_n)$ be an observation of $\mathbf{X}$ and $F_1, \ldots, F_n$ denote the cumulative univariate margins of $F$. Sklar's Theorem states that the copula $C : [0,1]^n \to [0,1]$ associated with $F$ is a distribution function with uniform margins that satisfies

$$F(x_1, \ldots, x_n) = C(F(x_1), \ldots, F(x_n)),$$

and consequently,

$$C(u_1, \ldots, u_n) = F\left(F_1^{(-1)}(u_1), \ldots, F_n^{(-1)}(u_n)\right), \tag{1}$$

where $F_1^{(-1)}, \ldots, F_n^{(-1)}$ denote the pseudo-inverses of the margins $F_1, \ldots, F_n$. If $F$ is continuous then the copula $C(u_1, \ldots, u_n)$ is unique.

A wide range copula families have been presented in the literature, being popular choices among practitioners the members of the elliptical and Archimedean families. Elliptical copulas are obtained from elliptical distributions, such as the normal and $t$ distributions, by using (1). The multivariate $t$ and normal copulas are obtained in this way and they allow the construction of multivariate distributions with different margins, but preserving the multivariate $t$ or normal dependence structure. Archimedean copulas are defined in terms of a function $\varphi : (0,1] \to [0,\infty)$ called *generator*, which is convex and strictly decreasing with positive second derivative such that $\varphi(1) = 0$. A bivariate copula is said to be Archimedean if it admits the representation

$$C(u_1, u_2) = \varphi(\varphi^{-1}(u_1) + \varphi^{-1}(u_2)), \tag{2}$$

where $\varphi^{-1}$ denotes the pseudo-inverse of some Archimedean generator $\varphi$. This generator usually depends on a parameter that controls the strength of the dependence between the variables and different Archimedean copulas are obtained by using different generator functions. Popular choices include Clayton, Gumbel, Frank, Ali-Mikhail-Haq, and Farlie-Gumbel-Morgenstern copulas.

The estimation of a copula-based multivariate distribution involves the estimation of both the copula and the marginal distributions [14, 15]. The process of fitting the parameters often proceeds in two phases: first estimating the parameters of the marginal densities, and then estimating the parameters of the copula. The estimation of the parameters of the copula is usually carried out by maximum likelihood [24, 40] or the method of moments [60]. Also, a non-parametric estimation of the copula can be performed through the empirical copula [19]. Copulas can be simulated through the conditional distribution method [20] but there also exist algorithms for the simulation specific families, such as Archimedean copulas [49] and normal copulas [65].

For more information on dependence modeling with copulas, and a list of the copula families available, see the comprehensive books [39, 52]. In the following two sections, examples of multivariate copulas and dependence models that rely on lower-dimensional copulas as building blocks are described in more detail.

## 2.1 Multivariate Copulas

A consequence of Sklar's Theorem is that random variables are independent if and only if their underlying copula is the product or independence copula, which is given by

$$C\left(u_1, \ldots, u_n\right) = u_1 \cdot \ldots \cdot u_n.$$

This means that a multivariate distribution with independent margins can be represented through the marginal distributions and the product copula. Another important copula is the multivariate normal copula, which is a member of the family of elliptical copulas and is defined as follows,

$$C\left(u_1, \ldots, u_n; R\right) = \Phi_R\left(\Phi^{-1}\left(u_1\right), \ldots, \Phi^{-1}\left(u_n\right)\right),$$

where $\Phi_R$ is the standard multivariate normal distribution function with positive-definite correlation matrix $R$, and $\Phi^{-1}$ denotes the inverse of the standard univariate normal distribution.

Archimedean copulas, as presented in (2), share many nice properties but most of the members of this family are only defined for two variables. One way of extending bivariate Archimedean copulas to higher dimensions is by using exchangeable Archimedean copulas [10, 33] given by

$$C(u_1, \ldots, u_n) = \varphi(\varphi^{-1}(u_1) + \ldots + \varphi^{-1}(u_n)), \tag{3}$$

where $\varphi$ denotes the generator and $\varphi^{-1}$ its pseudo-inverse. These copulas usually have only one parameter that characterize the dependence structure between the variables regardless of the dimension.

## 2.2 Copula Factorizations

The use of multivariate copulas to model the dependence structure offers several advantages over the use of a multivariate normal distribution, but it presents some limitations. The number of tractable copulas when more than two variables are involved is rather limited and multivariate copulas may not be appropriate when all pairs of variables do not share the same type of dependence. Another shortcoming is that multivariate extensions, such as exchangeable Archimedean copulas or the multivariate $t$ copula, have only one parameter to describe certain aspects of the overall dependence. This situation has motivated the development of more appropriate multivariate copula-based dependence models. In the rest of this section, we briefly describe two of these models, which are based on lower-dimensional copulas as building blocks: pair-copula constructions and nested Archimedean copulas.

### 2.2.1 Pair-Copula Constructions

By using pair-copula constructions [1, 8, 9, 38], it is possible to build multivariate dependence models in terms of bivariate copulas. These models are represented through a graphical model called vine. A vine on $n$ variables is a set of nested trees $T_1, \ldots, T_{n-1}$, where the edges of tree $T_j$ are the nodes of the tree $T_{j+1}$ with $j = 1, \ldots, n-2$. C-vines (canonical vines) and D-vines (drawable vines) are two types of vines, each of which determine a specific decomposition of the multivariate density. In a C-vine, each tree $T_j$ has a unique root node that is connected to $n - j$ edges. In a D-vine, no node is connected to more than two edges. The edges of these trees represent the copulas in the decomposition. Since the bivariate copulas can belong belong to different families, vines can model a rich variety of dependences.

The estimation procedures of C-vines and D-vines [1, 36] consist of the following main steps: the selection of the structure of the vines, and the choice of the bivariate copulas in the factorization along with the estimation of their parameters. Simulation from vines [7, 8, 43] can be performed by following the conditional distribution method. For more information on dependence modeling with vines see [17, 42].

### 2.2.2 Nested Archimedean Copulas

The construction of exchangeable Archimedean copulas, as defined in (3), results in a very restrictive dependence structure: the distribution is invariant under permutations, and in many cases the dependence structure is determined by a single scalar parameter. Nested Archimedean copulas [39] provide an alternative to build more flexible multivariate Archimedean copulas by using nested generators. At each nesting level, an exchangeable Archimedean copula that determines the dependence between a group of variables is added in order to conform the overall structure. This flexibility comes with some restrictions that most be satisfied for the resulting multivariate distribution to be a proper copula.

Various models based on different nesting structures has been proposed in the literature [10]. Among them, the hierarchically nested Archimedean copulas constitute one of the most flexible models. The parameters of the nested Archimedean copulas may be estimated by maximum likelihood. However, it is not straightforward to derive a density for all parametric families, and the calculation of numerical derivatives may be necessary. Simulating from nested Archimedean copulas is also not simple. A procedure based on Laplace-transforms method is available [50], but it is not applicable to neither all families of Archimedean copulas nor all dimensions. Therefore, simulation is usually carried out by using conditional distribution method, for which it may also necessary to calculate numerical derivatives. For more information on nested Archimedean copulas see [37].

## 3 Estimation of Distribution Algorithms

EDAs explore the search space by iteratively estimating and sampling a probability distribution built from promising solutions. The basic steps of this iterative procedure are outlined in Algorithm 1. Each repetition of this procedure is defined as one generation.

The first step of an EDA is to simulate an initial population of solutions, which is usually generated randomly. The simulated population is scored according to the value of the objective function and then, a group of promising solutions are selected according to a selection method (*e.g.* truncation, tournament). The estimation and simulation (also called learning and sampling) of the search distributions are essential steps of an EDA. A probabilistic model of the selected population is built, and then a new population is generated by sampling the estimated model. This simulated population constitutes the current population. If elitism is used, a small set of the best points of the previous population is incorporated into the current population. Finally, the algorithm stops when certain criteria are satisfied; for example, to reach a maximum number of function evaluations or a value of the objective function.

For continuous domains, the normal distribution has been commonly adopted to model search distributions in EDAs [11, 41]. Nevertheless, this distribution can be inconsistent

```
$i \leftarrow 1$
repeat
    if $i == 1$ then
        Generate an initial population $P_1$.
        Evaluate the solutions in the population $P_1$ using the fitness function.
    else
        Select a population $P_i^{Selected}$ from $P_{i-1}$ according to a selection method.
        Estimate a probabilistic model $M_i$ of $P_i^{Selected}$.
        Sample a new population $P_i^{Sampled}$ from $M_i$.
        Evaluate the solutions in the population $P_i^{Sampled}$ using the fitness function.
        Create a population $P_i$ from $P_i^{Sampled}$ and $P_{i-1}$ if elitism is applied.
    end if
    $i \leftarrow i + 1$
until Termination criteria are met.
```

Algorithm 1: General procedure of an EDA.

with the empirical evidence and could lead to the construction of incorrect models. Copula functions offer an alternative to construct more flexible distributions since the multivariate distribution can be decomposed into the margins and a copula that sets the dependence structure among the variables.

The learning step of copula-based EDAs involve the estimation of both the marginal distributions and a dependence structure, which is determined by the copula. The marginal distributions can be modeled by parametric and nonparametric distributions. Once the margins are determined, the selected population is transformed into uniform variables in $(0, 1)$ by means of the evaluation of each marginal cumulative distribution function. A copula-based model of the dependence structure is then estimated from this transformed population. The sampling step usually starts with the generation of a population of uniform variables in $(0, 1)$ that resembles the dependence structure described by the copula estimated in the learning step. Finally, this uniform population is transformed to the domain of the variables through the evaluation of the inverse of each marginal cumulative distribution function.

# 4 Survey of EDAs Based on Copulas

To the best of our knowledge, the technical report [68] and the theses [3, 6] can be considered as the first attempts to incorporate the explicit use of copulas into EDAs. Since then, a considerable number of EDAs based on copula theory have been proposed in the literature. Figure 1 shows a time-line of the publications that deal with copula-based EDAs that are described in this paper. According to the copula model being employed, these algorithms can be classified into two groups as EDAs based on either multivariate or factorized copulas. Sections 4.1 and 4.2 provide a description of the algorithms belonging to each group.

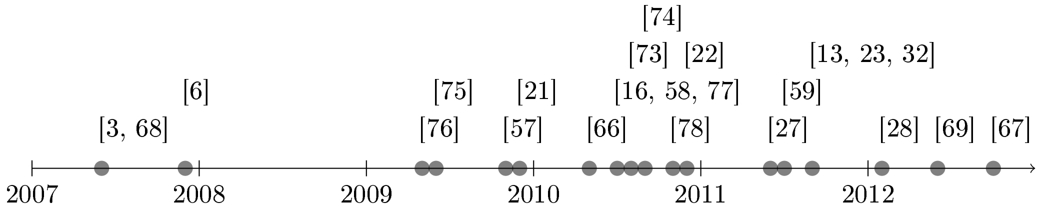

Figure 1: Time-line of publications dealing with EDAs based on copulas.

## 4.1 EDAs Based on Multivariate Copulas

Among the EDAs based on multivariate copulas, research has focused on the product, normal and Archimedean copulas. In the following sections, these algorithms are described.

### 4.1.1 EDAs Based on Product Copula

The Univariate Marginal Distribution Algorithm (UMDA) for continuous variables [45, 46] assumes a model of independence of marginal distributions. An EDA based on the product copula is a generalization of UMDA, which can support different types of marginal distributions [68].

### 4.1.2 EDAs Based on the Normal Copula

The Gaussian Copula Estimation of Distribution Algorithm (GCEDA) [3, 67, 68] uses a multivariate normal copula to model the probabilistic dependence structure of the best solutions at each generation. Regarding the types of margins, two instances of GCEDA are presented: one uses normal marginal distributions and the other empirical margins smoothed with normal kernels. If the marginal distributions are not normal, the correlation matrix is estimated through the inversion of the non-parametric estimator of Kendall's tau for each pair of variables. Otherwise, maximum likelihood estimation is used. If the matrix resulting from the inversion of Kendall's tau is not positive-definite, the correction proposed in [56] is applied.

In [3, 68], GCEDA is compared to UMDA and the Estimation of Multivariate Normal Algorithm (EMNA) [47] according to their performance in the functions Summation Cancellation, Sphere, Ackley and Griewank in 10 and 50 dimensions. Notice that GCEDA is equivalent to EMNA when all marginal distributions are normal. These algorithms are compared according to the number of function evaluations required to reach a certain value of the objective function. Different population sizes are used for each algorithm-function pair. An empirical procedure is used to select the minimum value of the population size that guarantees that the algorithm converges 95% of the total number of independent runs. The goal of the experiments is to show the effect of using neither normal or normal kernel margins under a normal dependence structure when the initial population is sampled symmetrically and asymmetrically with respect to the optimum. This issue is studied in more detail in [67].

Other algorithms, theoretically similar to GCEDA from the copula point of view, are described in [6] and [22]. The former presents differences regarding the estimation of the marginal distributions and the use of techniques such as variance scaling [31]. The experimental study presented in [6] includes 20 unimodal and multimodal functions. The performance of the EDA presented in [22] is evaluated in one 2-dimensional function and eight 10-dimensional functions. The population size of 100 individuals is used for all the test functions.

The paper [76] proposes a multivariate copula to model the search distributions in EDAs, but only a 2-dimensional EDA based on a bivariate normal copula and normal marginal distributions is described in detail. The parameter of the normal copula, here a scalar value, is estimated either by maximum likelihood or using the empirical Pearson correlation coefficient. During the simulation step, observations of the bivariate copula are simulated by conditional sampling and then, the value of each variable in the new population is obtained through the inverse of the corresponding normal marginal distribution function. Part of the new population is generated using a mutation operator, but no details are given for its effect. Notice that in this case, since both the bivariate copula and the margins are normal, the proposed EDA is equivalent to EMNA.

The performance of the proposed algorithm is compared with a continuous extension of the Population Based Incremental Learning (PBIL) algorithm for continuous domains [62] in nine 2-dimensional functions with and without the use of the mutation operator. The termination criteria are to reach the optimum of the function or a maximum of 1000 generations. All algorithms use a population size of 500 individuals for the first three functions and 100 individuals for the rest. The results show that the EDA based on copula reaches better fitness values than PBIL with a smaller number of function evaluations, in most of the test functions. Nevertheless, these same results could have been obtained with EMNA and the advantages of the use of copulas are not therefore clearly illustrated.

In [23], a multivariate normal copula is combined with an opposition-based learning mechanism [70]. The proposed EDA incorporates opposition-based learning after the generation of the initial population and after the simulation of the copula-based probabilistic model. The best solutions of both the new population and its opposite population are selected to form the population of the new generation. The correlation matrix of the multivariate normal copula is estimated through the inversion of the non-parametric estimator of Kendall's tau for each pair of variables. No information is given about the estimation of the marginal distributions.

The behavior of the proposed algorithm is studied in test functions and a real-world problem. The benchmark problems are eight functions in 10 dimensions and one bivariate problem. The real-world problem involves the optimal placement of readers in Radio Frequency Identification (RFID) networks. The proposed algorithm is compared with an EDA called *original version of EDA* or *standard EDA* (no description is given about this algorithm). Both algorithms use the same population size of 50 individuals for all functions and the maximum number of generations is 500. The paper also claims that the use of opposite populations improve the convergence performance of the algorithm, but this issue is not illustrated in the experiments. The paper does not include an experiment that compares the behavior of the EDA based on the multivariate normal copula with and without the use of opposition-based learning. Only the results for the algorithm using opposition-based learning are presented and therefore, it is difficult to assess the effect of this technique in the optimization. The result presented regarding the optimal placement of readers in RFID networks is an optimal solution of a problem instance. It is stated that the proposed algorithm is *effective in the placement of readers in RFID networks.*

### 4.1.3 EDAs Based on Archimedean Copulas

All the EDAs described in this section, except the algorithm presented in [75], are based on exchangeable Archimedean copulas. These copulas constitute the most common multivariate generalization of bivariate Archimedean copulas. In this case, the multivariate dependence structure is determined by a single copula parameter, regardless of the dimension, and all the variables in the structure are exchangeable.

The paper [75] constitutes a follow up of the EDA presented in [76] that proposes the use of a bivariate Archimedean copula instead of the bivariate normal copula. Two EDAs are described: one employs a Clayton copula and the other an Ali-Mikhail-Haq copula. It is not specified how the parameters of the Archimedean copula are determined. The margins are modeled with normal, $t$ or empirical distributions. As the EDAs presented in [76], these algorithms only optimize bivariate functions. The algorithms based on the Clayton and Ali-Mikhail-Haq copulas are studied in nine bivariate test functions. Both algorithms use a population size of 100 individuals in all the test functions, and terminate if the optimum is found or after 1000 evaluations. It is reported that the proposed algorithms perform better than the EDA presented in [76], which is based on the bivariate normal copula. Nevertheless, it is not described how the parameters of the Archimedean copulas are determined or how the margins are modeled.

The paper [21] presents an algorithm called Estimation of Distribution Algorithm with Laplace Transform Archimedean Copula (EDALTAC), which uses exchangeable Archimedean copulas and normal kernels for the marginal distributions. The parameters of the generator of the Archimedean copulas are estimated by maximum likelihood. EDALTAC is compared to an EDA that the author identifies as *the original Estimation of Distribution Algorithm* in five 20-dimensional test functions. In all cases, both algorithms employ the same population size of 50 individuals and stop if a maximum of 200 generations is reached. The paper states that EDALTAC uses an Archimedean copula but does not clarify which particular member of this family is used in the experiments.

On a similar note, the papers [73, 74, 77] present EDAs that model the search distributions using an exchangeable Archimedean copula with constant parameter (Clayton on the first and third, and Gumbel on the second) and normal or empirical marginal distributions. The papers [73] and [74] are similar, the Gumbel copula is used in the former and the Clayton copula in the latter. These EDAs are particular cases of the algorithm described in [21], with the difference that the parameters of the generator are fixed.

In the experiments, the algorithms based on copulas are compared to UMDA and MIMIC for continuous domains, two instances of the Estimation of Gaussian Networks Algorithm (EGNA) [46], and Evolution Strategies (ES) [4] in Summation Cancellation, Schwefel and Griewank functions in 10 dimensions. In the first paper [77], the parameter of the Clayton copula is set to 0.1 for the first function, and 0.5 for the second and third functions. In the second paper [73], the parameter of the Gumbel copula is set to 1.5, while in the third paper [74] the parameter of the Clayton copula is set to 1.0. These values are close to the values corresponding to independence, which theoretically should make these algorithms achieve results that are similar to the ones achieved by an algorithm that assumes independence and uses the same marginal distributions. The paper does not explain why in some cases the proposed algorithms achieve results that are considerably different from the ones achieved by UMDA, while their probabilistic models are similar.

Two EDAs based on Clayton and Gumbel exchangeable Archimedean copulas are also presented in [32]. In both cases, the marginal distributions are normal. For the estimation of the parameters of the Clayton and Gumbel copulas, two variables are randomly chosen and the values of the parameters that maximize the corresponding 2-dimensional marginal density are selected. It is important to stress out that this method is not a maximum pseudo-likelihood estimation but an approach based on it. Maximum pseudo-likelihood estimation involves finding the values of the parameters that maximize the likelihood function of the multivariate copula, not the likelihood of an arbitrary 2-dimensional margin.

In the experiments, the proposed EDAs are compared with UMDA and MIMIC for continuous domains in six 100-dimensional benchmark functions. All algorithms use the same population size of 100 individuals and the selected population is one-half of the population. In a glance, a sample of 50 observations seems scarce for the estimation of a probabilistic model for the dependence structure of 100 variables and the results of the algorithms may be affected by the use of a small population. For instance, the algorithms based on copulas behave much better than UMDA in Sphere, although it is known that the variables of this function are independent.

As far as we are concerned, the paper [13] constitutes the first attempt to use probabilistic models based on copulas in the context of parallel EDAs. The proposed algorithm is a distributed EDA based on the island model [48]. The general idea of the algorithm is to evolve various subpopulations simultaneously by using a different multivariate copula to model the individuals of each subpopulation. Specifically, the authors study the behavior of an algorithm that evolves two subpopulations modeled by exchangeable Clayton and Gumbel copulas. Constant values of 1.00 and 1.05 are assumed for the parameters of the Clayton and Gumbel copulas, respectively. In both cases, the marginal distributions are modeled with the empirical marginal distribution.

The performance of the proposed parallel EDA is compared to UMDA and MIMIC for continuous domains, and the algorithms based on multivariate Clayton and Gumbel copulas proposed in [73, 74]. The experiments that were carried out are rather limited and only descriptive comments are given in the analysis of the results. No other parallel algorithm was included in the comparison and therefore, it is difficult to asses the effect of the use of probabilistic models based on copulas in the context of parallel EDAs. Also, the results achieved by the proposed parallel algorithm are similar to the ones obtained by the sequential algorithms based on multivariate Clayton and Gumbel copulas. We consider the reason behind this issue is that the dependence structures modeled by each copula is very similar for the parameter values that were selected. Finally, as commented before in the case of the numerical results presented in [73, 74], the parallel algorithm achieves results that are considerably better than the ones achieved by UMDA in the function Summation Cancellation, but the probabilistic models that are used are not so different. In our opinion, this issue deserves more study.

## 4.2 EDAs Based on Copula Factorizations

One alternative to overcome some of the shortcomings of the multivariate copula modeling approach is to use copula factorizations that build high-dimensional probabilistic models by using lower-dimensional copulas as building blocks. The algorithms based on copula factorization reviewed in this paper are divided into three groups: EDAs based on empirical factorizations, vines and nested Archimedean copulas. The next three sections describe the algorithms belonging to each group.

### 4.2.1 EDAs Based on Empirical Factorizations

The algorithms described in this section use probabilistic models based on copula factorizations that are based on either empirical approaches or constitute extensions of existing algorithms that originally were not based on copulas.

The paper [57] proposes an extension of the Mutual Information Maximization for Input Clustering (MIMIC) algorithm for continuous domains [45, 46]. Similarly to MIMIC, the proposed algorithm learns a chain dependence structure but it uses bivariate copulas instead of bivariate normal distributions. Two instances of this algorithm are presented, one employs normal copulas and the other Frank copulas. The marginal distributions are beta in both cases, and consequently, a linear transformation is used to extend the beta distribution off the boundaries of the $(0, 1)$ interval. The learning step of the proposed EDA begins with the calculation of mutual information between all pairs of variables. The values of the parameters of the copulas obtained by the inversion of Kendall's tau are used as the initial approximation for the maximum likelihood estimation. Next, a permutation of the variables that maximizes the sum of the mutual information of the copulas in the corresponding chain structure is selected. Performing such selection is a computational intensive task and, for efficiency reasons, a greedy algorithm originally proposed in [18] and adapted in [45] is used to compute an approximate solution. During the simulation step, a conditional simulation approach is followed for the generation of a sample with the relationships described in the chain dependence structure.

The performance of the two algorithms based on normal and Frank copulas is compared with the results of MIMIC for continuous domains in five test problems: Ackley, Griewank, Rastrigin, Rosenbrock and Sphere in 10 dimensions. The algorithms use a population size of 100 individuals and stop after 300000 function evaluations or when the improvement of the fitness of the best individual in the population is not greater than a threshold for 25 generations. The three algorithms perform similarly on the five test problems. In our opinion, this result is due to the fact that the functions Ackley, Griewank, Rastrigin and Sphere display weak interactions between the variables, so they can be optimized easily by UMDA [27, 66, 69]. Therefore, the best performance of these algorithms would be achieved when the values of copula parameters correspond to the independence model. The Rosenbrock function presents non-linear interactions [11], which can not be captured by neither normal nor Frank copulas.

The paper [16] investigates the use of bivariate empirical copulas and a multivariate extension of Archimedean copulas. The EDA based on bivariate empirical copulas uses empirical margins, so this algorithm is completely non-parametric. Both the marginal distributions and the bivariate empirical copulas are defined through a linear interpolation of the sample in the selected population. As the paper states, the construction based on bivariate empirical copulas does not provide neither a proper multidimensional copula nor a joint probability density function of the selected population. This issue constitutes a limitation for studying the characteristics of the probabilistic model learned at each generation. The EDA based on Archimedean copulas uses a construction, which is similar to a fully nested Archimedean copula. This algorithm uses Frank, Clayton or Clayton survival copulas. The parameters of the copulas are fixed to a constant value. The marginal distributions are modeled as in the EDA based on bivariate empirical copulas.

Six instances of the algorithm based on Archimedean copulas and the EDA based on empirical copulas are compared with UMDA for continuous domains, EMNA and the eigenspace EDA (EDDA) [72] with 24 functions in 2, 5 and 10 dimensions. The six instances of the EDA based on Archimedean copulas correspond to two different values of the copula parameters. The algorithms use a population size of 1000 individuals for all functions and dimensions. The results show that the EDAs based on copulas behave better than the other algorithms on 20% of the functions.

The algorithm presented in [59] constitutes an extension to the continuous domain of the EDA based on discrete dependency trees described in [5]. This EDA employs a dependency tree along with bivariate copulas. The algorithm selects the copula that best fits a bivariate sample among six candidate copulas: Ali-Mikhail-Haq, Clayton, Farlie-Gumbel-Morgenstern, Frank, normal and Gumbel. The authors also incorporate this copula selection mechanism to the copula-based MIMIC proposed in [57]. The margins are modeled using normal and normal kernel distributions. The strategy followed to learn the tree structure is in the same spirit of that used in [57]. The goal is to select the dependency tree that minimizes Kullback-Leibler divergence between the true unknown density and the estimated density function. This aim is achieved by finding the tree that results in the highest pairwise mutual information by means an algorithm that finds a minimum spanning tree. The parameters of the copulas are estimated by maximum likelihood and use the inversion of Kendall's tau as an initial approximation. The selection of the copula for each pair of variables is based on the highest value of the likelihood function.

The experimental studies compare two instances of each one of the algorithms MIMIC and the dependency tree EDA, one with normal copulas and normal margins, and the other with copula selection and normal kernel margins. The algorithms are compared on a benchmark of 7 functions in 4 and 12 dimensions. The population size is 10 times the dimension of the problem for all algorithms and functions. The algorithms stop if the optimum of the function is found, after 100000 function evaluations, or if the improvement of the fitness of the best individual in the population is not greater than a threshold for 30 generations. The discussion of the results of the experiments states that the algorithms based on copula have better performance than the EDAs based on normal copulas on various functions, but it was not illustrated why a certain dependence structure was more appropriate for a particular case. The paper concludes that the algorithms based on copula selection perform very similar, but in our opinion the reasons for this behavior are the same as given above in the case of the algorithms proposed in [57].

### 4.2.2 EDAs Based on Vines

Pair-copula constructions constitute an alternative to the multivariate copula approach. These constructions, and specifically the graphical models known as vines, allow extending bivariate copulas to higher dimensions and representing a rich variety of dependences by combining bivariate copulas from different families.

To the best of our knowledge, the use of vines in the context of EDAs begins with the introduction of the class of Vine EDAs (VEDAs) [27, 28, 66, 67] and two specific members of this class: CVEDA and DVEDA, based on C-vines and D-vines, respectively. The learning procedure of CVEDA and DVEDA include two main steps: the selection of the structure of the vines and the choice of the bivariate copulas in the factorization.

The selection of a specific pair-copula decomposition implies to choose an appropriate order of the variables. To do so, greedy heuristics are used to detect the most important bivariate dependences. The first step of the estimation procedure consist in assigning weights to the edges and the absolute value of empirical Kendall's tau between pair of variables is used with this purpose. The next step consist in determining the appropriate order of the variables of the decomposition, which depend on the type of pair-copula decomposition. In a C-vine, the tree that maximizes the sum of the weights of the root node to the others is chosen as the appropriate factorization. In a D-vine, the first tree is selected by maximizing the weighted sequence of the variables through the cheapest insertion heuristic to compute an approximate solution of the TSP (see [12, 55]). In both decompositions, the selection of pair-copulas is accomplished through the Cramér von Mises statistics [26]. Firstly, the product copula is considered [25]; if there is enough evidence against the null hypothesis of independence, it is rejected and the bivariate copula that minimizes the Cramér von Mises statistics is chosen. As the cost of the construction of C-vines and D-vines increases rapidly with the number of variables, CVEDA and DVEDA use a truncation strategy [12] in order to simplify these models. If a vine is truncated at a given tree, all the copulas in the subsequent trees are assumed to be product copulas. A model selection procedure based on either AIC [2] or BIC [61] is applied to detect the required number of trees.

The experimental studies with CVEDA and DVEDA assess the effect of combining different copulas, applying the truncation strategy, and selecting the structure of C-vines and D-vines in the performance of VEDA. CVEDA and DVEDA are compared to UMDA and GCEDA in the functions Sphere, Griewank, Ackley and Summation Cancellation in 10 dimensions. All the algorithms use normal margins and stop when the global optimum is found or after 500000 function evaluations. The population size corresponds to the smallest size that the algorithm requires to find the global optimum in 30 of 30 independent runs. [53]. The bivariate copulas used by the vine-based algorithms are normal, $t$, Clayton, Gumbel and the rotated versions of Clayton and Gumbel copulas.

CVEDA and DVEDA exhibit good performance in problems with both strong and weak correlations between the variables: while UMDA uses the independence model and GCEDA assumes a linear dependence structure, CVEDA and DVEDA do not assume the same type of dependence across all pairs of variables. The estimation procedures used by the vine-based algorithms select among a group of candidate bivariate copulas, the one that fits the data appropriately. CVEDA and DVEDA perform, in general, between UMDA and GCEDA in terms of the number of function evaluations. It is also illustrated that the construction of the vines up to an arbitrary number of trees may lead to a poor performance of the algorithm, since the appropriate number of trees is tightly related to the characteristics of the problem. The truncation methods based on AIC and BIC are more appropriate. Additionally, it is shown that it is important to make a conscious selection of the dependences that are explicitly modeled in the first tree of the vines.

The paper [58] also presents an EDA based on vines called D-vine EDA. In this algorithm, only normal copulas are used in the first two trees and conditional independence is assumed for the rest of the trees, i.e. the D-vine is always truncated at the second tree. Although the paper states that for practical purposes it is not necessary to build the complete D-vine, it was illustrated in [27, 66, 67] that an arbitrary selection of the number of trees in the vines compromises the convergence of EDAs based on vines. The selection of the structure of the D-vine is based on the minimization of the Kullback-Leibler divergence between the true unknown density function and the density function estimated using the truncated D-vine factorization. The algorithm used to simulate the truncated D-vine is similar to that used in [57]. The marginal distributions are beta.

In the experimental section, D-vine EDA is compared to MIMIC and UMDA in three test problems (Ackley, Rosenbrock and Sphere functions) in 10 dimensions. The same population size of 300 individuals is used for all the algorithms in all the test problems. The algorithms stop after 300000 evaluations or if the improvement of the fitness of the best individual in the population is not greater than a threshold for 25 generations. The algorithms obtain similar results on Ackley and Sphere functions, although according to the results obtained in [66, 69], UMDA should obtain the best performance on these functions.

#### 4.2.3 EDAs Based on Nested Archimedean Copulas

The paper [78] proposes an EDA called LNAcopula-EDA that uses a representation of hierarchically nested Archimedean copulas based on Lévy subordinators [35]. By using Lévy subordinators, the probabilistic model is specified through a division of the variables into $s$ groups and the selection of an outer $s$-dimensional Archimedean copula, an arbitrary positive random variable $V$ and $s$ Lévy subordinators. In LNAcopula-EDA, the variables are divided into $s = 2$ groups and the outer copula is a 2-dimensional Clayton copula with parameter $\theta$. The variable $V$ is gamma with parameters $1/\theta$ and 1, and the two Lévy subordinators are gamma and inverse Gaussian processes. The marginal distributions are modeled using either empirical or normal distributions.

Key aspects of the proposed algorithm are not clearly described in the paper. The variables are divided into two groups according to the correlation between the variables but the procedure followed for the selection of the groups is not described. At this point, it is also important to stress out the the use of correlation as a measure of dependence in the context of copulas presents several limitations (see e.g. [71]). Besides, no information is given about the method used to estimate the parameter of the outer Clayton copula or if it is assumed to be constant.

The behavior of the proposed algorithm is compared to the results presented in [57] by using the same experimental set of functions. LNAcopula-EDA uses the same population size of 100 individuals for all functions. The experiments show that LNAcopula-EDA outperforms the other algorithms.

## 5 Summary and Conclusions

In recent years, a considerable number of papers that propose EDAs based on copulas have appeared in the literature. Nevertheless, this is still a highly unexplored topic. There are many aspects that should be considered in future research in order to achieve a deeper understanding of mechanisms governing the behavior of these methods. It is important to pay more attention to how the inherent features of each copula-based model affect the optimization procedure. Regarding the experimental studies, in occasions the description of the proposed EDAs lacks essential information to guarantee the reproducibility of the results. It is also important to draw attention to the experimental design, since poor experimental setups often result in misleading conclusions. Finally, it would be also beneficial to have more publicly available implementations of copula-based EDAs, since this would facilitate performing numerical comparisons. To the best of our knowledge, the only publicly available implementations of EDAs based on copulas are provided as part of the `copulaedas` package [29, 30] for R [54].

# References

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

[2] H. Akaike. A new look at statistical model identification. *IEEE Transactions on Automatic Control*, 19:716–723, 1974.

[3] R. J. Arderí. Algoritmo con estimación de distribuciones con cópula gaussiana. Bachelor thesis, University of Havana, Cuba, June 2007.

[4] T. Bäck. *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, 1996.

[5] S. Baluja and S. Davies. Using optimal dependency-trees for combinatorial optimization: Learning the structure of the search space. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 30–38, 1997.

[6] S. Barba-Moreno. Una propuesta para algoritmos de estimación de distribución no paramétricos. Master's thesis, Center for Research in Mathematics, Mexico, December 2007.

[7] T. Bedford and R. M. Cooke. Monte Carlo simulation of vine dependent random variables forapplications in uncertainty analysis. In *Proceedings of ESREL 2001*, Turin, Italy, 2001.

[8] T. Bedford and R. M. Cooke. Probability density decomposition for conditionally dependent random variables modeled by vines. *Annals of Mathematics and Artificial Intelligence*, 32(1):245–268, 2001.

[9] T. Bedford and R. M. Cooke. Vines — a new graphical model for dependent random variables. *The Annals of Statistics*, 30(4):1031–1068, 2002.

[10] D. Berg and K. Aas. Models for construction of multivariate dependence. Note SAMBA/23/07, Norwegian Computing Center, NR, Norway, June 2007.

[11] P. A. N. Bosman and D. Thierens. Numerical optimization with real-valued estimation of distribution algorithms. In M. Pelikan, K. Sastry, and E. Cantú-Paz, editors, *Scalable Optimization via Probabilistic Modeling. From Algorithms to Applications*, pages 91–120. Springer-Verlag, 2006.

[12] E. C. Brechmann. Truncated and simplified regular vines and their applications. Diploma thesis, University of Technology, Munich, Germany, October 2010.

[13] C. Chang and L. Wang. A multi-population parallel estimation of distribution algorithms based on Clayton and Gumbel copulas. In *Proceedings of the Third International Conference on Artificial Intelligence and Computational Intelligence (AICI 2011), Part I*, volume 7002 of *Lecture Notes in Computer Science*, pages 634–643. Springer-Verlag, 2011.

[14] A. Charpentier, J. D. Fermanian, and O. Scaillet. The estimation of copulas: Theory and practice. In *Copulas: From Theory to Application in Finance*, pages 35–62. Risk Books, 2007.

[15] B. Choroś, R. Ibragimov, and E. Permiakova. Copula estimation. In P. Jaworski, F. Durante, W. K. Härdle, and T. Rychlik, editors, *Copula Theory and Its Applications*, number 198 in Lecture Notes in Statistics, pages 77–93. Springer-Verlag, 2010. ISBN 978-3-642-12464-8.

[16] A. Cuesta-Infante, R. Santana, J. I. Hidalgo, C. Bielza, and P. Larrañaga. Bivariate empirical and $n$-variate archimedean copulas in estimation of distribution algorithms. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2010)*, pages 1355–1362, July 2010.

[17] C. Czado. Pair-copula constructions of multivariate copulas. In P. Jaworski, F. Durante, W. K. Härdle, and T. Rychlik, editors, *Copula Theory and Its Applications*, number 198 in Lecture Notes in Statistics, pages 94–111. Springer-Verlag, 2010. ISBN 978-3-642-12464-8.

[18] J. S. De Bonet, C. L. Isbell, and P. Viola. MIMIC: Finding optima by estimating probability densities. In M. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 424–430. The MIT Press, 1997.

[19] P. Deheuvels. La fonction de dépendance empirique et ses propriétés: Un test non paramétrique d'indépendance. *Bulletin de la Classe des Sciences, V. Série, Académie Royale de Belgique*, 65:274–292, 1979.

[20] L. Devroye. *Non-Uniform Random Variate Generation.* Springer-Verlag, 1986. ISBN 0-387-96305-7.

[21] Y. Gao. Multivariate estimation of distribution algorithm with Laplace transform archimedean copula. In W. Hu and X. Li, editors, *Proceedings of the International Conference on Information Engineering and Computer Science (ICIECS 2009)*, December 2009.

[22] Y. Gao, X. Hu, and H. Liu. Estimation of distribution algorithm based on multivariate gaussian copulas. In *Proceedings of the International Conference on Progress in Informatics and Computing (PIC 2010)*, pages 254–257, December 2010.

[23] Y. Gao, X. Hu, H. Liu, F. Li, and L. Peng. Opposition-based learning estimation of distribution algorithm with gaussian copulas and its application to placement of RFID readers. In *Proceedings of the Third International Conference on Artificial Intelligence and Computational Intelligence (AICI 2011), Part I*, volume 7002 of *Lecture Notes in Computer Science*, pages 219–227. Springer-Verlag, 2011.

[24] C. Genest, K. Ghoudi, and L. P. Rivest. A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika*, 82:543–552, 1995.

[25] C. Genest and B. Rémillard. Tests of independence or randomness based on the empirical copula process. *Test*, 13(2):335–369, 2004.

[26] C. Genest and B. Rémillard. Validity of the parametric bootstrap for goodness-of-fit testing in semiparametric models. *Annales de l'Institut Henri Poincaré: Probabilités et Statistiques*, 44:1096–1127, 2008.

[27] Y. González-Fernández. Algoritmos con estimación de distribuciones basados en cópulas y vines. Bachelor thesis, University of Havana, Cuba, June 2011.

[28] Y. González-Fernández, D. Carrera, M. Soto, and A. Ochoa. Vine estimation of distribution algorithm. In *VIII Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB 2012)*, February 2012. Available at `http://congresomaeb2012.uclm.es/papers/paper_99.pdf`.

[29] Y. González-Fernández and M. Soto. *copulaedas: Estimation of Distribution Algorithms Based on Copulas*, 2011. R package version 1.1.0. Available at `http://CRAN.R-project.org/package=copulaedas`.

[30] Y. González-Fernández and M. Soto. copulaedas: An R package for estimation of distribution algorithms based on copulas. *Submitted for publication*, February 2012. Preprint available at `http://arxiv.org/abs/1209.5429`.

[31] J. Grahl, P. A. N. Bosman, and F. Rothlauf. The correlation–triggered adaptive variance scaling IDEA. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2006)*, pages 397–404, 2006.

[32] X. Guo, L. Wang, J. Zeng, and X. Zhang. Copula estimation of distribution algorithm with PMLE. In *Proceedings of the Seventh International Conference on Natural Computation (ICNC 2011)*, pages 1077–1081, 2011.

[33] W. Härdle, O. Okhrin, and Y. Okhrin. Modeling dependencies with copulae. In W. Härdle, N. Hautsch, and L. Overbeck, editors, *Applied Quantitative Finance*, pages 3–36. Springer-Verlag, second edition, 2009.

[34] M. Hauschild and M. Pelikan. An introduction and survey of estimation of distribution algorithms. *Swarm and Evolutionary Computation*, 1:111–128, 2011.

[35] C. Hering, M. Hofert, J.-F. Mai, and M. Scherer. Constructing hierarchical archimedean copulas with Lévy subordinators. *Journal of Multivariate Analysis*, 101:1428–1433, 2010.

[36] I. Hobæk Haff. Parameter estimation for pair-copula constructions. Note SAMBA/36/10, Norwegian Computing Center, NR, October 2010.

[37] M. Hofert. Construction and sampling of nested archimedean copulas. In P. Jaworski, F. Durante, W. K. Härdle, and T. Rychlik, editors, *Copula Theory and Its Applications*, number 198 in Lecture Notes in Statistics, pages 94–111. Springer-Verlag, 2010. ISBN 978-3-642-12464-8.

[38] H. Joe. Families of $m$-variate distributions with given margins and $m(m-1)/2$ bivariate dependence parameters. In L. Rüschendorf, B. Schweizer, and M. D. Taylor, editors, *Distributions with fixed marginals and related topics*, pages 120–141, 1996.

[39] H. Joe. *Multivariate Models and Dependence Concepts.* Chapman & Hall, 1997.

[40] H. Joe and J .J. Xu. The estimation method of inference functions for margins for multivariate models. Technical Report 166, University of British Columbia, 1996.

[41] S. Kern, S. D. Müller, N. Hansen, D. Büche, J. Ocenasek, and P. Koumoutsakos. Learning probability distributions in continuous evolutionary algorithms – A comparative review. *Natural Computing*, 3:77–112, 2003.

[42] D. Kurowicka and R. M. Cooke. *Uncertainty Analysis with High Dimensional Dependence Modelling*. John Wiley & Sons, 2006. ISBN 978-0-470-86306-0.

[43] D. Kurowicka and R. M. Cooke. Sampling algorithms for generating joint uniform distributions using the vine-copula method. *Computational Statistics & Data Analysis*, 51:2889–2906, 2007.

[44] P. Larrañaga. A review on estimation of distribution algorithms. In P. Larrañaga and J. A. Lozano, editors, *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*, pages 57–100. Kluwer Academic Publisher, 2002.

[45] P. Larrañaga, R. Etxeberria, J. A. Lozano, and J. M. Peña. Optimization by learning and simulation of Bayesian and Gaussian networks. Technical Report EHU-KZAA-IK-4/99, University of the Basque Country, December 1999.

[46] P. Larrañaga, R. Etxeberria, J. A. Lozano, and J. M. Peña. Optimization in continuous domains by learning and simulation of gaussian networks. In *Proceedings of the Workshop in Optimization by Building and Using Probabilistic Models in the Genetic and Evolutionary Computation Conference (GECCO 2000)*, pages 201–204, 2000.

[47] P. Larrañaga, J. A. Lozano, and E. Bengoetxea. Estimation of distribution algorithms based on multivariate normal and gaussian networks. Technical Report EHU-KZAA-IK-1/01, University of the Basque Country, 2001.

[48] J. Madera, E. Alba, and A. Ochoa. Parallel estimation of distribution algorithms. In E. Alba, editor, *Parallel Metaheuristics: A New Class of Algorithms*. John Wiley & Sons, 2005.

[49] A. W. Marshall and J. Olkin. Families of multivariate distributions. *Journal of the American Statistical Association*, 83:834–841, 1988.

[50] A. J. McNeil. Sampling nested archimedean copulas. *Journal of Statistical Computation and Simulation*, 78(6):567–581, 2008.

[51] H. Mühlenbein and G. Paaß. From recombination of genes to the estimation of distributions i. Binary parameters. In *Parallel Problem Solving from Nature — PPSN IV*, pages 178–187. Springer-Verlag, 1996.

[52] R. B. Nelsen. *An Introduction to Copulas*. Springer-Verlag, second edition, 2006. ISBN 978-0387-28659-4.

[53] M. Pelikan. *Hierarchical Bayesian Optimization Algorithm. Toward a New Generation of Evolutionary Algorithms*. Springer-Verlag, 2005.

[54] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0. `http://www.R-project.org/`.

[55] D. J. Rosenkrantz, R. E. Stearns, and P. M. Lewis II. An analysis of several heuristics for the traveling salesman problem. *SIAM Journal on Computing*, 6(3):563–581, 1977.

[56] P. Rousseeuw and G. Molenberghs. Transformation of nonpositive semidefinite correlation matrices. *Communications in Statistics: Theory and Methods*, 22:965–984, 1993.

[57] R. Salinas-Gutiérrez, A. Hernández-Aguirre, and E. Villa-Diharce. Using copulas in estimation of distribution algorithms. In *Proceedings of the Eight Mexican International Conference on Artificial Intelligence (MICAI 2009)*, pages 658–668, November 2009.

[58] R. Salinas-Gutiérrez, A. Hernández-Aguirre, and E. Villa-Diharce. D-vine EDA: A new estimation of distribution algorithm based on regular vines. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2010)*, pages 359–365, July 2010.

[59] R. Salinas-Gutiérrez, A. Hernández-Aguirre, and E. Villa-Diharce. Dependence trees with copula selection for continuous estimation of distribution algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2011)*, pages 585–592, July 2011.

[60] F. Schmid, R. Schmidt, T. Blumentritt, S. Gaißer, and M. Ruppert. Copula-based measures of multivariate association. In P. Jaworski, F. Durante, W. K. Härdle, and T. Rychlik, editors, *Copula Theory and Its Applications*, number 198 in Lecture Notes in Statistics, pages 209–236. Springer-Verlag, 2010. ISBN 978-3-642-12464-8.

[61] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6:461–464, 1978.

[62] M. Sebag and A. Ducoulombier. Extending population-based incremental learning to continuous search spaces. In *Parallel Problem Solving from Nature — PPSN V*, pages 418–427. Springer-Verlag, September 1998.

[63] A. Sklar. Fonctions de répartition à $n$ dimensions et leurs marges. *Publications de l'Institut de Statistique de l'Université de Paris*, 8:229–231, 1959.

[64] A. Sklar. Random variables, joint distribution functions, and copulas. *Kybernetica*, 9:449–460, 1973.

[65] P. X. Song. Multivariate dispersion models generated from gaussian copula. *Scandinavian Journal of Statistics*, 27:305–320, 2000.

[66] M. Soto and Y. González-Fernández. Vine estimation of distribution algorithms. Technical Report ICIMAF 2010-561, Institute of Cybernetics, Mathematics and Physics, Cuba, May 2010. ISSN 0138-8916.

[67] M. Soto, Y. González-Fernández, and A. Ochoa. Modeling with copulas and vines in estimation of distribution algorithms. *In preparation*, 2012. Preprint available at `http://arxiv.org/abs/1210.5500`.

[68] M. Soto, A. Ochoa, and R. J. Arderí. Gaussian copula estimation of distribution algorithm. Technical Report ICIMAF 2007-406, Institute of Cybernetics, Mathematics and Physics, Cuba, June 2007. ISSN 0138-8916.

[69] M. Soto, A. Ochoa, Y. González-Fernández, Y. Milanés, A. Álvarez, D. Carrera, and E. Moreno. Vine estimation of distribution algorithms with application to molecular docking. In S. Shakya and R. Santana, editors, *Markov Networks in Evolutionary Computation*, volume 14 of *Adaptation, Learning, and Optimization*, pages 209–225. Springer-Verlag, 2012. ISBN 978-3-642-28899-9.

[70] H. R. Tizhoosh. Opposition-based learning: A new scheme for machine intelligence. In *Proceedings of the International Conference on Computational Intelligence for Modelling Control and Automation (CIMCA 2005)*, volume 1, pages 695–701, 2005.

[71] P. K. Trivedi and D. M. Zimmer. Copula modeling: An introduction for practitioners. *Foundations and Trends in Econometrics*, 1:1–111, 2005.

[72] M. Wagner, A. Auger, and M. Schoenauer. EEDA: A new robust estimation of distribution algorithm. Research Report RR-5190, INRIA, France, 2004.

[73] L. Wang, X. Guo, J. Zeng, and Y. Hong. Using Gumbel copula and empirical marginal distribution in estimation of distribution algorithm. In *Proceedings of the Third International Workshop on Advanced Computational Intelligence (IWACI 2010)*, pages 583–587, August 2010.

[74] L. Wang, Y. Wang, J. Zeng, and Y. Hong. An estimation of distribution algorithm based on Clayton copula and empirical margins. In K. Li, X. Li, S. Ma, and G. W. Irwin, editors, *Life System Modeling and Intelligent Computing*, pages 82–88. Springer-Verlag, 2010.

[75] L. Wang, J. Zeng, and Y. Hong. Estimation of distribution algorithm based on archimedean copulas. In *Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation (GEC 2009)*, pages 993–996, June 2009.

[76] L. Wang, J. Zeng, and Y. Hong. Estimation of distribution algorithm based on copula theory. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2009)*, pages 1057–1063, May 2009.

[77] L. Wang, J. Zeng, Y. Hong, and X. Guo. Copula estimation of distribution algorithm sampling from Clayton copula. *Journal of Computational Information Systems*, 6(7):2431–2440, July 2010.

[78] B. Ye, H. Gao, X. Wang, and J. Zeng. Estimation of distribution algorithm based on nested archimedean copulas constructed with Lévy subordinators. In *Proceedings of the Eleventh International Conference on Computer-Aided Industrial Design & Conceptual Design (CAIDCD 2010)*, pages 1586–1590, November 2010.