Raport Badawczy

Research Report

RB/13/2012

On computational problems in the analysis of statistical dependence for imprecise data

O. Hryniewicz, K. Opara

Instytut Badań Systemowych
Polska Akademia Nauk

Systems Research Institute
Polish Academy of Sciences

POLSKA AKADEMIA NAUK

Instytut Badań Systemowych

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 3810100

fax: (+48) (22) 3810105

Kierownik Zakładu zgłaszający pracę:
Prof. dr hab. inż. Olgierd Hryniewicz

Warszawa 2012

On computational problems in the analysis of statistical dependence for imprecise data

Olgierd Hryniewicz, Karol Opara

Systems Research Institute, Newelska 6, 01-447 Warsaw, Poland

Abstract

This paper provides a comprehensive analysis of computational problems concerning calculation of Kendall's τ dependence (association) measure for interval data. Exact algorithms solving this task have unacceptable computational complexity for larger samples, therefore we concentrate on computational problems arising in approximate algorithms. In particular, we propose a set of heuristics solutions for finding minimal and maximal value of Kendall's τ for interval data. Extensive simulation experiments show that some of the heuristics yield very good starting points for optimization procedures based on random generation of linear extensions or evolutionary and direct-search algorithms.

Keywords:
interval data, measures of dependence, Kendall's τ, computation, algorithms

1. Introduction

The analysis of statistical dependence is one of the most important parts of statistics. First statistical procedures for the the analysis of dependent data were invented more than one hundred years ago. Since that time hundreds of particular methods have been devised. The introduction of data mining techniques significantly extended the area of applications where the analysis of dependencies in data plays a crucial role. Despite the fact that the real statistical data are often imprecise, as the data are gathered from intrinsically imprecise measurements, the interest in the statistical analysis of imprecise data is relatively new. First statistical methods applicable in the analysis of imprecise data were proposed in papers published in the 1980s.

The most important publication from that times which established a new statistical methodology for coping with imprecise (fuzzy) data is the book by Kruse and Meyer (1987). Since that time many books and papers on fuzzy statistics have been published. Interesting overview of these methods can be found e.g. in the paper by Gil and Hryniewicz (2009) or in the book by Viertl (2007). The general methodology for the statistical analysis of imprecise (fuzzy) data is still under development, and some new techniques, see e.g. Couso and Sanchez (2011), have been proposed recently.

First publications devoted to the problem of testing statistical hypotheses for dependent statistical data were published in the early 2000s. Testing statistical hypotheses for categorical data displayed in the form of contingency tables was considered in (Hryniewicz, 2004, 2006). The statistical analysis of dependence using the well known Kendall's τ statistics for imprecise data was considered for the first time in the paper by Hébert *et al.* (2003). The most important paper related to the problem of statistical testing of independence with fuzzy data is written by Denœux *et al.* (2005), where this problem has been presented in a more general framework of using rank tests for fuzzy data. In all these papers the authors have noticed important difficulties with the calculation of the values of fuzzy statistics. Hébert *et al.* (2003) proposed an algorithm for the calculation of the exact interval value of Kendall's τ which, unfortunately, was computationally effective only for very small samples (less than 10 elements). Denœux *et al.* (2005) considered an algorithm for the calculation of the approximate interval value of Kendall's τ which was effective also for relatively small samples (max. 30 elements). These findings prompted Hryniewicz and Szediw (2008) to look for the approximate interval value of Kendall's τ that could be used as the starting point in the procedure for the calculation of more precise value of this statistic. In that paper the approximate interval value of Kendall's τ was calculated using a heuristic algorithm for autocorrelated imprecise data coming from statistical quality control. The results appeared promising, especially for highly correlated data. In the paper by Hryniewicz and Opara (2012a) an extended set of heuristic algorithms has been proposed to calculate the approximate interval value of Kendall's τ for usual bivariate interval data. Preliminary results presented in this paper have shown that further investigations are needed. One result of these investigations is described in the paper by Hryniewicz and Opara (2012b), who used a general purpose genetic optimization algorithm for finding better approximations.

In this paper we present a comprehensive analysis of several experiments that significantly extend our knowledge about the efficiency of the algorithms described in these both papers. Moreover, we propose a new algorithm for the calculation of the starting point of the optimization procedure used for the calculation of the interval value of Kendall's τ. The application of this algorithm significantly improves the accuracy of the calculation of the interval value of τ. However, its practical applicability is still restricted by the size of the analyzed sample.

The paper is organized as follows. In its second section we recall some basic information about the measures of dependence used in the statistical analysis of imprecise (fuzzy) data with a special emphasis on Kendall's τ. In the third section we describe in details heuristic algorithms used for the calculation of approximate interval values of Kendall's τ. The fourth section of the paper is devoted to the presentation of simulation experiments that reveal the benefits from using the proposed heuristic solutions. In this section we present these benefits when heuristic solutions are used together with the algorithm proposed by Denœux *et al.* (2005). Similar analysis in the case of the algorithm proposed in Hryniewicz and Opara (2012b) is also presented in the sixth section of the paper. In the fifth section we propose a new method for finding efficient starting points for optimization algorithms. This method is based on the interval-valued Pearson's linear correlation coefficient. We also present results of simulation experiments that show advantages and limitations of the usage of a hybrid algorithm based on this new method. The sixth section provides an analysis of computing Kendall's τ with an evolutionary algorithm. The paper is concluded with some proposals for the efficient calculation of fuzzy Kendall's τ.

2. Statistical measures of dependence for imprecise data

Statistical dependence is modeled by multivariate probability distributions that describe multidimensional random data. Let $X_1, X_2, \ldots, X_p$ be a p-dimensional random vector. The full description of statistical dependencies between the components of this vector is implied by the knowledge of its p-dimensional cumulative distribution function $F(x_1, x_2, \ldots, x_p)$. Sklar (1959) proved that for every two-dimensional cumulative probability distribution function $H(x, y)$ with one-dimensional marginal cumulative probability functions denoted by $F(x)$ and $G(y)$, respectively, there exists a unique function C, called a *copula*, such that $H(x, y) = C(F(x), G(y))$. The original result of Sklar has been later generalized for the case of any p-dimensional probability distribution. The definition of the copula together with the exposition of its properties can be found in many sources, such as e.g. the monograph by Nelsen (1999).

The most popular bivariate probability distribution, the bivariate normal distribution, is defined by the following copula (usually called Gaussian):

$$C(u_1, u_2; \rho) = \Phi_N(\Phi^{-1}(u_1), \Phi^{-1}(u_2); \rho) \tag{1}$$

where $\Phi_N(u_1, u_2)$ is the cumulative probability distribution function of the bivariate normal distribution, and $\Phi^{-1}(u)$ is the inverse of the cumulative probability function of the univariate normal distribution (the quantile function). Parameter ρ is equal to the well known Pearson's coefficient of linear correlation r only in the case of normal marginal probability distributions. In the general case, when marginals have probability distributions different then the normal, the parameter ρ measures the strength of dependence, but adopts different values than the coefficient of linear correlation.

As it can be derived from Sklar's theorem (and its multivariate generalization) any measure that fully describes the dependence between random variables must be a function of the respective copula. Therefore, Pearson's coefficient of linear correlation, whose definition cannot be reformulated in terms of copulas, cannot be used as an universal measure of dependence. However, it can be used as a measure of dependence in the case of the so-called elliptical multivariate probability distributions having symmetric marginal distributions with a finite second moment. For more information on this problem see the work of Embrechts *et al.* (2003).

Other measures of dependence, such as Spearman's ρ_S or Kendall's τ, can be also defined in terms of copulas; see e.g. Nelsen (1999). Therefore, they can be used as general measures of dependence (association) between random variables. These both measures are mutually related, but the population versions of Kendall's τ are easier to calculate for the most popular copulas, and thus easier to use in simulation experiments. For this reason we will restrict our further investigations to this particular measure of dependence.

The coefficient of association (dependence) τ was introduced by Kendall in the 1930s. Its population version is interpreted as the difference between the probability of observing a concordant pair of observations, and probability of observation of a disconcordant pair of observations. There also exists a definition of the population version of Kendall's τ in terms of copulas. For a pair of random variables X, Y whose bivariate probability distribution is defined by the copula $C(F(x), G(y))$, where $F(x)$ and $G(y)$ are the cdf's of X and Y, respectively, Kendall's τ is defined as

$$\tau(X, Y) = 4\mathbb{E}(C(F(X), G(Y)) - 1, \tag{2}$$

where operator $\mathbb{E}()$ denotes the expectation. An alternative version of (2) was proposed by Genest and MacKay (1986). Let for a given copula $C(x, y)$, $K(t)$ be the cumulative probability function of the random variable $T = C(U_1, U_2)$, where U_1 and U_2 are random variables uniformly distributed on $[0, 1]$. The following relation links a copula with Kendall's τ:

$$\tau = 3 - 4\int_0^1 K(t)dt \tag{3}$$

The special cases of (2) for different specific copulas are given in many papers and textbooks, such as e.g. the book by Nelsen (1999).

Better known is the sample version of τ originally introduced by Kendall in the 1930s. Let $(X_i, Y_i), i = 1, \dots, n$ be a random sample representing n independent pairs of observations of dependent random variables X and Y. An alternative to Kendall's original version of τ_n sample statistic which measures the association between random variables X and Y is given by the following formula proposed by Genest and Rivest (1993).

$$\tau_n = \frac{4}{n-1}\sum_{i=1}^{n-1} V_i - 1, \tag{4}$$

where

$$V_i = \frac{card\{j : X_j < X_i, Y_j < Y_i\}}{n-2}, i = 1, \dots, n. \tag{5}$$

When the vectors $(X_1, X_2, \dots, X_n)$ and $(Y_1, Y_2, \dots, Y_n)$ are mutually independent, the pairs of observations (X_i, Y_i), $i = 1, \dots, n$ are also independent, and the probability distribution of (4) is known. Its expected value is equal to $E(\tau_n) = 0$, and its variance is equal to $Var(\tau_n) = \frac{2(2n+5)}{9n(n-1)}$. For sufficiently large sample size n Kendall's τ_n has the normal distribution with these parameters. It is easy to show that (4) is also a rank statistic.

If we represent the observed values of X and Y in ascending order we get the respective vectors of ranks $R_1, R_2, \dots, R_n$ and $S_1, S_2, \dots, S_n$. Then

$$\tau_n = \frac{P - Q}{\frac{1}{2}n(n-1)} = 1 - \frac{2Q}{\frac{1}{2}n(n-1)} = \frac{2P}{\frac{1}{2}n(n-1)} - 1, \tag{6}$$

where

$$P = \sum_{i=1}^{n}\sum_{j=1}^{n} I(X_i < X_j, Y_i < Y_j) = \sum_{i=1}^{n}\sum_{j=1}^{n} I(R_i < R_j, S_i < S_j), \tag{7}$$

$$Q = \sum_{i=1}^{n}\sum_{j=1}^{n} I(X_i < X_j, Y_i > Y_j) = \sum_{i=1}^{n}\sum_{j=1}^{n} I(R_i < R_j, S_i > S_j). \tag{8}$$

Alternative formulation of the sample version of Kendall's τ is presented in the paper by Denœux *et al.* (2005). Let L_X be a linear order on sample elements induced by the observed values of the random variable X, and expressed by the set of all pairs of observations (x_i, x_j) that belong to the whole set of observations $(x_1, x_2, \ldots, x_n)$, and are such that $x_i < x_j$ for all $i \neq j$. Similarly, L_Y be a linear order on sample elements induced by the observed values of the random variable Y. The number of pairs ordered in the same way by L_X and L_Y is the cardinality of their intersections $|L_X \cap L_Y|$. Then,

$$\tau_n = \tau_n(L_X, L_Y) = \frac{4|L_X \cap L_Y|}{n(n-1)} - 1. \tag{9}$$

Now, let us assume that instead of crisp values of (X_i, Y_i), $i = 1, \ldots, n$ we observe *imprecise* values $(\bar{X}_i, \bar{Y}_i)$, $i = 1, \ldots, n$ where $\bar{X}_i = [X_{i,L}, X_{i,U}]$ and $\bar{Y}_i = [Y_{i,L}, Y_{i,U}]$.

For such observed data the observed value of Kendall's τ will be also imprecise, and given as the interval $\bar{\tau}_n = [\tau_{n,L}, \tau_{n,U}]$, where the values of $\tau_{n,L}$ and $\tau_{n,U}$ are obtained by inserting in (4) instead of V_i the respective values

$$V_{i,L} = \min_{\substack{X_j \in [X_{j,L}, X_{j,U}] \\ Y_j \in [Y_{j,L}, Y_{j,U}]}} \frac{card\{j : X_j < X_i, Y_j < Y_i\}}{n-2}, \tag{10}$$

$$V_{i,U} = \max_{\substack{X_j \in [X_{j,L}, X_{j,U}] \\ Y_j \in [Y_{j,L}, Y_{j,U}]}} \frac{card\{j : X_j < X_i, Y_j < Y_i\}}{n-2}. \tag{11}$$

The optimization tasks defined by (10)-(11) consist in finding such possible crisp observations, which yield minimal and maximal value of Kendall's τ, i.e. $\tau_{n,L}$ and $\tau_{n,U}$. The search space contains all possible crisp observations $([x_{1,L}, x_{1,U}] \times ... \times [x_{n,L}, x_{n,U}]) \times ([y_{1,L}, y_{1,U}] \times ... \times [y_{n,L}, y_{n,U}]) \subset \mathbb{R}^{2n}$. This is a $2n$-dimensional continuous optimization problem with box constraints, which can be solved with an optimization algorithm.

$$\tau_{n,L} = \min_{\substack{x_1 \in [x_{1,L}, x_{1,U}], ..., x_n \in [x_{n,L}, x_{n,U}] \\ y_1 \in [y_{1,L}, y_{1,U}], ..., y_n \in [y_{n,L}, y_{n,U}]}} \tau(x_1, ..., x_n, y_1, ..., y_n) \tag{12}$$

$$\tau_{n,U} = \max_{\substack{x_1 \in [x_{1,L}, x_{1,U}], ..., x_n \in [x_{n,L}, x_{n,U}] \\ y_1 \in [y_{1,L}, y_{1,U}], ..., y_n \in [y_{n,L}, y_{n,U}]}} \tau(x_1, ..., x_n, y_1, ..., y_n) \tag{13}$$

In the case of imprecise (interval) data sample elements are only partially ordered. Let P_X be a partial order of sample elements induced by imprecise observations of random variables $\bar{X}_i$, $i = 1, \ldots, n$. Similarly, let P_Y be a partial order of sample elements induced by imprecise observations of random variables $\bar{Y}_i$, $i = 1, \ldots, n$. A linear order L_X (or L_Y) is a linear extension of P_X (or P_Y) if and only if $P_X \subseteq L_X$ (or $P_Y \subseteq L_Y$). Following Denœux *et al.* (2005) denote the sets of linear extensions of partial orders P_X and P_Y by $\Lambda(P_X)$ and $\Lambda(P_Y)$, respectively. Then, the lower and upper values defining the interval-valued $\bar{\tau}_n$ are calculated from the following formulae proposed in Hébert *et al.* (2003)

$$\tau_{n,L} = \min_{L_X \in \Lambda(P_X), L_Y \in \Lambda(P_Y)} \tau(L_x, L_Y), \tag{14}$$

$$\tau_{n,U} = \max_{L_X \in \Lambda(P_X), L_Y \in \Lambda(P_Y)} \tau(L_x, L_Y). \tag{15}$$

Fig. 1 presents an three possible solutions to optimization problems (12)-(13). Each rectangle represents one of $n = 4$ pairs of interval data $(\bar{x}_i, \bar{y}_i)$. Circles, diamonds and asterisks denote three possible $2n$-dimensional solutions resulting in Kendall's τ equal respectively to 0, $-1/3$ and $2/3$.

The optimization problem defined by (14)-(15) has the form of integer programming, exploiting the fact that Kendall's τ is a rank statistic with a finite number of possible values. This allows for exact calculation of minimal and maximal values of the correlation coefficient for small samples, as it was proposed by Hébert *et al.* (2003). This algorithm is effective only for small samples (less than 10 elements). For larger samples Denœux *et al.* (2005)

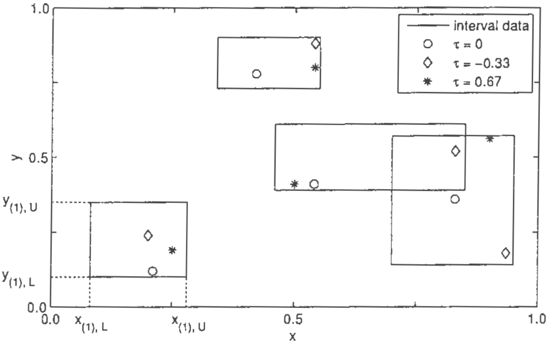

Figure 1: Interval data for $n = 4$ imprecise pairs of observations (denoted by rectangles) and three sets of $2n$-dimensional solutions denoted by circles, diamonds and asterisk yielding respectively τ equal to 0, $-1/3$ and $2/3$

proposed to look for an approximate solution using a Monte Carlo simulation method based on the algorithm by Bubley and Dyer (1998) for generating uniformly distributed linear extensions of a partial order. This algorithm, according to Denœux *et al.* (2005), is effective for moderate sample sizes (less than 30 elements).

3. Heuristic solutions

The results published in Hébert *et al.* (2003) and Denœux *et al.* (2005) show undoubtedly that the calculation of the interval value of Kendall's τ (or similar rank statistics) may by computationally extremely demanding even in the case of moderate samples of imprecise data. Therefore, there is a need either to propose easier to compute approximations or to propose methods for speeding-up computational procedures. These both tasks can be considered jointly, as good approximations can be used as starting points which speed-up optimization procedures that are necessary for the calculation of the interval values of the considered statistic.

It is well-known that certain types of statistical dependence result in specific patterns in observed data. For example, strong positive dependence means that large values of one variable are accompanied by large values of the second one, and small values of one variable are accompanied by small values of the second variable. Therefore, there exists a permutation $i_1, i_2, \ldots, i_n$ of sample elements for which observed values $x_{i_1}, x_{i_2}, \ldots, x_{i_n}$ and $y_{i_1}, y_{i_2}, \ldots, y_{i_n}$ form simultaneously at least nearly decreasing (increasing) sequences. Note that in the case of perfect positive dependence ($\tau = 1$) these sequences will be strictly decreasing (increasing). On the other hand, in the case of strong negative dependence large values of one variable are accompanied by small values of the second one, and small values of one variable are accompanied by large values of the second variable. Therefore, there exists a permutation $i_1, i_2, \ldots, i_n$ of sample elements for which observed values $x_{i_1}, x_{i_2}, \ldots, x_{i_n}$ form at least nearly decreasing (increasing) sequence, and simultaneously, values $y_{i_1}, y_{i_2}, \ldots, y_{i_n}$ form at least nearly increasing (decreasing) sequence. Similarly to the previous case, for perfectly negatively dependent data ($\tau = -1$) both sequences should be strictly decreasing or increasing.

Hryniewicz and Szediw (2008) have found that specific patterns described above that depict strongly correlated observations can be used for the construction of heuristic algorithms that find minimal and maximal values of measures of dependence in presence of interval data. If we look at (10)–(11) we can see that we have to find sets of points $x^{\star}_{(1)}, \ldots, x^{\star}_{(n)}$ and $y^{\star}_{(1)}, \ldots, y^{\star}_{(n)}$ fulfilling the constraints $x^{\star}_{(i)} \in \bar{x}_i = [x_{(i),L}, x_{(i),U}]$ and $y^{\star}_{(i)} \in \bar{y}_i = [y_{(i),L}, y_{(i),U}]$ for all $i = 1, \ldots, n$, and forming at least nearly decreasing (increasing) sequences. There are many ways to achieve this goal. One of these has been proposed in Hryniewicz and Opara (2012b) in a form of an simple algorithm presented below as *Algorithm 1*.

In this algorithm at the first step we order pairs of interval data vectors $(\bar{x}_i, \bar{y}_i), i = 1, \ldots, n$ in such a way that certain points of one variable, e.g. $x^{\star}_{(i)} \in \bar{x}_i = [x_{(i),L}, x_{(i),U}], i = 1, \ldots, n$ (or $y^{\star}_{(i)} \in \bar{y}_i = [y_{(i),L}, y_{(i),U}], i = 1, \ldots, n$) form a non-increasing (non-decreasing) series. Then, at the second step, we find the respective values of the second variable $y^{\star}_{(i)} \in \bar{y}_i = [y_{(i),L}, y_{(i),U}], i = 1, \ldots, n$ (or $x^{\star}_{(i)} \in \bar{x}_i = [x_{(i),L}, x_{(i),U}], i = 1, \ldots, n$) which form a sequence that looks, at least approximately, as non-increasing (non-decreasing). The symbol O^t_p in the the description of the algorithm denotes the set of n sample items ordered according to some chosen values that belong to observed intervals. The lower index p can take three values: u (if the upper limit of the data interval is taken for ordering observations), c (if the center of the data interval is taken for ordering observations), and l (if the lower limit of

Algorithm 1 Minimization (maximization) heuristic—finding $\tau_{n,L}$ ($\tau_{n,U}$)

```
Step 1: order first interval variable to obtain $O_p^t$
$x^\star_{(i)} \leftarrow$ sort first variable $\mathbf{x}_i = [x_{(i),L}, x_{(i),U}]$ decreasing for $i = 1, ..., n$
Step 2: compute values for the second interval variable to obtain $T^t$
$y^\star_{(1)} \leftarrow y_{(1),U}$        $\left(\text{for maximization use } y^\star_{(1)} \leftarrow y_{(1),L}\right)$
for $k = 1, 2, ..., n-$ do
    $y' \leftarrow y^\star_{(k)} - \epsilon$        $\left(\text{for maximization use } y' \leftarrow y^\star_{(k)} + \epsilon\right)$
    $y'' \leftarrow \min\left(y', y_{(k+1),U}\right)$
    $y^\star_{(k+1)} \leftarrow \max\left(y'', y_{(k+1),L}\right)$
end for
return pair of series $\left(x^\star_{(i)}, y^\star_{(i)}\right)$ for $i = 1, ..., n$ as $(O_p^t, T^t)$
```

the data interval is taken for ordering observations). The upper index t can take two values: d (if points are ordered in a non-increasing series), and a (if points are ordered in a non-decreasing series). For example, O_c^d means that sample items are ordered in such a way that the centers of intervals for the considered variable, e.g X, form a non-increasing sequence $x^\star_{(1)}, x^\star_{(2)}, \ldots, x^\star_{(n)}$. The second variable whose values are computed at the second step of our algorithm is denoted by T^t, where the upper index indicates the direction of the trend (d or a). For example, T^a means that the values of the second variable, e.g. Y, calculated at the second step of our algorithm, form a sequence $y^\star_{(1)}, y^\star_{(2)}, \ldots, y^\star_{(n)}$ that is approximately non-decreasing.

For finding the maximal value of the interval-valued $\bar{\tau}_n$ statistic we considered six types of heuristics described as: (O_p^t, T^t) (where $p = u, c, l$ and $t = d, a$). The interpretation of this notation is the following: the set of crisp data points belonging to observed intervals of one variable (X or Y) is generated according to the procedure O_p^t, and the set of crisp data points belonging to observed intervals of the second variable is generated according to the procedure T^t. Because of symmetric usage of variables X and Y we have used altogether 12 heuristics of these types. Moreover, we used heuristics (T^t, T^t) (where $t = d, a$), described together as (T, T), for which the values of both variables have been calculated using only the second step of of our heuristic algorithm (i.e. without pre-ordering of sample items).

For finding the minimal value of Kendall's correlation coefficient of $\tau_{n,L}$ we considered the following heuristics: (O_p^a, T^d), (O_p^d, T^a), and (T^a, T^d). Because of symmetric usage of variables X and Y we have used altogether 14 heuristics of these types.

Generation of data points $(x^*_{(i)}, y^*_{(i)}), i = 1, \ldots, n$ using the heuristics described above is very simple, and can be done even manually. Therefore, in practice one can use all of the considered heuristics in order to find approximate minimal and maximal observed values of Kendall's τ_n.

4. Efficiency of heuristics

4.1. Description of experiments

The efficiency of computational procedures aimed at finding approximately optimal solutions can be evaluated in many ways. The simplest one, consisting in the calculation of approximation error, is not applicable in our problem, as we do not know (except for the cases of very small samples) the exact solutions to the considered optimization problems. Another possibility is to fix the amount of computational effort, and to look at the results of optimization. When comparing two computational procedures, the procedure which for a given computational effort yields on average a wider interval for Kendall's τ is considered more efficient. The computational effort may be measured also in different ways. In our experiments we measured it either by the number of random generations of linear extensions (when we evaluated the algorithm proposed by Denœux *et al.* (2005)) or by the time required to find the approximately optimal interval of τ. The first measure does not depend upon the hardware used in simulation experiments. Thus, the results of simulations where this measure was applied are more informative. The second measure strongly depends upon the performance of the computer used for simulations. Therefore, the results of simulations where this measure was used are less informative, and provide information which is more qualitative than quantitative. In our experiments we also evaluated the speed of convergence of considered algorithms. In those experiments we investigated how the length of the approximately optimal intervals for τ are increasing with increasing amount of computation.

In our evaluation of computational procedures we used Monte Carlo experiments. We generated samples of interval-valued observations, and for those samples we computed the interval values of Kendall's τ or the values of other interesting characteristics. The final results have been found by the averaging of the results obtained for the generated samples.

The generation of interval data has been performed in two steps. At the first step the crisp data were simulated from the following copulas:

- Gaussian - defined by (1),
- Clayton's - defined by

$$C(u,v) = \max\left(\left[u^{-\alpha} + v^{-\alpha} - 1\right]^{-1/\alpha}, 0\right), \alpha \in [-1,\infty) \setminus 0, \qquad (16)$$

- Frank's - defined by

$$C(u,v) = -\frac{1}{\alpha}\ln\left(1 + \frac{\left(e^{-\alpha u} - 1\right)\left(e^{-\alpha v} - 1\right)}{e^{-\alpha} - 1}\right), \alpha \in (-\infty,\infty) \setminus 0,, \qquad (17)$$

- Gumbel's defined by

$$C(u,v) = \exp\left(-\left[(-\ln u)^{1+\alpha} + (-\ln v)^{1+\alpha}\right]^{\frac{1}{1+\alpha}}\right), \alpha \in (0,\infty). \qquad (18)$$

with fixed values of the strength of dependence measured by Kendall's τ. The marginal probability distribution functions of the simulated crisp data were normal, uniform, exponential and Weibull. Then, the crisp data were replaced with intervals of random length and different location around crisp points. The level of imprecision was defined by setting the maximum width of an interval z, measured as the multiplicity of the standard deviation of the marginals which was assumed the same for both variables. The actual width of interval z_a for each data point was generated from the uniform distribution on $(0,z)$. The location of the interval around the generated crisp point was established by the generation of an anchoring point from the uniform distribution $(0,z_a)$, and placing it at the generated crisp value of considered random variable. Fig. 2 presents optimization tasks generated for different copulas and moderate dependence strength (note that Gumbel's copula describes only positive dependencies).

When we evaluate computational procedures using Monte Carlo simulation statistically significant results can be obtained if the number of simulation runs is large enough. In our experiment it could be done only for small and moderate samples. In such cases we used at least 10000 simulation runs. Unfortunately, for some most interesting cases of large samples (200 elements and more) the time for the computation of one simulation run was too long, and the number of simulation runs was limited (usually to 100). In such cases the results of simulations were rather of qualitative character.

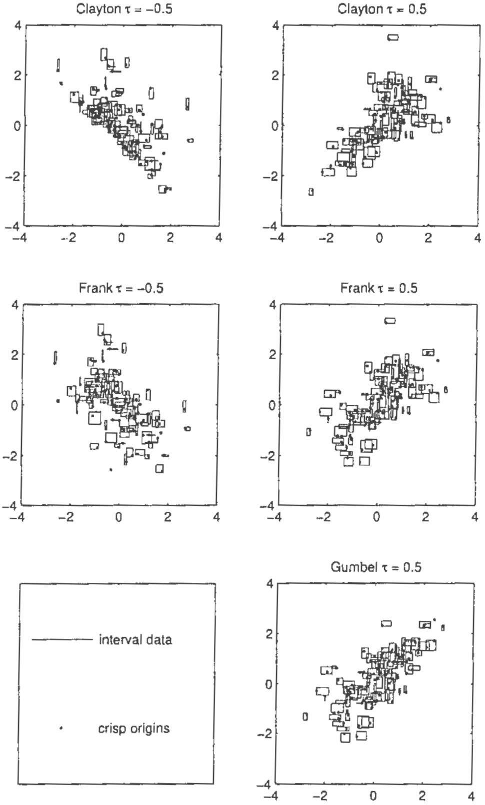

Figure 2: Optimization problems consisting of $n = 100$ pairs of intervals $(\bar{x}_i, \bar{y}_i)$ generated using the crisp origins from various copulas with moderately negative and positive dependencies ($\tau = \pm 0.5$)

4.2. Evaluation of heuristics

In Section 3 we introduced altogether 28 heuristic methods for the calculation of crisp data points $x^\star_{(i)} \in \bar{x}_{(i)} = [x_{(i),L}, x_{(i),U}], i = 1, \ldots, n$ and $y^\star_{(i)} \in \bar{y}_(i) = [y_{(i),L}, y_{(i),U}], i = 1, \ldots, n$ which may be used for the calculation of the approximate lower and upper limits of the interval value of Kendall's τ. One can ask a question whether this seemingly large number of heuristics is necessary. Moreover, it is not clear if all these heuristics are necessary for specific types of dependence between the observed data. To address these issues we conducted extensive simulation experiments. In each simulation run we have calculated all available heuristics and computed respective approximate values of $\tau_{n,L}$ and $\tau_{n,U}$. Then we have verified which heuristics gave the best result. The efficiency of each heuristic was evaluated as the percentage of cases (simulation runs) in which this heuristic yielded the best results.

In the first experiment we considered the case of the Gaussian copula with normal marginals (i.e. the case of the classical bivariate normal distribution) with moderate imprecision (maximal value of the interval equal to one standard deviation of the generated crisp random variable) of generated interval data. The sample size in this case has been set to $n = 20$. The samples have been generated for six types of the strength of dependence: strong positive ($\tau = 0.9$), moderate positive ($\tau = 0.5$), weak positive ($\tau = 0.1$), weak negative ($\tau = -0.1$), moderate negative ($\tau = -0.5$), and strong negative ($\tau = -0.9$). The results of the experiment are presented in Table 1 for the evaluation of the minimal value of the interval Kendall's τ ($\tau_{n,L}$), and in Table 2 for the evaluation of the maximal value of the interval Kendall's τ ($\tau_{n,U}$).

The results presented in Table 1 and Table 2 reveal that the efficiency of the considered heuristics strongly depends upon the strength and the direction of dependence. The most interesting is the case of heuristics with unordered observations. The heuristic (T^a, T^d) is the most efficient among the considered heuristics in finding the minimal value of Kendall's τ_n in the case of strong positive dependence. The situation becomes just opposite in the case of strong negative dependence. When the maximal value of Kendall's τ_n is calculated a similar heuristic (T, T) is the best heuristic for finding this value in the case of strong negative dependence, and completely inefficient in the case of strong positive dependence. What is similar in these both cases it is the fact that these heuristics are the best heuristics in the cases

Table 1: Percentages of best values; Minimum value of τ_n; moderate imprecision; Gaussian copula, normal marginals, $n = 20$

τ	(O^a_u, T^d)	(O^a_c, T^d)	(O^a_l, T^d)	(O^d_u, T^a)	(O^d_c, T^a)	(O^d_l, T^a)	(T^a, T^d)
0.9	8.8	8.3	16.6	5.0	17.4	4.6	39.3
0.5	11.8	12.4	14.2	7.0	14.2	7.0	33.4
0.1	15.9	16.1	17.2	10.5	17.0	10.3	13.0
-0.1	17.0	16.1	17.5	13.4	18.3	11.9	6.8
-0.5	16.7	17.0	17.9	14.8	18.8	14.0	0.7
-0.9	14.0	13.6	12.5	23.5	12.1	24.3	0.0

Table 2: Percentages of best values; Maximum value of τ_n; moderate imprecision; Gaussian copula, normal marginals, $n = 20$

τ	(O^d_u, T^d)	(O^d_c, T^d)	(O^d_l, T^d)	(O^a_u, T^a)	(O^a_c, T^a)	(O^a_l, T^a)	(T, T)
0.9	12.1	24.2	13.9	13.7	12.2	23.9	0.0
0.5	18.5	14.2	16.6	17.4	17.9	14.4	1.0
0.1	17.9	11.6	16.7	16.8	18.7	12.0	6.3
-0.1	17.8	10.7	15.6	15.7	16.6	10.1	13.5
-0.5	14.3	7.1	11.6	12.5	14.2	6.5	33.8
-0.9	16.8	6.6	8.9	8.6	16.7	4.6	39.8

where they are not aimed at. This does not mean, however, that they are better than e.g. solutions obtained by Monte Carlo simulations. The series of observations calculated by the proposed heuristics mimic the real strongly positively data in the case of looking for the maximal value of τ_n, and the real strongly negatively dependent data in the case of looking for the minimal value of τ_n. In such cases the heuristics based on unordered observations are practically useless. When the opposite cases are considered, i.e. the minimal value of τ_n for strongly positively dependent data or the maximal value of τ_n for strongly negatively dependent data, the heuristics based on unordered observations are visibly the best.

In all other cases neither of the considered heuristics is significantly better, even in the case of strong dependence. In the case of weak dependence, both positive and negative, their behavior is similar. Therefore, when the lower and upper approximate values of the interval-valued Kendall's τ_n are calculated it is advised to calculate all pertaining heuristics, and to choose the best one.

Table 3: Percentages of best values; Minimum value of τ_n; small imprecision; Gaussian copula, normal marginals, $n = 20$

τ	(O^a_u, T^d)	(O^a_c, T^d)	(O^a_l, T^d)	(O^d_u, T^a)	(O^d_c, T^a)	(O^d_l, T^a)	(T^a, T^d)
0.9	12.9	12.6	14.2	10.0	14.2	10.3	25.8
0.5	12.9	13.5	13.5	9.9	13.8	10.5	26.1
0.1	15.0	14.8	15.6	11.4	14.8	11.3	17.1
-0.1	11.8	15.7	16.0	11.2	15.7	12.3	13.3
-0.5	16.7	16.6	16.7	13.2	16.4	12.9	7.5
-0.9	16.6	17.2	17.6	14.3	17.3	14.5	2.5

Table 4: Percentages of best values; Maximum value of τ_n; small imprecision; Gaussian copula, normal marginals, $n = 20$

τ	(O^d_u, T^d)	(O^d_c, T^d)	(O^d_l, T^d)	(O^a_u, T^a)	(O^a_c, T^a)	(O^a_l, T^a)	(T, T)
0.9	17.7	14.2	17.7	16.9	17.6	14.2	2.7
0.5	16.5	13.1	16.0	16.5	16.8	12.8	8.3
0.1	16.3	11.5	15.4	15.4	15.6	11.8	14.0
-0.1	14.7	11.7	14.5	15.2	14.9	10.9	18.0
-0.5	13.5	9.8	13.0	13.5	13.9	9.9	26.4
-0.9	14.3	12.3	12.5	12.4	14.6	6.6	26.3

A similar experiments have been performed for the case of small imprecision (maximal value of the interval equal to one tenth of the standard deviation of the generated crisp random variable) of the generated interval data. The results of these experiments are shown in Table 3 for the case of the minimal value of τ_n, and in Table 4 for the case of the maximal value of τ_n.

In general, the results presented in Tables 3 and 4 are similar to those presented in Tables 1 and 2. One can notice however that for less imprecise data the percentage of cases where a given heuristic is the best are more evenly distributed than in the case of more imprecise data. This finding has been also confirmed by experiments whose results are not presented in this paper. For example, in the case of very large imprecision (maximal value of the interval equal to two or more standard deviations of the generated crisp random variable) the best results have been obtained mainly for those heuristics whose performance looks better in the case of moderate imprecision. On the other hand, those heuristics whose performance is worse in the case of

Table 5: Percentages of best values; Minimum value of τ_n; moderate imprecision; Frank copula, exponential marginals, $n = 20$

τ	(O_u^a, T^d)	(O_c^a, T^d)	(O_l^a, T^d)	(O_u^d, T^a)	(O_c^d, T^a)	(O_l^d, T^a)	(T^a, T^d)
0.9	17.5	4.0	31.7	2.1	6.1	10.7	27.9
0.5	16.7	8.7	26.4	3.8	8.0	10.6	25.8
0.1	15.5	15.9	28.2	9.6	15.7	11.1	11.5
-0.1	14.6	19.0	18.6	12.5	18.8	11.8	4.8
-0.5	13.0	20.3	14.6	18.6	20.3	12.7	0.4
-0.9	12.2	17.9	11.8	24.3	14.2	16.9	0.0

Table 6: Percentages of best values; Maximum value of τ_n; moderate imprecision; Frank copula, exponential marginals, $n = 20$

τ	(O_u^d, T^d)	(O_c^d, T^d)	(O_l^d, T^d)	(O_u^a, T^a)	(O_c^a, T^a)	(O_l^a, T^a)	(T, T)
0.9	12.8	22.8	13.2	16.9	17.6	16.7	0.0
0.5	13.7	19.4	13.7	19.3	21.2	12.2	0.5
0.1	14.9	13.8	15.5	17.2	23.0	10.4	5.1
-0.1	15.9	10.7	16.6	15.4	20.2	9.9	11.3
-0.5	17.1	4.8	18.2	7.8	14.4	8.9	28.8
-0.9	20.8	2.4	20.2	4.3	10.7	8.8	32.8

moderate imprecision have been rarely indicated as the best heuristics in the case of large imprecision.

In the experiments described above all data were imprecise to smaller or larger extent. We have also performed experiments when only part of data was imprecise. The results of those experiments have been similar to those presented above for the case of weakly imprecise data.

In the experiments described above we assumed the Gaussian copula and the normal marginal distributions of generated crisp random variables, i.e. the case of the bivariate normal distribution. In the next set of experiments we have investigated cases of different copulas and different marginal distributions (but the same for each copula). In Tables 5 and 6 we present these results for the case of Frank's copula defined by (17) and the exponential (with the parameter of scale equal to one) marginal distributions.

The results presented in Tables 5 and 6 show that the type of a copula and the type of marginal distribution influence the efficiency of different heuristics. However, the differences between the considered case of Frank's

Table 7: Percentages of best values; Minimum value of τ; moderate imprecision; Gaussian copula, normal marginals, $n = 200$

τ	(O^a_u, T^d)	(O^a_c, T^d)	(O^a_l, T^d)	(O^d_u, T^a)	(O^d_c, T^a)	(O^d_l, T^a)	(T^a, T^d)
0.9	0.0	0.0	0.0	0.0	0.0	0.0	100
0.5	0.1	0.1	0.0	0.1	0.0	0.0	99.7
0.1	13.1	14.1	11.0	7.7	11.5	6.2	36.4
-0.1	16.1	17.9	20.8	12.2	19.3	12.1	1.6
-0.5	13.0	14.1	14.5	22.5	13.4	22.5	0.0
-0.9	0.8	1.6	0.9	48.0	0.5	48.2	0.0

Table 8: Percentages of best values; Maximum value of τ; moderate imprecision; Gaussian copula, normal marginals, $n = 200$

τ	(O^d_u, T^d)	(O^d_c, T^d)	(O^d_l, T^d)	(O^a_u, T^a)	(O^a_c, T^a)	(O^a_l, T^a)	(T, T)
0.9	0.5	46.8	2.0	2.2	0.3	48.2	0.0
0.5	13.1	24.1	11.3	15.1	14.4	22.0	0.0
0.1	18.6	13.5	19.4	19.3	15.7	11.2	2.2
-0.1	11.6	7.1	14.1	12.9	12.1	5.9	36.3
-0.5	0.2	0.0	0.0	0.1	0.2	0.0	99.5
-0.9	0.0	0.0	0.0	0.0	0.0	0.0	100.0

copula and the exponential marginals and the previous case of the bivariate normal distribution is not very significant. Many heuristics behave in both cases qualitatively similarly. For example, their performance is the best for strong positive dependence, and the worst for strong negative dependence (or vice versa). One can notice, however, that in the case of Frank's copula and the exponential marginals these trends are more visible than in the case of the bivariate normal distribution.

In the next set of experiments we have investigated if the size of imprecise data influences the efficiency of considered heuristics. In Tables 7 and 8 we present the results of experiments for the case of a relatively large sample ($n = 200$) of moderately imprecise data generated from the bivariate normal distribution.

The results presented in Table 7 and Table 7 show a similar behavior of the proposed heuristics. However, for strongly dependent data only few heuristics are really effective.

4.3. Analysis of the method based on the random generation of linear orders

The heuristics described in Section 3 have been devised in order to improve the search of the interval-valued Kendall's τ described by the interval $[\tau_{n,L}, \tau_{n,U}]$, where $\tau_{n,L}$ and $\tau_{n,U}$ are calculated as the solutions of optimization problems defined in Section 2. In this section we present the results of investigations showing the potential benefits of using our heuristics when the methodology proposed by Denœux *et al.* (2005) is used for finding the approximate solutions of the pertaining optimization problems.

Let $\tilde{z}_i$ where $\tilde{z}_i = [z_{i,L}, z_{i,U}]$, for all $i = 1, \ldots, n$ be the elements of the set of interval-valued data points. For each sample item i of the variable Z we can define a set of its possible ranks $\bar{R}_{Z,i} = [R_{Z,i,L}, R_{Z,i,U}]$ of crisp values $z_i^\star \in [z_{i,L}, z_{i,U}]$ in the Cartesian product $\tilde{z}_1 \times \ldots \times \tilde{z}_n$. The lowest rank $R_{Z,i,L}$ can be computed as $card\{j \neq i : z_{i,L} > z_{j,U}\}$. Similarly, The greatest rank $R_{Z,i,U}$ can be computed as $card\{j \neq i : z_{i,U} < z_{j,L}\}$.

The optimization algorithm for computing the minimal (maximal) value of τ_n begins with the determination of the sets of possible ranks: $\bar{R}_{X,i}$ for the observed imprecise values of the variable X, and $\bar{R}_{Y,i}$ for the observed imprecise values of the variable Y. In the next step the set of points $(x_i^\star, y_i^\star), i = 1, \ldots, n$, such that $x_i^\star \in [x_{i,L}, x_{i,U}]$ and $y_i^\star \in [y_{i,L}, y_{i,U}]$ is determined. These values, and their respective ranks $R_{X,i}^{(1)}$ and $R_{Y,i}^{(1)}$, determine the starting point of the optimization algorithm. Then the algorithm works as it was proposed by Bubley and Dyer (1998).

Let $\gamma_X^{(t)} = (R_{X,1}^{(t)}, \ldots, R_{X,n}^{(t)})$ and $\gamma_Y^{(t)} = (R_{Y,1}^{(t)}, \ldots, Y_{X,n}^{(t)})$ be the sets of ranks assigned to vectors of of imprecise observation at the t-th step of the algorithm. Then with probability 0.5 we have $\gamma_X^{(t+1)} = \gamma_X^{(t)}$ (or $\gamma_Y^{(t+1)} = \gamma_Y^{(t)}$), and with probability 0.5 the following procedure is performed. An index $i^\star$ is randomly chosen from the uniform probability distribution on the set $\{1, 2, \ldots, n-1\}$. Then it is verified if the ranks $R_{X,i^\star}$ and $R_{X,i^\star+1}$ (or $R_{Y,i^\star}$ and $R_{Y,i^\star+1}$) can be interchanged. If it is possible, i.e if $R_{X,i^\star} \in \bar{R}_{X,i^\star+1}$ and $R_{X,i^\star+1} \in \bar{R}_{X,i^\star}$ (or $R_{Y,i^\star} \in \bar{R}_{Y,i^\star+1}$ and $R_{Y,i^\star+1} \in \bar{R}_{Y,i^\star}$, the ranks are interchanged, and a new value of τ is calculated. Then, it is verified if this value is smaller (greater) then the current optimal value of τ. If it is true, the newly calculated value replaces the current value of $\tau_{n,L}$ (or $\tau_{n,L}$).

In the original version proposed by Bubley and Dyer (1998) the algorithm of the random generation of linear extensions of a partial order the algorithm stops when a certain measure of the accuracy (related to the uniformity of the obtained distribution) has been reached. Such measure cannot be directly

Table 9: Optimization of the interval-valued τ. Different starting points

		Heuristic		Middle		Random	
τ	$NGLE$	$\tau_{n,L}$	$\tau_{n,U}$	$\tau_{n,L}$	$\tau_{n,U}$	$\tau_{n,L}$	$\tau_{n,U}$
0.9	0	0.6927	0.9790	0.8227	0.8227	0.7732	0.7732
	10^5	**0.6619**	**0.9806**	0.6874	0.8579	0.6673	0.8195
0.5	0	0.3804	0.5653	0.4823	0.4823	0.4759	0.4759
	10^5	**0.3509**	**0.6672**	0.3728	0.5577	0.3821	0.5450
0.1	0	-0.0298	0.2422	0.0975	0.0975	0.0939	0.0939
	10^5	**-0.0495**	**0.2581**	-0.0007	0.1902	0.0054	0.1794
-0.1	0	-0.2424	0.0281	-0.0977	-0.0977	-0.0954	-0.0954
	10^5	**-0.2588**	**0.0478**	-0.1895	-0.0008	-0.1805	-0.0062
-0.5	0	-0.6568	-0.3817	-0.4834	-0.4834	-0.4675	-0.4675
	10^5	**-0.6682**	**-0.3513**	-0.5590	-0.3740	-0.5414	-0.3725
-0.9	0	-0.9792	-0.6927	-0.8231	-0.8231	-0.7740	-0.7740
	10^5	**-0.9809**	**-0.6612**	-0.8579	-0.6884	-0.8197	-0.6679

applied in our case, as it is not known how this measure of accuracy is related to the accuracy of the optimization procedure. Denœux *et al.* (2005) propose to stop such algorithm when after a certain number of generations the optimal value remains the same. Another possibility, frequently used in the analysis of optimization procedures, is to set a finite number of performed steps, i.e. the number of generated linear extensions, denoted below as $NGLE$.

In the first part of our experiments we analyzed how the method used for obtaining the starting point influences the optimization process. In Table 9 we present the averaged over 10000 simulation runs values of $(\tau_{n,L}, \tau_{n,U})$ calculated for $NGLE = 0$ (i.e. for the starting point), and for $NGLE = 10^5$. For the same moderately imprecise samples of size $n = 20$, generated from a bivariate normal distribution, we calculated optimal values using three methods for setting the starting point: heuristic (described in Section 3), using middle points of data intervals, and using points randomly selected from data intervals. For improved readability, the best results are typed in bold font.

The results presented in Table 9 show without any doubts that the proposed heuristics, used for the calculation of a starting point, significantly improve the optimization procedure based on random generation of linear extensions. It is worth noticing that 'natural 'methods of the generation of starting points (centers of intervals, randomly generated points in intervals) do not guarantee good approximations of the minimal and maximal values of τ even for a relatively large number of generated linear extensions. The intervals $(\tau_{n,L}, \tau_{n,U})$ obtained using such starting points are even visibly worse than the intervals calculated using proposed heuristics. What is also important, when we use the same number of generated linear extensions in the case of starting points computed heuristically the improvement (measured by the width of the computed interval) is significant.

The results presented in Table 9 show only that the proposed heuristics used as the starting points for further optimization are better than central points of intervals or randomly chosen points of intervals. It does not mean, however, that they are sufficiently accurate in all considered cases. For example, it would not be unexpected if the heuristics for finding the minimal value of τ had bad properties for positively dependent data. Similarly, the heuristics for finding the maximal value of τ should not perform well for negatively dependent data. Both conjectures stem from the fact that they were devised to mimic strong positively dependent data (for the calculation of maximum) or strongly negatively dependent data (for the calculation of minimum). The advantage of the heuristics over other considered starting points might be due to the fact that the the heuristically chosen data points are taken from a set of 14 vectors of data. Thus, one might think that this advantage is due to this particular cause. In order to answer this question we performed simulation experiment in which the starting point for further optimization has been the best from among randomly generated sets of m data points. We have considered two values of m: $m = 14$ (i.e. exactly the number of considered heuristics), and $m = 100$. The results of this experiment, for the sample size equal to 20 and moderately imprecise data, are presented in Table 10.

The results presented in Table 10 fully confirm our conjecture. The heuristics considered in this section outperform solutions based on sets of randomly simulated data points only for strongly dependent data. In presence of strong dependence the heuristics are the best for the calculation of the maximum of τ when this dependence is positive, and for the calculation of the minimum of τ when this dependence is negative. This feature is preserved for the moderately dependent data, but only for small number of simulated data points (e.g. $m = 14$). In all remaining cases the random generation of the set of data points, and choosing the best one as the starting point in further optimization, seems to be a better strategy.

In the experiments described above the time for the generation of one

Table 10: Optimization of the interval-valued τ. Random starting points

		Heuristic		14 points		100 points	
τ	$NGLE$	$\tau_{n,L}$	$\tau_{n,U}$	$\tau_{n,L}$	$\tau_{n,U}$	$\tau_{n,L}$	$\tau_{n,U}$
0.9	0	**0.6927**	**0.9790**	0.6946	0.8509	0.6569	0.8844
	10^5	0.6619	**0.9806**	0.6157	0.8763	**0.5901**	0.9035
0.5	0	0.3804	0.5653	0.3869	0.5540	**0.3475**	**0.5903**
	10^5	0.3509	**0.6672**	0.3143	0.6069	**0.2855**	0.6359
0.1	0	**-0.0298**	**0.2422**	0.0155	0.1838	-0.0230	0.2229
	10^5	-0.0495	0.2581	-0.0512	0.2478	**-0.0815**	**0.2770**
-0.1	0	**-0.2424**	**0.0281**	-0.1738	-0.0052	-0.2136	-0.0337
	10^5	-0.2588	0.0478	-0.2378	-0.0622	**-0.2686**	**0.0931**
-0.5	0	**-0.6568**	**-0.3817**	-0.5516	-0.3835	-0.5889	-0.3442
	10^5	**-0.6682**	-0.3513	-0.6051	-0.3116	-0.6349	**-0.2792**
-0.9	0	**-0.9792**	-0.6927	-0.8519	-0.6951	-0.8844	**-0.6567**
	10^5	**-0.9809**	-0.6612	-0.8778	-0.6170	-0.9033	**-0.5890**

linear extension is rather short, so we are able to generate very large numbers of linear extensions, and arrive at good solutions in acceptable time. Thus, for small sample sizes the findings of Denœux *et al.* (2005) seem to be correct. However, the situation changes dramatically if we have large samples of imprecise data. In Table 11 we present the results of investigations whose aim was to evaluate the convergence rate and the execution time of the optimization algorithm. We present the results obtained for the case of Gumbel's copula defined by (18) and the exponentially distributed marginals. We have chosen this bivariate probability distribution as it is very different from the well known bivariate normal distribution (e.g. Gumbel's copula describes only positively dependent data). The sample size is $n = 200$, and the imprecision of data is moderate. Because of relatively long computation time we have been able to perform only a limited number of simulation runs (equal to 100). Thus, the results presented in Table 11 have rather qualitative than quantitative character. The times displayed in the rightmost column of the table represent the average times of execution (in seconds), and have been obtained in the experiment run on a relatively fast Intel Pentium PC machine.

The results presented in Table 11 show very low convergence rate of the optimization algorithm based on the random generation of linear extensions. This is hardly unexpected as for the sample size of $n = 200$ imprecise in-

Table 11: Optimization of the interval-valued τ. Convergence of optimal solutions

		Heuristic		Middle		
τ	$NGLE$	$\tau_{n,L}$	$\tau_{n,U}$	$\tau_{n,L}$	$\tau_{n,U}$	Time [sec]
0.9	0	0.6812	0.8749	0.7432	0.7432	0.
	10^3	0.6588	0.8749	0.6717	0.7473	1.13
	10^4	0.6450	0.8752	0.6502	0.7476	9.88
	10^5	0.6205	0.8764	0.6356	0.7478	100.0

formation the number of possible linear extensions is much larger than the numbers of linear extensions generated in our experiment. However, even for these apparently too small numbers the computation time has grown very rapidly One can predict that for a larger number of linear extensions (e.g. equal to 10^6 or more) the computation time will be prohibitively long for practitioners, and the obtained results still far from the optimal. Therefore, a good choice of the starting point plays a crucial role. Looking at the results presented in Table 11 we see that the comparable times of computations much better results have been obtained when we use heuristics for the computation of the starting point for further optimization.

5. Hybrid algorithm based on the optimization of Pearson's correlation

In Section 2 we recalled a well known fact that the popular statistics used for the analysis of dependent data, Pearson's coefficient of linear correlation, is not a proper measure of dependence. It can be used as a measure of dependence only in the case of the elliptical multivariate probability distributions. Moreover, it is not difficult to show that one can construct a bivariate vector $X, Y = \gamma(X)$, where γ is a highly nonlinear function, such that for the increasing value of Kendall's coefficient of association between X and Y the respective value of Pearson's coefficient of linear correlation will be decreasing. However, for many bivariate distributions, not only belonging to the class of the elliptical distributions, there exists a monotonic relation between values of τ and ρ.

The close formula that links the values of τ and ρ is known for the case of the bivariate normal distribution (described by the Gaussian copula with normal marginals), and is given by the following formula

$$\tau_{Norm} = \arcsin(\rho)/(\pi/2). \tag{19}$$

Table 12: Values of A in the expansion of $\rho(\tau)$

Copula:	Gaussian	Gaussian	Clayton	Clayton	Frank
Marginals:	Exp.	Weibull ($\delta = 0.5$)	Normal	Exp.	Normal
A:	-0.64515	-0.20719	-1	-0.64912	-1

For the most popular copulas the function linking τ and ρ does not exist. We have performed extensive simulation experiments in order to establish a similar relationship. We fitted an approximate function of the following form

$$\rho_a(\tau) \approx \sum_{i=1}^{8} a_i f_i(\tau). \tag{20}$$

where $a_8 = 1$, $f_1(\tau) = \tau - \tau^9$, $f_2(\tau) = \tau^2 - \tau^8$, $f_3(\tau) = \tau^3 - \tau^9$, $f_4(\tau) = \tau^4 - \tau^8$, $f_5(\tau) = \tau^5 - \tau^9$, $f_6(\tau) = \tau^6 - \tau^8$, $f_7(\tau) = \tau^7 - \tau^9$, $f_8(\tau) = \tau^8[1 + (1 - \tau)(A + 1)/2]$, and $A = \rho(-1)$ is the value of Pearson's ρ in the case of full negative dependence, i.e. for $\tau = -1$ which is equal to -1 in the case of symmetric marginals or estimated from Monte Carlo experiments for asymmetric marginals. Exemplary values of A are given in Table 12.

The coefficients a_i for different combinations (copula, marginal distributions) have been evaluated using Monte Carlo simulations. They are given for some of those combinations in Table 13. The approximation given by (20) seems to be good enough for practical applications. The largest absolute difference between the empirical values $\rho(\tau)$ and the approximate values $\rho_a(\tau)$ is smaller than 0.01.

Visual analysis the function $\rho_a(\tau)$ for the cases considered in Table 13 shows that this function is monotonically increasing. Therefore, larger values of τ correspond to larger values of ρ. Thus, the set of crisp data points $x^\star_{(1)}, \ldots, x^\star_{(n)}$ and $y^\star_{(1)}, \ldots, y^\star_{(n)}$ fulfilling the constraints $x^\star_{(i)} \in \bar{x}_i = [x_{(i),L}, x_{(i),U}]$ and $y^\star_{(i)} \in \bar{y}_i = [y_{(i),L}, y_{(i),U}]$ for all $i = 1, \ldots, n$ that minimizes (maximizes) Pearson's coefficient of linear correlation should be close to a similar set that minimizes (maximizes) the value of τ. This hypothesis is the base of an algorithm for the calculation of a new starting point for further computation of $\tau_{n,L}$ and $\tau_{n,L}$.

Formally, the optimization problem can be formulated as

$$\rho_L = \min_{\substack{x_i \in [x_{i,L}, x_{i,U}] \\ y_i \in [y_{i,L}, y_{i,U}]}} r(x_1, \ldots, x_n, y_1, \ldots, y_n), \tag{21}$$

Table 13: Coefficients of the function $\rho(\tau)$

Copula Marginals	a_1	a_2	a_3	a_4	a_5	a_6	a_7
Gaussian Exp.	0.4354	-0.2258	-0.4003	-0.1768	-0.3293	0.4129	1.2673
Gaussian Weibull	-5.5561	-2.0679	3.1017	0.4186	-0.5554	0.7901	0.7633
Clayton Normal	7.4643	0.3090	-5.6402	-0.3099	0.9786	0.1325	1.3693
Clayton Exp.	-4.9277	3.5989	5.5548	-3.2707	-3.4342	1.2717	1.9262
Frank Normal	0.3373	-0.0055	-0.2050	-0.0040	-0.4111	-0.0010	1.4308

$$\rho_U = \max_{\substack{x_i \in [x_{i,L}, x_{i,U}] \\ y_i \in [y_{i,L}, y_{i,U}]}} r(x_1, \ldots, x_n, y_1, \ldots, y_n). \tag{22}$$

where $r(x_1, \ldots, x_n, y_1, \ldots, y_n)$ is Pearson's coefficient of linear correlation between vectors $x_1, \ldots, x_n$ and $y_1, \ldots, y_n$, described in every statistics textbook.

For solving optimization problems (21)-(22) we used a general purpose constrained optimization code developed by Powell in two versions: COBYLA and BOBYQA. The description of the algorithm used in these constrained optimization routines can be found in the paper Powell (1998). COBYLA minimizes an objective function $F(x_1, ..., x_N)$, $F : \mathbb{R}^N \to \mathbb{R}$ subject to M inequality constraints of the form $g_i(x_1, ..., x_N) \geq 0, i = 1, ..., M$. BOBYQA is based on the same algorithm, but for constraints in the form of intervals.

We have performed numerical experiments comparing the performance of these both routines for solving the problems (21)-(22). In the case of small and moderate sample sizes the performance of both routines was similar. When the number of calls of the objective function was used as the measure of performance COBYLA was better. However, when the computation time was analyzed BOBYQA was slightly faster. Because the time of computation is very important in our analysis in further experiments we used that code.

By computing the approximations given by (21)-(22) we do not escape from the problem of the sample size. One can notice that we have to solve constrained optimization problem in $2n$ dimensions. The time of computa-

Table 14: Optimization of the interval-valued τ. Hybrid algorithm

	Hybrid			Heur+LinEx		
τ	$NGLE$	$\tau_{n,L}$	$\tau_{n,U}$	$NGLE$	$\tau_{n,L}$	$\tau_{n,U}$
0.9	0	**0.5086**	**0.9925**	0	0.6959	0.9788
	10^5	**0.4715**	**0.9935**	$2 \cdot 10^5$	0.6653	0.9805
0.5	0	**0.1844**	**0.7416**	0	0.3878	0.6608
	10^5	**0.1525**	**0.7600**	$2 \cdot 10^5$	0.3571	0.6716
0.1	0	**-0.1905**	**0.3421**	0	-0.0181	0.2530
	10^5	**-0.2116**	**0.3800**	$2 \cdot 10^5$	-0.0408	0.2691
-0.1	0	**-0.3794**	**0.1893**	0	-0.2323	0.0385
	10^5	**-0.4059**	**0.2174**	$2 \cdot 10^5$	-0.2490	0.0582
-0.5	0	**-0.7429**	**-0.1858**	0	-0.6562	-0.3808
	10^5	**-0.7609**	**-0.1535**	$2 \cdot 10^5$	-0.6680	-0.3495
-0.9	0	**-0.9925**	**-0.5085**	0	-0.9805	-0.6976
	10^5	**-0.9935**	**-0.4710**	$2 \cdot 10^5$	-0.9821	-0.6666

tion of (21)-(22) grows rapidly, and is incomparably long in comparison to our heuristic solution. However, if we use the data points obtained by the minimization (maximization) of Pearson's ρ as the starting point for further optimization using the algorithm of random generation of linear extensions the results are astonishingly good. In Table 14 we present a comparison of the performance of two algorithms whose computation times are comparable. The data ($n = 20$) have been generated from the bivariate normal distribution with a moderate level of imprecision. After performing the search of the starting point by solving (21)-(22) we tuned the results by applying 10^5 random generations of linear extensions. We compared the results of this experiment with the results when only random generations of linear extensions were used for the optimization purpose. In this second case we used $2 \cdot 10^5$ random generations of linear extensions. These two values were chosen in order to have similar times of computation.

The results presented in Table 14 reveal a significant advantage of the newly proposed method over the methods described in the previous section. It is quite clear that it is beneficial to devote more time for finding a better starting point than to choose a starting point without special computational effort, and then to run optimization procedure for longer time.

Other experiments with the hybrid optimization algorithm revealed that using heuristics for the calculation of the starting point in the first optimization step is not advantageous anymore, even when the heuristics give better approximation for the minimal and maximal value of τ. The reason of this somewhat unexpected feature stems from the characteristics of the optimization routine. A close look at the data points of the optimal solutions shows that these points are usually on constraints or close to them. In such a case a general purpose constrained optimization routine, like BOBYQA or COBYLA, do not perform well (they spend most of the computational time for coping with the problem of constraints violation). For this reason apparently worse starting points, but situated far from the constraints, may be better than the points closer to the optimal ones.

6. Computing interval-valued τ using evolutionary algorithms

As mentioned in section 2 optimization tasks defined by (10)-(11) consist in finding such possible crisp observations, which yield minimal and maximal value of Kendall's τ, i.e. $\tau_{n,L}$ and $\tau_{n,U}$. One way to approach this problem using one of many available general-purpose optimizers. In this section we first try to characterize the properties of the optimization problem. Next, we evaluate in which cases and to what extend heuristics proposed in section 3 may be helpful for problems (10)-(11).

6.1. Optimization problem analysis

Optimization problem of finding minimal and maximal value of Kendall's τ can be stated in two forms: discrete (14)-(15) in the set of linear extensions of partial orders as well as continuous (12)-(13) in the $2n$-dimensional space $\mathbb{R}^{2n}$ with box constraints. Discrete optimization problems are highly dependent on the representation of data and usually require using dedicated methods, such as algorithm by Bubley and Dyer (1998). Therefore, in this section we concentrate on the continuous variant, since a variety of elaborated and reliable optimizers are available. Choosing among them is easier, when one is aware of the properties of optimized function τ.

Sample version of Kendall's τ, which is optimized in problems (12) and (13) has some interesting characteristics. To visualize them we took a data set of $n = 50$ interval pairs generated using Clayton copula and moderate dependence and plotted the value of Kendall's τ in a grid spanned over all feasible values of i-th observation (x_i, y_i). The remaining 49 crisp samples were chosen according to a uniform random distribution within their feasible values. The upper and lower pairs of plots in the Fig. 3 differ only by the choice of random seed for generating the remaining samples.

Analysis of Fig. 3 and definition of the Kendall's τ for a sample (6) leads to conclusions with important implications for the proper choice of optimization algorithm. First, Kendall's τ takes only a finite number of values. The search space is hence partitioned into many $2n$-dimensional subintervals each having a constant value of τ. This prevents the use of gradient- and Hessian-based optimization methods, since all derivatives are nearly everywhere equal to zero. Next, boundary of each subinterval consists of hyperplanes parallel to the coordinates of the search space. Such property is exploited by some optimization procedures, for instance by evolutionary algorithms using binomial crossover. This also suggests that the taxi-cab metric may be more appropriate then Euclidean one to measure distances between points in the search space. Moreover, Kendall's τ is non-convex and usually multimodal. Consequently, to obtain good results one should choose some derivate-free global optimizer such as (multistart) direct search procedure or an evolutionary algorithm. Special care should be also given to the constraint handling technique, as subintervals having minimal or maximal values on τ are often adjacent to constraints, specially for large values of search space dimension $d = 2n$. Finally, the global optimum is not unique and the set of points with minimal or maximal value of τ may be non-convex or even not connected.

6.2. Simulation results with Differential Evolution

To compute minimal and maximal values of Kendall's τ we chose Differential Evolution algorithm (DE) introduced by Storn and Price (1997). This is a simple, yet effective real-parameter global optimization procedure, which has been applied and further developed over the last decade, see (Price *et al.*, 2005) and (Nerri and Tirronen, 2010) for overview.

Differential Evolution is a population-based search procedure, as in each iteration a population (set) consisting of Np individuals (vectors in $\mathbb{R}^d$) is processed. In our case dimensionality of the search space d equals $2n$. The population-based character of optimization algorithm allows for straightforward application of heuristic solutions, as each of them may be used as one point in the initial population.

In this paper we use a variant of DE called DE/rand/∞/bin, whose description and discussion can be found in (Opara and Arabas, 2010). In this algorithm, first population is initialized, either randomly or with use of

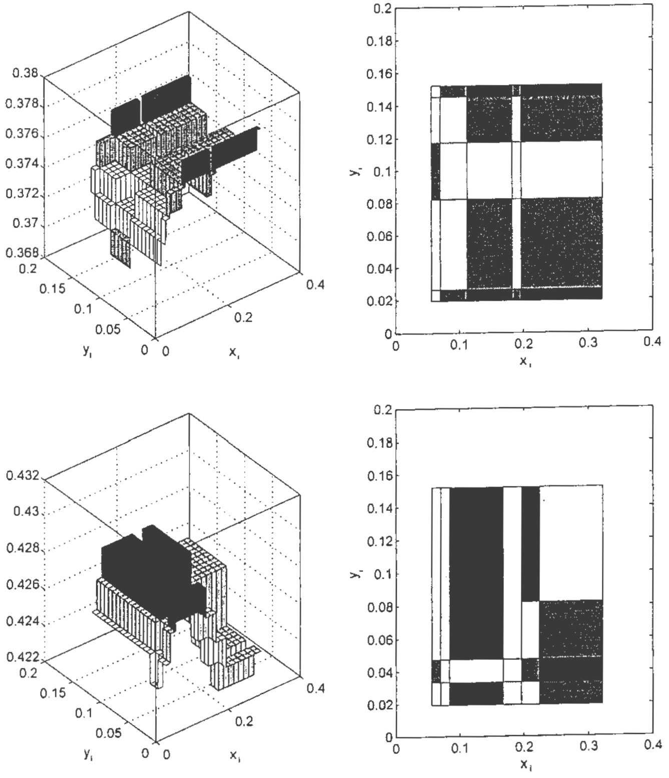

Figure 3: Kendall's τ plotted in a grid spanned over possible values of i-th observation (x_i, y_i) for randomly chosen values of other observations (different for upper and lower pair of figures)

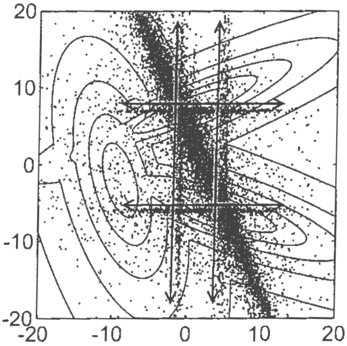

Figure 4: Set of points generated during a single run of DE/rand/∞/bin algorithm for three-modal test function; arrows indicate artifacts introduced by binomial crossover

heuristics. Then in a loop a new population is created from the current one until the stopping condition is not met. New vectors are created by with use of multivariate normal distribution and binomial crossover, while succession is based on a binary tournament.

The DE/rand/∞/bin algorithm uses binomial crossover. This is a classical genetic operator, which out of two vectors $(x_1, ..., x_d)$ and $(y_1, ..., y_d)$ creates another one $(z_1, ..., z_d)$, whose j-th element is with probability Cr set to the value y_j and with probability $1 - Cr$ to the x_j for $j = 1, ..., d$. Binomial crossover influences the dynamics of the optimization procedure by decorrelating the distribution of newly created vectors and thus promoting search along directions parallel to the axis of the coordinate system. Fig. 4 examplifies this phenomenon by showing distribution of all points generated during a single run of DE/rand/∞/bin for a three-modal objective function. Vertical and horizontal arrows align with the search directions introduced by binomial crossover.

Arabas *et al.* (2010) show that in case of box constraints the choice of constraint handling technique significantly affects overall performance of the Differential Evolution algorithm. In problem analyzed in this paper both heuristic values and solutions often lie near the boundaries. We decided to use reflection as a constraint handling technique. The stopping criterion was set to exceeding the maximal number of function evaluations or reaching a situation when all points in a population have the same value of Kendall's τ. In the latter case algorithm was reinitialized with appropriately decreased number of maximal function evaluations and restarted. We chose the following setting of parameters: population size $Np = 4n$, where n is the number of observations $(\tilde{x}_i, \tilde{y}_i)$, scaling factor $F = 0.9$ and crossover probability $Cr = 0.9$. In case of initialization with use of heuristics, first 16 points in the population were replaced with the heuristic values. It is noteworthy that the number of iterations of the algorithm is equal to $FEs/2n$ where, FEs denotes the number of function evaluations. This measure corresponds to the previously discussed number of generated linear extensions (NGLE), since for each extension one needs to compute one τ value.

6.3. Introducing problem-specific knowledge

Using general-purpose optimization algorithms for finding minimal and maximal value of Kendall's τ is very easy to implement, as computations are performed using natural representation of a problem (12)-(13). Using global optimization methods also leads to improvement over pure Monte Carlo sampling, see (Hryniewicz and Opara, 2012a) for comparison of CMA-ES (Covariance Matrix Adaptation Evolutionary Strategy), SA (Simmulated Annealing) and GA (Genetic Algorithm), the last two taken from Matlab Optimization Toolbox.

To further improve performance of optimization procedures one may introduce problem-specific knowledge into the optimization task. Such transformations are effort-consuming, as they require implementing specialized subroutines. Hryniewicz and Opara (2012a) show that they can improve performance of both optimization method and pure Monte Carlo sampling. The general idea behind this approach is to use highly elaborated, general-purpose optimizers encode the problem-specific knowledge into the definition of the search space or objective function.

Kendall's τ takes only finite number of values, which are, moreover, invariant to order-preserving transformations of the input vectors. This means, that bounds of every interval can be substituted with bounds defined by minimal and maximal ranks, which a point within this interval can take. Consider a vector $(x_i, y_i) \in (\tilde{x}_i, \tilde{y}_i) = [x_{i,L}, x_{i,U}] \times [y_{i,L}, y_{i,U}]$ from interval i. Its minimal rank $R_{x_i,L}$ along axis x is equal to the number of intervals, whose upper bounds are lower than lower bounds of the i-th interval, while its maximal rank $R_{x_i,U}$ is equal to the number of intervals, whose lower bounds are lower than the upper bound of i-th interval.

$$R_{x_i,L} = card\{j \neq i : x_{j,U} < x_{i,L}\} \tag{23}$$

$$R_{x_i,U} = card\{j \neq i : x_{j,L} < x_{i,U}\} \tag{24}$$

Analogous equations can be written for ranks with respect to values of y. To obtain continuous optimization problem we use the following transformation.

$$\bar{x}'_i = \left[\frac{r_{x_i,L}}{n}, \frac{R_{x_i,U}+1}{n}\right] \tag{25}$$

$$\bar{y}'_i = \left[\frac{r_{y_i,L}}{n}, \frac{R_{y_i,U}+1}{n}\right] \tag{26}$$

Division by n ensures that both axes x and y range from 0 to 1. Fig. 5 presents an example of transforming real-valued problem $(\bar{x}_i, \bar{y}_i), i = 1, ...n$ to its rank representation $(\bar{x}'_i, \bar{y}'_i), i = 1, ...n$ together with points obtained through heuristics. In the transformed problem the area of intersections of two rectangles relatively increases. Sizes of rectangles are set onto a comparable scale. It may also happen that all vectors in a given rectangle can take only one value of the rank, i.e. $R_{x_i,L} = R_{x_i,U}$. This means that the minimal and maximal value of τ do not depend on the choice of this vector. Consequently, such intervals can be removed from the search procedure, which in turn decreases the dimension of the search space.

Hryniewicz and Opara (2012a) show that use of transformations (23)-(26) improves performance of different optimization algorithms. Finally, is should also be noted, that for each transformation obtained applying formulas (23)-(26) there exist a unique reverse transformation.

6.4. Convergence curves

Ideally, optimization algorithm should provide good solutions in short time. Those two characteristics can be visualized in convergence curves, which present value of the best solution achieved within a given computational time (typically measured in the number of function evaluations FEs). In an example shown in Fig. 6 each curve describes one run of an algorithm. Horizontal distance between two curves presents the difference in time required to achieve certain accuracy, while vertical distance describes a difference in quality between best solutions obtained within a certain computational budget. The most common approach to compare algorithms consists

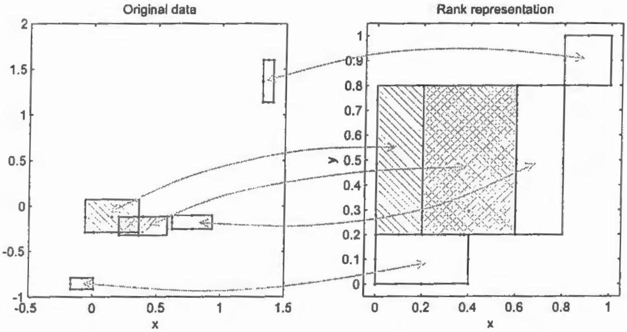

Figure 5: Example of transformation between original interval data and its rank-based representation

in fixing maximal computational budget and comparing final optimization errors. On the other hand Hansen *et al.* (2009) argue that it is not clear how much more difficult is it to reach lower error and suggests comparing expected running times of optimizers as a more interpretable measure. They also propose an aggregation methods for results obtained through many runs over many optimization problems. These methods require, however, knowledge of the globally best value of objective function and yield some problems with estimation of expected runtime, as discussed in (Opara and Arabas, 2011). In this section, we constrain to qualitative analysis of convergence curves only.

Fig. 7 presents convergence curves obtained for finding minimal and maximal value of Kendall's τ for Frank copula and different dependence strengths: strong (crisp origins of interval data have $\tau = \pm 0.9$), moderate ($\tau = \pm 0.5$) and weak ($\tau = \pm 0.1$). The thick lines in the plot depict the median value out of convergence curves obtained within 15 independent runs of the optimization algorithm. Similarly, the shaded area behind it represents the interquartile range between convergence curves, which provides information about the variability of best results between independent runs.

Graphs presented in Fig. 7 show typical properties of the convergence curves. We used $n = 100$ pairs of intervals, whose origins were generated with Frank copula. Nevertheless, our observations show that characteristics

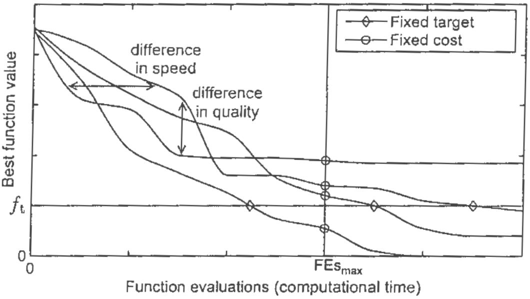

Figure 6: Example of analysis of convergence curves

depicted in this example hold for all investigated copulas and other sample sizes as well. For weak dependencies ($|\tau| < 0.4$) there is practically no difference between optimization algorithm initialized with and without heuristics. For moderate dependencies ($0.4 < \tau < 0.7$) starting with heuristic gives some improvement compared to random initialization, however on a longer run randomly initialized algorithm often yields better solution. For strong dependencies ($0.7 < |\tau|$) the initial heuristic solution is very good and optimization algorithm is usually not able to improve it at all, therefore, the respective convergence curve is a horizontal line.

Initializing population of evolutionary algorithms with heuristic solutions may, however hinder its global optimization properties. In Fig. 8, in case of minimization we observe, that randomly initialized algorithm at first gets poorer results than the one initialized with use of heuristics. After a few thousand function evaluations this situation reverses, since the algorithm initialized with heuristic solutions starts improving value of the best found result on average only between 10 and 20 thousand function evaluations. It is supposedly a result of premature convergence of the optimization procedure. Heuristics initialize algorithm in a local minimum, which becomes a strong attractor to the whole population of an evolutionary algorithm, which relatively increases selection pressure. To leave such minimum evolutionary algorithm must randomly create a vector witch has better value of objective function than the value already optimized by means of heuristics. This may take considerable amount of time, in which there is no improvement of the

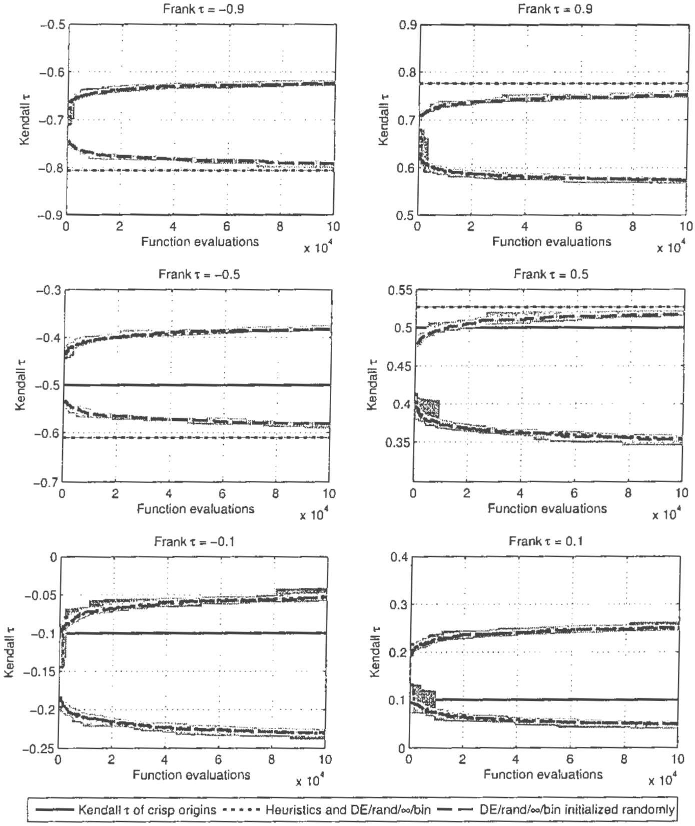

Figure 7: Comparison of convergence curves for data generated from Frank copula with different dependence strengths

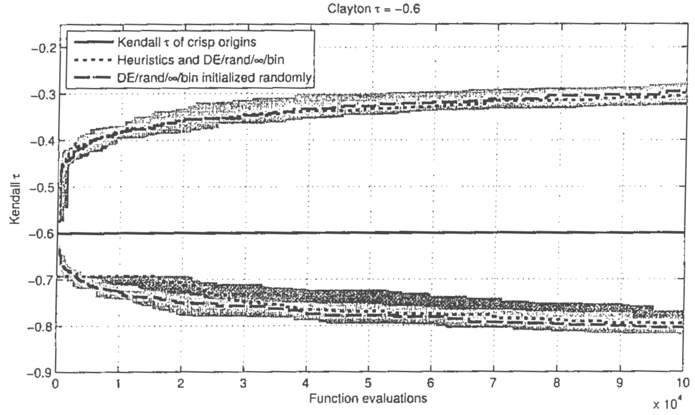

Figure 8: Comparison of convergence curves for $n = 100$ interval observations based on the Clayton copula

objective function and convergence curve is constant. The same phenomenon is visible for strong dependencies , however in this case the heuristic guess seems to be much better than random one. This also means that it is very difficult for an evolutionary algorithm to leave such (potentially local) optimum in search for more promising areas and running it gives little or no improvement. On the other hand, randomly initialized evolutionary algorithm needs high computational budget to get similar or better results than the heuristic solutions.

In case of negative (positive) dependence heuristics for finding maximal (minimal) value of τ gives quite poor results. This does not affect the performance of optimization procedure, as those poor vectors are rejected by evolutionary algorithm within a few first iterations. Nevertheless, devising appropriate heuristics to handle this case is an important open problem. To sum up, heuristics proposed in section 3 prove to be useful for fining minimal value of τ for strong and moderate negative dependencies and for finding maximal value of τ for strong and moderate positive dependencies, which is summarized in Table 15.

Table 15: Guide for using heuristics (Heur.) or randomly initiated global optimization algorithm (Opt.) for finding minimal and maximal value of Kendall's τ for interval data

	Negative dependence			Positive dependence		
	weak	moderate	strong	weak	moderate	strong
$\tau_{n,L}$	Heur.	Heur. and Opt.	Opt.	Opt.	Opt.	Opt.
$\tau_{n,U}$	Opt.	Opt.	Opt.	Opt.	Heur. and Opt.	Heur.

7. Conclusions

In this paper we have presented a comprehensive analysis of computational problems related to the computation of the interval-valued Kendall's τ dependence (association) measure when statistical data are imprecise and given in the form of intervals. First, we have shown that some heuristics, originally proposed in (Hryniewicz and Opara, 2012a,b) can be efficiently used for the generation of a starting point of the optimization procedure based on the random generation of linear extensions described in the paper by Denœux *et al.* (2005). It appears that the heuristic solutions yield very good approximations of the maximum value of τ in the case of strong positive dependence, and of the minimum value of τ in the case of strong negative dependence. In the similar cases of moderate dependence the approximations are not so accurate, but still can be used as starting points for further search of the minimal and maximal values of τ.

When multivariate statistical data are described by the most popular copulas such as Gaussian (Normal), Clayton, Frank and Gumbel, much better approximations can be proposed. They are computed by the routine for the constrained optimization of Pearson's linear correlation coefficient. The hybrid algorithm consisted of minimization (maximization) of Pearson's ρ, and then the application of the algorithm based on the random generation of linear extensions yields extremely good approximations which seem to be close to the exact ones even in the case of weak dependence.

Finally, the usage of the randomly initialized general-purpose optimization procedure has been considered. The performance of such algorithms has been considered when it is possible to introduce problem-specific knowledge into the search space definition. The problem of finding heuristics useful for maximal values of τ for negative dependencies and minimal values of τ for positive dependencies unfortunately still remains an open question. Because of the properties of the considered algorithms it is not advised to use heuristics in order to initialize optimization algorithm, since this bears risk of premature convergence. Consequently the algorithm initialized with heuristics performs worse than the same algorithm that is initialized randomly.

In the paper we have considered only the case of imprecise data of the interval character. One can easily extend the obtained results for fuzzy data, as fuzzy numbers representing such imprecise data points can be expressed as nested sets of intervals. For instance, one can start the optimization procedure with the shortest intervals, and then to use the data points for which minimal (maximal) values of τ have been obtained as the starting points for the optimization needed for the computation of the next (wider) interval of τ. Approach and algorithms proposed in this contribution can be also directly used for the calculation of other rank-based measures of monotonic dependence in presence of imprecise data.

References

Arabas, J., Szczepankiewicz, A., Wroniak T., 2010. Experimental comparison of methods to handle boundary constraints in differential evolution. In: LNCS 6239, Proc. of PPSN'10(2), 411–420.

Bubley, R., Dyer, M., 1998. Faster random generation of linear extensions. In: Proc. 9th Annu. ACM-SIAM Symp. on Discrete Algorithms, San Francisco, CA, 175–186.

Couso, I., Sanchez, L., 2011. Mark-recapture techniques in statistical tests for imprecise data. International Journal of Approximate Reasoning 52, 240-260.

Denœux T., Masson M.-H., Hébert P.A., 2005. Nonparametric rank-based statistics and significance tests for fuzzy data. Fuzzy Sets and Systems 153, 1–28.

Embrechts, P., Lindskog, F., McNeil, A., 2001. Modelling Dependence with Copulas and Applications to Risk Management. In: Svetlozar T. Rachev (Ed.), Handbook of Heavy Tailed Distributions in Finance, Elsevier, Amsterdam, 329–384 (also available as a Technical Report of ETHZ, Zurich, 2001).

Genest C., McKay R.J., 1986. The joy of copulas: Bivariate distributions with uniform marginals. American Statistician 88,1034–1043.

Genest C., Rivest L.-P., 1993. Statistical inference procedures for bivariate Archimedean copulas. Journal of the American Statistical Association 88, 1034–1043.

Gil, M.A., Hryniewicz, O., 2009. Statistics with imprecise data. In: Robert A. Meyers (Ed.): Encyclopedia of Complexity and Systems Science. Springer, Heidelberg, 8679-8690.

Hansen N., Auger A., Finck S., and Ros, R., 2009. Real-parameter black-box optimization benchmarking 2009: Experimental setup. Technical Report RR–6828, INRIA, 2009.

Hébert P.-A., Masson M.H., Denœux T., 2003 Fuzzy rank correlation between fuzzy numbers. In: Proc. of IFSA World Congress, Istanbul, 224–227.

Hryniewicz, O., 2004. Measures of Association for Fuzzy Ordered Categorical Data. In: Lopez-Diaz M., Gil M.A., Grzegorzewski P., Hryniewicz O., Lawry J. (Eds.): Soft Methodology and Random Information Systems. Springer Verlag, Berlin, Heidelberg, New York 2004, 503-510.

Hryniewicz, O., 2006. Goodman-Kruskal */gamma* measure of dependence for fuzzy ordered categorical data. Computational Statistics & Data Analysis 51, 323-334.

Hryniewicz O., Opara K., 2012. Computation of the measures of dependence for imprecise data. In: Atanassov K.T., Baczynski M., Drewniak J., Kacprzyk J., Krawczak K., Szmidt E., Wygralak M., Zadrozny S.(eds): New Developments in Fuzzy Sets, Intuitionistic Fuzzy Sets, Generalized Nets and Related Topics. Volume I: Foundations. SRI PAS, Warszawa (in press).

Hryniewicz O., Opara K., 2012. Efficient calculation of Kendall's τ for interval data (in press).

Hryniewicz, O., Szediw, A., (2008). Fuzzy Kendall τ statistic for autocorrelated data. In: Dubois D., Lubiano M.A., Prade H., Gil M.A., Grzegorzewski P., Hryniewicz O. (eds): Soft Methods for Handling Variability and Imprecision. Springer, Berlin, 155–162.

Kruse, R., Meyer, K.D., 1987. Statistics with Vague Data. Riedel, Dodrecht.

Nelsen R.B., 1999. Introduction to Copulas. Springer, New York.

Neri, F. and Tirronen, V., 2010. Recent advances in differential evolution: a survey and experimental analysis. Artificial Intelligence Review 33, 61–106.

Opara K. and Arabas J., 2010. Differential mutation based on population covariance matrix. LNCS 6238, Proc. of PPSN'10(1):114–123.

Opara K., Arabas J., 2011. Benchmarking Procedures for Continuous Optimization Algorithms. Journal of Telecommunications and Information Technology 4/2011, 73-80.

Powell, M. J. D., 1998. Direct search algorithms for optimization calculations. Acta Numerica 7, 287-336.

Price, K. V., Storn, R. M. and Lampien, J. A., 2005. Differential evolution a practical approach to global optimization. Natural computing series, Springer.

Sklar A., 1959. Fonctions de rpartitions a n dimensions et leur marges. Publications de lInstitut de statistique de lUniversite de Paris 8, 229–231.

Storn R. and Price, K., 1997. Differential Evolution — a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11, 341-359.

Viertl, R., 2007. Statistical Methods for Fuzzy Data, John Wiley, London.