MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **75** (2016) 485-510

ISSN 0340 - 6253

An Attempt to Understand Noisy Posets

Rainer Bruggemann[1], Lars Carlsen[2]

1) Leibniz-Institute of Freshwater Ecology and Inland Fisheries, Department Ecohydrology Berlin, Müggelseedamm 310, Germany. brg_home@web.de
2) Awareness Center, Linkøpingvej 35, Trekroner, DK-4000 Roskilde, Denmark

(Received October 7, 2015)

Abstract

The role of data noise is analyzed for a pair of objects and m indicators (m > 1) with respect to order relations. A general probability scheme is developed and by specifying a noise-model and a distribution for noisy values explicit expressions are derived. These expressions allow an a priori estimation at which noise level effects the order relation between the two objects are affected. It turns out that a useful quantity can be introduced, by which it can be decided at which level of noise perturbations on the set of order relations can be expected. We call this quantity the "crucial noise". Some toy examples and an example out of environmental chemistry are discussed.

1. Introduction

The application of partial order on data matrices for the purpose of data analyses found some interest in the scientific community. Nevertheless, a basic criticism concerns the data uncertainty which is not adequately accounted by partial order concepts up to now. Several publications put their fingers on that point (for example already 1998 an important paper appeared, where the problem of noise is discussed [1,2], or for a more recent paper, see [3]. However, it appears difficult to apply the concepts in a simple framework, which can be used by everybody. In a recent paper [4] an approach is presented, how by Monte-Carlo-simulations applying MCbasedHD8_2 of the software package PyHasse [5] the effect of noisy data sets on data-driven partial orders (also known as Hasse diagram technique, HDT) can be studied. Whereas basically different distributions and models of noise can be simply

established within the logical scheme described in the above mentioned paper [4], a main problem was let unanswered: How can “noise” be related to the stability of certain order theoretical configurations.

Some preliminary considerations have quickly shown that the establishment of certain stability ranges for a complete set of objects, with a number of objects >> 2 is by far too ambitious, especially since the transitivity axiom needs to consider a multitude of objects simultaneously. In this paper we show, how, starting from very simple systems nevertheless some predictions for more complex systems can be done, before an explicit calculation by MCbasedHD8_2 is performed. Hence, this paper should be understood as a first attempt to understand - on a theoretical basis - the effect of noise within the framework of data driven posets.

The paper is organized as follows:
First we briefly give some background information, concerning “data-driven partial orders”, and then we explain how noise is introduced. The next sections introduce a theoretical concept for predicting the effect of noise on simple two-object- systems characterized by m indicators. An example taken from environmental chemistry will help to enlighten the theoretical concepts. A critical discussion concludes the paper.

2. Methods

2.1 Example data set

The dataset (X) has been taken from a study of Sailaukhanuly et al. [6], which includes a series of obsolete pesticides. The pesticides therein are to be ranked according to the hazard they exert on the environment. As there is no single quantity that describes the environmental hazard of a chemical a multi-indicator system (MIS) was used as a proxy. Accordingly the substances are characterized by persistence, bioaccumulation and toxicity in the environment. For the present study a subset, including 7 pesticides is selected from the complete data set with normalized data [7]. The object set, X encompasses 7 objects, i.e. $|X| = 7$ and the values are all in the interval [0, 1]. (Table 1). Within this series of indicators constituting the MIS, the three indicators are oriented such that a high value of an indicator expresses a high hazard on the environment. For further details, see [4,6,8].

Table 1. Indicator values for the 7 pesticides applied in the present study, the data matrix dm (for a description of the original dataset see [6]

Name	code	Pers	BioA	Tox
p,p-DDD	DDD	0	0.679	0.171
Methoxychlor	MEC	0.027	0.339	0.101
Aldrin	ALD	0.264	0.852	0.627
Dieldrin	DIE	0.293	0.383	0.041
Heptachlor	HCL	0.428	0.48	0.104
Chlordane	CHL	1	0.751	0.212
Hexachlorbenzene	HCB	0.057	0.574	0.187

2.2 The basic equation of Hasse diagram technique

In partial ordering the only mathematical relation among the objects is "≤" [9,10]. The "≤"-relation is the basis for a comparison of objects which are here characterized by indicators q_1, q_2,…,q_m. A given object x is comparable to object y if and only if the relation $x_j \leq y_j$ holds for all $j = 1,\ldots,m$, i.e. for all q_j of the MIS.

$$x_j \leq y_j \text{ for all } j = 1, 2, 3 \qquad (1)$$

Independent of the orientation $x \leq y$ or $x \geq y$ the very existence of a comparability is indicated by the symbol ⊥. Sometimes it is convenient to write $x <_j y$ to express that $x_j < y_j$. If there is no object z, for which $x < z < y$ is valid then the relation $x<y$ is called a cover relation. The cover relation is the basis to draw the Hasse diagrams visualizing the partial order. The construction of the cover matrix is based on the axiom of transitivity for partially ordered sets [11].

Eq. 1 is not necessarily fulfilled for all object pairs, x and y, i.e., a ≤-relation cannot be established for x and y for all indicators of the MIS. Consequently x and y are regarded as mutually incomparable (notation: $x \parallel y$). Hence, even very minor variations in the indicators values may be crucial for comparabilities/incomparabilities of the single object pairs being studied.

The analysis of eq. 1 results in an acyclic, transitively reduced, triangle free, directed graph of order relations, the so-called Hasse diagram. Hasse diagrams are visualizations of the order

relations due to eq. 1 and are a rather convenient tool when the set of objects is not too large (see for instance [9].

2.3 Modeling noise

Noise can be modeled in different way. We follow a 7 step procedure that is described in details by Carlsen and Bruggemann [4].

1. *uniform distribution* with percentage of variation p_j Randomly values are selected for the single objects out of the interval $[x_j - p/100 * x_j, x_j + p/100 * x_j]$. Such a sample subjected to noise is called xn_j (we are assuming the same noise level for all *m* indicators, to keep the analysis simple).
2. Other models of noise will be discussed in forthcoming papers.
3. Any run of the MC simulation leads to a perturbed data matrix, *dmp*. From each *dmp* a perturbed zeta matrix is obtained. The zeta matrix due to the k_{th} run describes the order relations as follows:
    a. $zeta_k(x,y) = 1$ if object ≤ object y otherwise 0 (2)
    b. When *mc* runs are performed, then *mc* zeta matrices are calculated. where *mc* denotes the total number of Monte Carlo simulations.
4. The mean of all *mc* zeta matrices is calculated:
5. $zetaav(x,y) = \Sigma zeta_k(x,y)/mc \; k = 1,\ldots,mc$ (3)
6. As *zetaav*(x,y) can take every value in the interval [0,1] it cannot a priori be expected that *zetaav* describes an order relation.

   The transitivity must be properly formulated, which is done according to the framework of fuzzy poset theory (DeBaets, DeMeyer [12]). By this method a matrix is obtained fulfilling the requirements of fuzzy transitivity [4]. This matrix is called *zetath*.
7. As in other fuzzy methods, a defuzzification applied on *zetath* is to be performed to obtain a crisp relational matrix, i.e., a matrix displaying only 0 and 1. After extracting possible equivalence relations the resulting matrix is called *zetacrisp.* The entries of *zetath* are usually ordered increasingly and are called α.

   The user selected cut-value will be called α*-cut*.

   Therefore
    a. the α *-cut* is to be selected,
    b. the equivalence relations are to be extracted and

c. the remaining matrix

$$zetacrisp(x,y)=\begin{cases}1 if\ zetath(x,y)\geq\alpha-cut\\ 0\quad otherwise\end{cases}$$

describes an order relation, which depends on the *α-cut* and thus on the actual noise.

2.4 The general probability scheme

In order to get insight into the role of noise on partial order, the most simple system, consisting of two objects and m (m > 1) indicators is selected. As the original data set can lead to x ⊥ y or to x ‖ y. Applying partial order on the original data matrix as it is, does not take into regard the noise. Therefore, to stress this situation we call the original data set the "frozen" data set.

In the present study the case x = y is excluded. Thus two different cases, i.e., x < y and x ‖ y will be considered (sect. 2.3.2, 2.3.3).

2.4.1 Basic concepts

To simplify notations we define:

$$n = p/100 \qquad (4)$$

and

xl_j = lower limit of x_j, xu_j upper limit of x_j.

Remind that we write xn_j for a noisy value of x_j according to the distribution range. An analogous notation is used for y.

Without restriction of generality we assume first x < y. In Fig. 1 a schematic overview is given, which does not refer to a specific distribution, but is valid for every distribution, where a certain overlap for the ranges of xn_j and yn_j is possible.

When $xu_j < yl_j$ (a situation not shown in Fig. 1) then obviously $xn_j < yn_j$ is obviously valid for any sample.

In case $yl_j < xu_j$ the range $[yl_j, xu_j]$ is not empty and is called an overlap "ovx_j, ovy_j". The length of the overlaps *ovx* and *ovy* is equal, thus

$$ovx_j = ovy_j \tag{5}$$

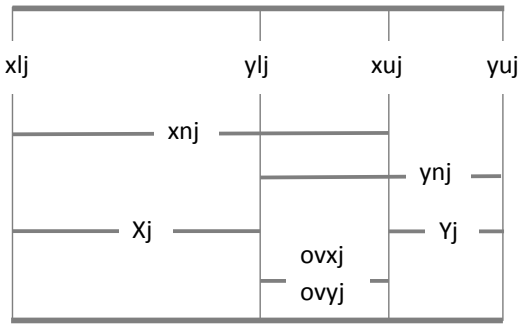

Figure 1. Due to the actual selected noise an overlapping of the zones accessible for xn_j and for yn_j appears: ovx_j, ovy_j; the zones X_j and Y_j are only accessible for xn_j and yn_j resp. (see 2.3.2 for further details)

The overlapping zone will also be abbreviated by ov, when now confusion is to be expected. We nevertheless differentiate between *ovx* and *ovy*, as the probability for xn_j to be in the interval ovx_j differs from the probability of yn_j to be in ovy_j. Thus, the distribution for y_j can be different from that for x_j as a reflection of different sizes of the ranges for xn_j and yn_j and thus for X_j and Y_j. The probability $Prob(xn < yn)$ takes into account all m indicators and is an "and" coupling of the indicator-wise probabilities of $xn_j < yn_j$, denoted similarly as $\text{Prob}(xn_j < yn_j)$.

Therefore:

$$Prob(xn < yn) = \prod_{j=1}^{m} Prob(xn_j < yn_j) \tag{6}$$

2.4.2 Case 1: x < y, an estimation of the probability of $xn_j < yn_j$

According to Fig. 1, four ranges for randomly taken values for xn_j and yn_j exist:

1) xn_j in X_j , yn_j in ovy_j
2) xn_j in X_j , yn_j in Y_j
3) xn_j in ovx_j, yn_j in Y_j and
4) xn_j in ovx_j and yn_j in ovy_j.

As xn_j can only be in X_j or ovx_j , and yn_j only in Y_j or ovy_j, we have

$Prob(xn_j \text{ in } X_j) + Prob(xn_j \text{ in } ovx_j) = 1$; simplified: $X_j + ovx_j = 1$ (7a)

$Prob(yn_j \text{ in } Y_j) + Prob(yn_j \text{ in } ovy_j) = 1$ simplified: $Y_j + ovy_j = 1$ (7b)

$Prob(xn_j \text{ in } ovx_j \text{ and } yn_j \text{ in } ovy_j) = Prob(xn_j \text{ in } ovx_j) * Prob(yn_j \text{ in } ovy_j)$ (7c)

Eq. 7c is most important. Correspondingly the term $Prob(nx_j \text{ in } ovx_j \text{ and } yn_j \text{ in } ovy_j)$ appears very often and an abbreviation may be appropriate

$pov_j = Prob(xn_j \text{ in } ovx_j \text{ and } yn_j \text{ in } ovy_j) = Prob(xn_j \text{ in } ovx_j) * Prob(yn_j \text{ in } ovy_j)$ (7d)

a) Once $xn_j \in ovx_j$ and $yn_j \in ovy_j$, it is natural to assume

$$Prob(xn_j < yn_j) = Prob(xn_j > yn_j) = 0.5 * pov_j \quad (8)$$

Beside the equality $xn_j = yn_j$, which is considered as negligible in the probability scheme, there are no other possibilities.

b) For the following it should be noted that the numerical value of $Prob(xn_j, yn_j \text{ in } ov)$ is independent, whether in the frozen data set $x_j < y_j$ or $y_j < x_j$ is valid. This remark should be kept in mind, when reading the following sections.

The probabilities for $xn_j < yn_j$ are a sum of four terms (according to the aforementioned ranges) which are added because the location of xn_j or yn_j follows an "or" - rule, i.e. xn_j is in X_j, yn_j in Y_j or xn_j is in X_j and yn_j is in ovy_j etc.

$Prob(xn_j \in X_j) * Prob(yn_j \in ovy_j) = X * ovy$

$Prob(xn_j \in X_j)*Prob(yn_j \in Y_j) = X*Y$

$Prob(xn_j \in ovx_j)*Prob(yn_j \in Y_j) = ovx*Y$

$pov_j = Prob(xn_j \in ovx_j)*Prob(yn_j \in ovy_j) = ovx*ovy$

The sum of these terms should be 1, as for two scalars, xn_j, yn_j are always comparable, i.e either

$xn_j > yn_j$ or $xn_j < yn_j$ (linear orders: "tertium non datur").

$X*ovy + X*Y + ovx*Y + ovx*ovy = X*(ovy + Y) + ovx*(ovy + Y) = X*1 + ovx*1 = 1$

Hence

$Prob(xn_j < yn_j | x_j < y_j)) = 1 - 0.5* pov_j$ (9)

For an application of eq. 8 it is to check

a) whether there is an overlap and
b) whether $x_j < y_j$ with respect to the frozen data set (which is expressed by the notation $Prob(xn_j < yn_j | x_j < y_j)$).

If the overlap area is empty, then $Prob(xn_j < yn_j) = 1$, because $Prob(xn_j \in ovx_j) = Prob(yn_j \in ovy_j) = 0$.

Summarizing:

$$Prob(xn_j < yn_j \mid x < y) = \begin{cases} 1 - 0.5 \cdot pov_j \\ 1 \qquad otherwise \end{cases} \quad (10)$$

2.4.3 Case 2: x ‖ y

Excluding (as above) equality $xn_j = yn_j$, there is an index set *J1*, for which is valid

$xn_j < yn_j \qquad j \in J1$ (11a)

and an index set *J2* for which is valid

$xn_j > yn_j \qquad j \in J2.$ (11b)

$|J1| + |J2| = m$, with $|J1| = m1$ and $|J2| = m2$

It is necessary to state the selection of *J2* in a clear manner:
If $Prob(xn<yn|\ x \parallel y)$ is to be determined, then *J2* encompasses those indicators, for which $x_j > y_j$.

However, if the probability for *xn* greater than *yn* is to be determined, i.e.
if $Prob(xn>yn|\ x \parallel y)$ is wanted, then *J2* encompasses just those indicators, for which $x_j < y_j$.

Let us determine *Prob* ($xn < yn$) and consider $x_{j1} < y_{j1}$, $j1 \in J1$ and $x_{j2} > y_{j2}$, $j2 \in J2$. The probability of $xn_{j1} < yn_{j1}$ is given as before by eq.(9).

$Prob(xn_{j1} < yn_{j1}) = 1 - 0.5*pov_{j1}$, $j1 \in J1$..
$Prob(xn_{j2} < yn_{j2}) = 1 - 0.5 *pov_{j2}$, $j2 \in J2$, hence the searched probability
$Prob(xn_{j2} > yn_{j2}) = 0.5*pov_{j2}$, $j2 \in J2$.

Together with eq. 9 we arrive at

$$Prob(\ xn < yn\ ) = \prod_{j1 \in J1}^{m1} Prob(\ xn_{j1} < yn_{j1}\ ) \cdot \prod_{j2 \in J2}^{m2} Prob(\ xn_{j2} > yn_{j2}\ ) \qquad (12)$$

and by the results of the lines above

$$Prob(\ xn < yn | x || y\ ) = \prod_{j \in J1} (1 - 0.5 \cdot pov_j\ ) \cdot \prod_{j \in J2} 0.5 \cdot pov_j \qquad (13)$$

Eq. 13 is most important to understand the role of noise for two objects x ‖ y within a poset.

2.5 Realization by a uniform distribution

2.5.1 Relation between probability and zetaav

The Monte Carlo simulations by means of the PyHasse module MCbasedHD8_2, described by Carlsen, Bruggemann [4] are performed independently, i.e., the probabilities for xn_j are not depending on probabilities for yn_j, as well as the probabilities calculated for indicator 1, do not influence the probabilities calculated for indicator 2. Hence, joint probabilities are just products of the corresponding probabilities, concerned for the position of x and y in the their

accessible ranges and taking into account that establishing for example $xn < yn$ needs that all probabilities for $j = 1, \ldots, m$ are to be regarded. Furthermore, any statistical correlation among indicators, which could be taken into account by an appropriate design of the Monte Carlo simulations is not considered. As zetaav is just an average of the outcome of the MC-calculation, a term *zetaav*(x,y) is the probability after all Monte Carlo-simulations that $xn < yn$. Therefore there is a direct correspondence as follows (eq. 14)

$$zetaav(xn < yn) = Prob(xn < yn) \tag{14}$$

Note, that we are stepping back to *zetaav* and consider *zetath* (where the needed transformations are included which are required by the fuzzy transitivity) later in sect. 2.6

2.5.2 Formula for pov_j

We introduce now estimations of the probabilities, discussed in the former two sections. Hereto we define

$$\delta_j := \max(x_j,y_j) - \min(x_j,y_j) + n*(x_j+y_j)) \text{ if } ovx_j=ovy_j=\text{nonempty} \tag{15}$$

and

$$\delta_j := 0 \text{ if } ovx_j = ovy_j \text{ is empty.} \tag{16}$$

$$Lx_j := xu_j - xl_j = (1+n)*x_j - (1-n)*x_j = x_j + n*x_j - x_j + n*x_j = 2*n*x_j \tag{17a}$$

$$Ly_j := yu_j - yl_j = (1+n)*y_j - (1-n)*y_j = y_j + n*y_j - y_j + n*y_j = 2*n*y_j \tag{17b}$$

Then the probability $Prob(xn_j, yn_j \text{ in } ov)$ which is according to eq.(13) the central quantity can be estimated as follows

$$ovx_j = \delta_j/Lx_j \tag{18a}$$

$$ovy_j = \delta_j/Ly_j \tag{18b}$$

and the final leading term is just

$$Prob(xn_j \in ovx_j \text{ and } yn_j \in ovy_j) = pov_j = (\delta_j)^2/(4*n^2*x_j*y_j) \tag{19}$$

In order to get a compact description, especially for programming, it seems useful to introduce the H-tuple: Without loss of generalization of the estimation of *Prob*(*xn* < *yn*), we define

$$H_j := \begin{cases} 1 & if\ xj < yj \\ 0 & else \end{cases}$$

Then it is obtained:

$$Prob(xn < yn) = \prod_{j=1}^{m} [H_j \cdot (1 - pov_j) + 0.5 \cdot pov_j] \qquad (20)$$

2.5.3 Scaling invariance of x_j , y_j

The model for noise assumes that noise-values are taken as relative values with respect to x_j and y_j, respectively. Thus, it is necessary to show that the model is not depending on a scaling of x_j and y_j, respectively. Indeed, all expressions containing x_j and y_j are homogeneous with respect to x_j and y_j and any scaling factor would cancel out. For example assuming $x_j < y_j$ and a scaling factor λ

$$\delta_j = x_j - y_j + n*(x_j + y_j) \rightarrow \delta_j(\lambda) = \lambda*x_j - \lambda*y_j + n*\lambda*(x_j + yj) = \lambda*\delta_j$$

Together with eq. 19 we obtain (writing for the sake of clarity not the abbreviating terms)

$$Prob(xn_j, yn_j \text{ in } ov(\lambda)) = (\delta_j(\lambda))^2/(4*n^2*(\lambda*x_j)*(\lambda*y_j)) = \lambda^2*\delta_j^2/(4*\lambda^2*n^2*x_j*y_j) =$$
$$\delta_j^2/(4*n^2*x_j*y_j) = Prob(xn_j, yn_j \text{ in } ov)$$

2.6 Role of crucial noise-values

2.6.1 Case 1: x < y

In section 2.4 the assumption was made that there must be an overlap so that noise can actually influence the poset. Hence, the question is, as to how far an overlap is possible. This can be easily calculated following Fig. 1 when a uniform distribution is assumed by examining the condition

$$xu_j = yl_j \text{ for } x_j < y_j \text{ in the frozen data set} \qquad (21a)$$

$yu_j = xl_j$ for $y_j < x_j$ in the frozen data set (21b)

The crucial noise, where the eq. 21a and eq. 21b is fulfilled, is calculated by eq. 22

$$nc_j = (\max(x_j,y_j) - \min(x_j,y_j)/(x_j + y_j) \quad (22)$$

If noise, n, is less $\min\{nc_1, nc_2,\ldots,nc_m\}$ (abbreviated as min nc_j, similarly we abbreviate $\max\{nc_1,nc_2,\ldots,nc_m\}$ by max nc_j) then the $ovxj$, ovy_j-contributions are 0 hence $Prob(xn_j \in ovx_j$ and $yn_j \in ovy_j) = pov_j = 0$ and $Prob(xn_j < yn_j) = 1$.

Summarizing:

$$Prob(xn < yn \,|\, x < y) = \begin{cases} 1 \; if \;\; n < min\, nc_j \;\; j \in \{1,\ldots,m\} \\ \leq 1 \; if \;\; n > min\, nc_j \;\; j \in \{1,\ldots,m\} \end{cases} \quad (23)$$

2.6.2 Case 2: x ‖ y

If x ‖ y, eq. 10 still applies. However, now the orientation on the single couples (x_j , y_j) is important.

Let us examine $Prob(xn < yn)$. Then Table 2 about possible cases is most useful. Note, we introduced *J1* and *J2* as index sets (cf. eq's. (11a, 11b)).

Table 2. $Prob(xn < yn|$ x ‖ y)) is to be determined. 0_m:= (0*0*..,0) m factors, 1_m = 1*1*…*1 *m* factors, ε_m: ε*ε*…*ε *m* factors, where ε is a value > 0. An expression $\varepsilon_1 * 0_{m2\text{-}1}$ means: Evaluating *J2* (eq. 11b) one factor in the product in eq. 13 is unequal zero, the remaining *m2*-1 factors are still 0. Note the convention used for min nc_j and max nc_j respectively.

	$n <$ min nc_j $j \in J1$	min $nc_j < n <$ max nc_j $j \in J1$	$n >$ max nc_j $j \in J1$
$n <$ min nc_j $j \in J2$	$1_{m1} * 0_{m2}$	$\varepsilon_{m1} * 0_{m2}$	$\varepsilon_{m1} * 0_{m2}$
min $nc_j < n <$ max nc_j $j \in J2$	$1_{m1} * \varepsilon_1 * 0_{m2\text{-}1}$	$\varepsilon_{m1} * \varepsilon_1 * 0_{m2\text{-}1}$	$\varepsilon_{m1} * \varepsilon_1 * 0_{m2\text{-}1}$
$n >$ max nc_j $j \in J2$	$1_{m1} * \varepsilon_{m2}$	$\varepsilon_{m1} * \varepsilon_{m2}$	$\varepsilon_{m1} * \varepsilon_{m2}$

The resulting values are shown in Table 3. Note: ε In different cells of Table 2 does not necessarily mean that the values are all equal.

Table 3. Evaluation of Table 2.

	$n < \min nc_j$ $j \in J1$	$\min nc_j < n < \max nc_j$ $j \in J1$	$n > \max nc_j$ $j \in J1$
$n < \min nc_j$ $j \in J2$	0	0	0
$\min nc_j < n < \max nc_j$ $j \in J2$	0	0	0
$n > \max nc_j$ $j \in J2$	ε	ε	ε

The main result is that under the conditions assumed for Table 2, the max nc_j $j \in J2$ is of crucial importance.

$Prob(xn < yn | x \parallel y) \neq 0$ only if $n > \max ncj$ of those j, where within the frozen data set $x_j > y_j$). I.e. for $j \in J2$. (24).

2.6.3 Summarizing

Initially it must be decided which probability is to be termined, i.e.,

$Prob(xn < yn)$ or $Prob(xn > yn)$ or $Prob(xn \parallel yn)$ (within HDT "tertium non datur" is not valid!). Here $Prob\ (xn < yn)$ is chosen.

Whether $x \perp y$ or $x \parallel y$ the partitioning of $\{1,\ldots,m\}$ into $J1$ and $J2$ becomes an important point and the second decision has to be made (to determine, if necessary $J1$ and $J2$)

a) $x < y$: Then $J1 \cup J2 = \{1,2,\ldots,m\}$ is relevant as $\min nc_j$, $j \in \{1,\ldots,m\}$ determines when the probability $Prob(xn < yn)$ becomes less 1
b) $x \parallel y$: Now set $J2$ is relevant, and $\max nc_j$, $j \in J2$ determines when $Prob(xn < yn)$ becomes larger 0.

Therefore it is of main interest within an evaluation of objects to determine the noise level where the noise begins influencing the poset. Eq.s (23 and 24) provide the necessary information. The quantities $\min nc_j$, $j \in J1$, $\max nc_j$ $j \in J1$, and similarly $\min nc_j$, $j \in J2$, max

nc_j $j \in$ *J2* can take independent from each other different values for *J1* and for *J2*. When *Prob*(*xn* < *yn*) is of interest and the frozen data set implies x ‖ y, ,the transition from *Prob*(*xn* < *yn*) = 0 to *Prob*(*xn* < *yn*) ≠ 0 depends solely on the actual numerical value of max nc_j $j \in$ *J2*.

2.7 Extensions to more than two objects

The derivation of the *zetaav*-matrix is based on the pairwise comparison of any single couple (x,y), x,y ∈ *X*. Hence, the entries of this matrix "do not know" anything of a transitivity. The transitivity axiom is nevertheless fulfilled, which arises from the fact that the single indicators themselves guarantee the transitivity. The transitivity comes only into play, when finally a Hasse diagram is to be constructed on the basis of the cover relations. Therefore in discussing zetaav we are freed from the additional constraint of the (fuzzy) transitivity. This additional aspect is the task of the approximation, leading to *zetath* and finally to *zetacrisp*, which is the basis to get cover relations and from them the Hasse diagram (see [4]).

The sections above have shown that the matrix *zetaav* is influenced by noise in two processes

a) Existing x<y-relations with *zetaav*(x,y) = 1 (frozen data set) may transform to *zetaav*-values <1 and
b) x ‖ y relations with *zetaav*(x,y) = 0 and *zetaav*(y,x) = 0 (frozen data set) may transform to *zetaav* values sligjhtly larger than 0.
c) According to the *nc*-values (their minimal value with respect to *J1* and maximal values with respect to *J2*) the series of processes a) and of processes b) start at different noise levels.

Hence retaining the Hasse diagram (of the frozen data set) if noise is active, is possible, when the

α-cut > max(*Prob*(*xn* < *yn*| x||y)) and *α-cut* < min(*Prob*(*x'n*<*y'n* | x'<y') with

max(*Prob*(*xn* < *yn*| x||y)) < min(*Prob*(*x'n*<*y'n* | x'<y').

We summarize this finding as follows:

Let noise have a value > max(min nc_j $j \in$*J1*, max nc_j $j \in$*J2*) and
let be x,y, x',y' be objects where at least either x' ≠ x or y' ≠ y,
let furthermore be max(*Prob*(*xn* < *yn*| x||y)) < min(*Prob*(*x'n*<*y'n* | x'<y')
then the poset obtained from the frozen set equals the poset obtained from the noisy poset, if
α-cut ∈ (max(*Prob*(*xn* < *yn*| x||y)) , min(*Prob*(*x'n*<*y'n* | x'<y'))

3. Examples

3.1 Toy examples

3.1.1 Case 1, frozen data set: x < y

To demonstrate the theoretical part it is useful to look first to constructed examples, by which the crucial effects in noise can be better seen. Therefore we introduce a data matrix describing x < y (Table 4).

Table 4. Toy example for case 1: x < y

	q1	*q2*	*q3*	*q4*
x	0.1	0.2	0.3	0.4
y	0.9	0.7	0.6	0.5
nc	0.8	0.555	0.333	0.111

All four indicators are considered and the last row of Table 4 shows the values of *nc*. For the case x < y only min nc_j = 0.1111 is important according to eq. 23. Hence, long as noise is below this level, *xn* < *yn*, and consequently *Prob*(*xn* < *yn*) = 1. Only if noise exceeds 0.1111 the probability. Hence, *Prob*(xn_4 < yn_4) will be reduced and *zetaav*(x, y) would result in values less 1. (Fig. 2):

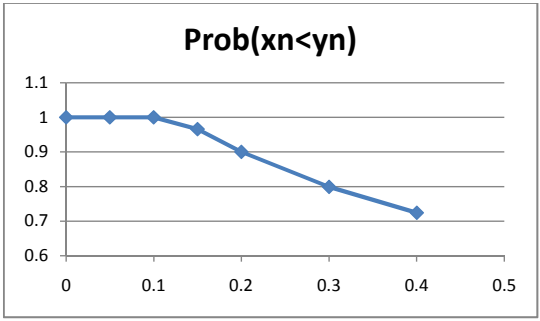

Figure 2. *Prob*(*xn* < *yn*) as function of noise. If *n* exceeds 0.1111 then the overlap zone $ovx_4 = ovy_4$ *is* no more empty and the reduction of the probability *xn* < *yn* starts.

If the MIS shown in Table 4, is now considered, however, only including the first three indicators are considered. It this case min nc_j = 0.3333. The resulting probability distribution for *xn* < *yn* is displayed in Fig. 3.

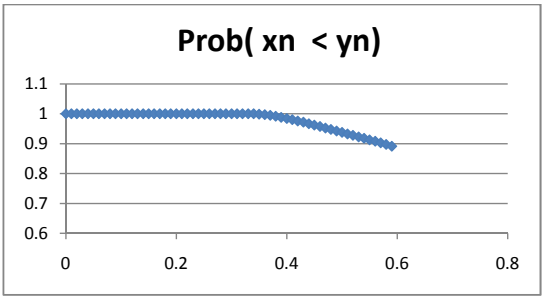

Figure 3. *Prob*(*xn*<*yn*) as function of noise. If *n* reaches 0.333 then the overlap zone is not empty. A reduction of the probability for *xn* < *yn* for *n* > 0.333 appears.

Fig.s 2 and 3 demonstrate the important role of the crucial noise *nc*, to detect the noise level, where the partial order becomes affected by noise. However to calculate *Prob*(*xn* < *yn*) = f(*n*) the equations (13) and (23) resp., are to be evaluated. The knowledge of f(*n*) is important, when noise exceeds min nc_j , $j \in J1$.

3.1.2 Case 2, frozen data set: x ‖ y

In Table 5 the same values are used as in Table 4, however their assignment to x and y is changed.

Table 5. Toy example for case 2: x ‖ y

	q1	*q2*	*q3*	*q4*
x	0.9	0.7	0.3	0.4
y	0.1	0.2	0.6	0.5
nc	0.8	0.555	0.333	0.111
orientation	x >1 y	x >2 y	x <3 y	x <4 y

As can be seen, the *nc* values are invariant with respect to exchange of x_j and y_j, following eq. 22.

The evaluation of data of Table5 is here based on the assumption that

a) *Prob*(*xn* < *yn*) is of interest, and
b) determination of *J2*. Here *J2* = {1, 2} (note: *J2* is an index set).
c) to identify the largest nc_j value. Here the largest *nc*-value for $j \in J2$ is that for *q1*, therefore the probability of *xn* < *yn*, is 0 as long as noise (*n*) is less 0.8 (Fig. 4)

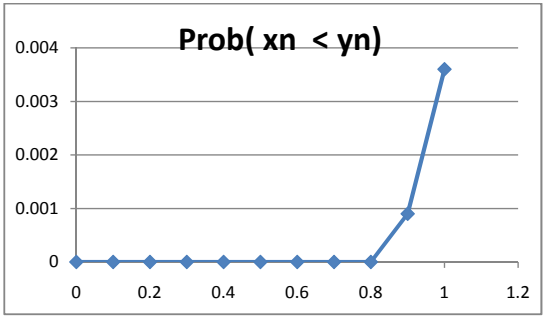

Figure 4. *Prob*(*xn* < *yn*) as a function of noise. The actual max nc_j value equals 0.8. The values of the probabilities *Prob*(*xn* < *yn*), *Prob*(*xn* > *yn*) and *Prob*(*xn*‖*yn*) add to 1, the major contribution arises from *Prob*(*xn* ‖ *yn*)

Analoguously to the above we turn to a MIS including only *q2*, *q3* and *q4* leads to the probability plot as shown in Table 5, the resulting probability distribution for *xn* < *yn* is shown in Fig. 5

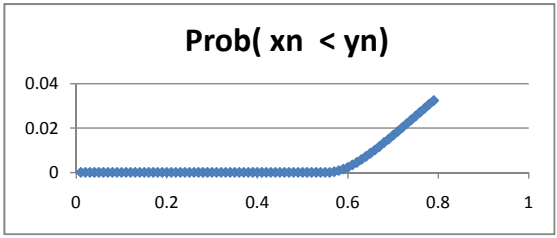

Figure 5. *Prob*(*xn* < *yn*) as a function of noise. Now values > 0 only after max nc_j = 0.5555, which is now the relevant crucial noise.

The set of the last four Fig's demonstrate that there are crucial noise values which determine

- whether or not an existing order relation x < y is affected and hence can be set to zero by a suitable defuzzification interval, namely min nc_j with $j \in \{q1,q2,q3,q4\}$
- whether or not a new order relation can be obtained from an object pair for which the original data set implies x ‖ y, namely max nc_j, with $j \in J2$
- thus, if noise is less min nc_j then the original Hasse diagram (i.e. the Hasse diagram based on a data matrix considered as non-noisy) will be retained.

  thus, if noise is less max nc_j) , $j \in J2$, then for no couple (x,y) with (x ‖ y) due to the original data set the incomparability can be replaced by a comparability (with a low probability ≠ 0) due to noise effects.

In Fig. 6 both *Prob*(*xn* < *yn*) and *Prob*(*xn* > *yn*) for the situation as described above in Table 5, i.e., the MIS from Table 4 excluding *q4*, are shown:

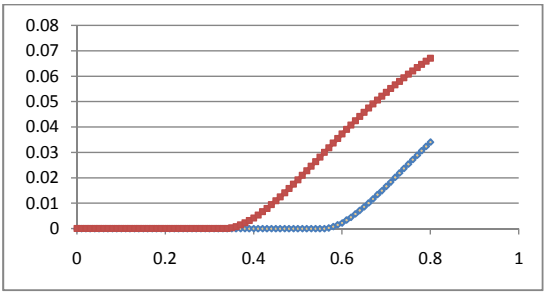

Figure 6. Upper curve: *Prob*(*yn* < *xn*), lower curve: *Prob*(*xn* < *yn*). Beside the slightly differing slopes, the effect of *nc* and of *J2* is demonstrated. Note (once again) that *Prob*(*xn*< *yn*) + *Prob*(*xn* > *yn*) + *Prob*(*xn* ǁ *yn*) = 1

Taking *Prob*(y < x) as example: Now *J2* is that set, where $y_j > x_j$: This is only for *q3* and *q4*, where the maximal value is $nc_3 = 0.333$.

3.2 Application of the concepts onto the real data matrix (sec. 2.1)

When the data are taken as given in Table 1 then an important question is, at which level of noise the Hasse diagram based on the data of Table1 (frozen data set) will be retained. In order to find an answer for this question it is only necessary to check the nc-values for each pair of chemicals.

In Fig. 7 the Hasse diagram of the frozen data set (Table 1) is shown. It is seen that there are three minimal elements: DDD, MEC and DIE, whereas there are two maximal elements, namely ALD and CHL. To check for stability of the partial order, visualized by the Hasse diagram, shown in Fig. 8 the list of nc-values is to be examined, due to eq. 22 (Table 6).

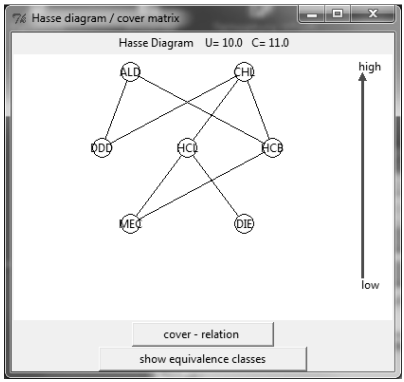

Figure 7. Hasse diagram based on data of Table 1 (PyHasse software)

Table 6. List of nc-values, available by MCbasedHD8_2.py (nc-values, chemicals, the property (indicator) and the last two columns the type of relation (0 0_ x l y, , 0 1: x < y, 1 0: x > y, x being the first, y the second object)

nc	object1	object2	prop	zeta-values	
1.0	DDD	MEC	Pers	0	0
0.334	DDD	MEC	BioA	0	0
0.257	DDD	MEC	Tox	0	0
1.0	DDD	ALD	Pers	1	0
0.113	DDD	ALD	BioA	1	0
0.571	DDD	ALD	Tox	1	0
1.0	DDD	DIE	Pers	0	0
0.279	DDD	DIE	BioA	0	0
0.613	DDD	DIE	Tox	0	0
1.0	DDD	HCL	Pers	0	0
0.172	DDD	HCL	BioA	0	0
0.244	DDD	HCL	Tox	0	0
1.0	DDD	CHL	Pers	1	0
0.05	DDD	CHL	BioA	1	0
0.107	DDD	CHL	Tox	1	0
1.0	DDD	HCB	Pers	0	0
0.084	DDD	HCB	BioA	0	0
0.045	DDD	HCB	Tox	0	0
0.814	MEC	ALD	Pers	1	0
0.431	MEC	ALD	BioA	1	0
0.723	MEC	ALD	Tox	1	0
0.831	MEC	DIE	Pers	0	0
0.061	MEC	DIE	BioA	0	0
0.423	MEC	DIE	Tox	0	0
0.881	MEC	HCL	Pers	1	0
0.172	MEC	HCL	BioA	1	0
0.015	MEC	HCL	Tox	1	0
0.947	MEC	CHL	Pers	1	0
0.378	MEC	CHL	BioA	1	0
0.355	MEC	CHL	Tox	1	0
0.357	MEC	HCB	Pers	1	0
0.257	MEC	HCB	BioA	1	0
0.299	MEC	HCB	Tox	1	0
0.052	ALD	DIE	Pers	0	0
0.38	ALD	DIE	BioA	0	0
0.877	ALD	DIE	Tox	0	0
0.237	ALD	HCL	Pers	0	0
0.279	ALD	HCL	BioA	0	0
0.715	ALD	HCL	Tox	0	0
0.582	ALD	CHL	Pers	0	0
0.063	ALD	CHL	BioA	0	0
0.495	ALD	CHL	Tox	0	0
0.645	ALD	HCB	Pers	0	1
0.195	ALD	HCB	BioA	0	1

0.541	ALD	HCB	Tox	0	1
0.187	DIE	HCL	Pers	1	0
0.112	DIE	HCL	BioA	1	0
0.434	DIE	HCL	Tox	1	0
0.547	DIE	CHL	Pers	1	0
0.325	DIE	CHL	BioA	1	0
0.676	DIE	CHL	Tox	1	0
0.674	DIE	HCB	Pers	0	0
0.2	DIE	HCB	BioA	0	0
0.64	DIE	HCB	Tox	0	0
0.401	HCL	CHL	Pers	1	0
0.22	HCL	CHL	BioA	1	0
0.342	HCL	CHL	Tox	1	0
0.765	HCL	HCB	Pers	0	0
0.089	HCL	HCB	BioA	0	0
0.285	HCL	HCB	Tox	0	0
0.892	CHL	HCB	Pers	0	1
0.134	CHL	HCB	BioA	0	1
0.063	CHL	HCB	Tox	0	1

As can be seen, the least value of nc equals 0.015 for MEC $<_{Tox}$ HCL. While n < 0.015 (1.5%) the Hasse diagram shown in Fig. 7 describes correctly the order relations. A new comparability, albeit with a low probability could appear for the chemicals DDD and HCB, when noise exceeds the corresponding *nc*-value of 0.045.

In Table 7 the results of an application of the MCbasedHD8_2 module is shown.

Table 7. *zetaav*-values $n < 0.015$

	DDD	MEC	ALD	DIE	HCL	CHL	HCB
DDD	1.0	0.0	1.0	0.0	0.0	1.0	0.0
MEC	0.0	1.0	1.0	0.0	1.0	1.0	1.0
ALD	0.0	0.0	1.0	0.0	0.0	0.0	0.0
DIE	0.0	0.0	0.0	1.0	1.0	1.0	0.0
HCL	0.0	0.0	0.0	0.0	1.0	1.0	0.0
CHL	0.0	0.0	0.0	0.0	0.0	1.0	0.0
HCB	0.0	0.0	1.0	0.0	0.0	1.0	1.0

The *zetaav*-value (MEC<HCL) = 1, indicating that there is no change in the order relations. It should further be noted that no entry of the *zetaav*-matrix deviates from 0 or 1.

Increasing the noise, *n* to 0.02 (2 %) a change in the *zetaav*-values are disclose (Table 8).

Table 8. *zetaav* (n = 0.02)

	DDD	MEC	ALD	DIE	HCL	CHL	HCB
DDD	1.0	0.0	1.0	0.0	0.0	1.0	0.0
MEC	0.0	1.0	1.0	0.0	0.965	1.0	1.0
ALD	0.0	0.0	1.0	0.0	0.0	0.0	0.0
DIE	0.0	0.0	0.0	1.0	1.0	1.0	0.0
HCL	0.0	0.0	0.0	0.0	1.0	1.0	0.0
CHL	0.0	0.0	0.0	0.0	0.0	1.0	0.0
HCB	0.0	0.0	1.0	0.0	0.0	1.0	1.0

The only change is for the entry *zetaav*(MEC < HCL) (see second row of Table 8) and as one can expect following the section 2, the value of *zetaav*(MEC<HCL) is almost 1. An *α-cut* must be selected rather high > 0.965 in order to obtain by defuzzification a MEC‖HCL-relation.

Now applying a noise of 10% Table 6 shows all nc_j -Values. It is immediately noted that many couples of chemicals whose nc_j-value < 0.1, are present (Table 9).

Table 9. Chemical couples where nc_j are < 0.1

Chemical 1	Chemical 2	indicator	relation	nc_j
DDD	CHL	BioA	<	0.05
DDD	HCB	BioA	‖	0.084
DDD	HCB	Tox	‖	0.045
MEC	DIE	BioA	‖	0.061
MEC	HCL	Tox	<	0.015
ALD	DIE	Pers	‖	0.052
ALD	CHL	BioA	‖	0.063
HCL	HCB	BioA	‖	0.089
HCB	CHL	Tox	<	0.063

As can be seen with noise = 10% some x < y relations become reduced and some x ‖ y relations get values ≠ 0 , as can be verified by inspecting the *zetaav*-values. Correspondingly the *α*-values allow a variety of possible *α-cuts* (Table 10).

Table 10. α-values at noise = 10% for all three indicators

0.0 0.013 0.113 0.637 0.876 0.927 1.0

When the couples of chemicals within Table 9 are incomparable, then their values for *Prob*(*xn*<*yn*) is small and may represented due to rounding effects by 0. The *zetaav*-values are

mostly affected by couples of chemicals, which are comparable, i.e. (DDD,CHL), (MEC,HCL) ad (HCB, CHL). In Table 11 the corresponding entries are underlined. :

Table 11. *zetaav*, (noise = 10%)

	DDD	MEC	ALD	DIE	HCL	CHL	HCB
DDD	1.0	0.0	1.0	0.0	0.0	0.8756	0.0126
MEC	0.0	1.0	1.0	0.0	0.6366	1.0	1.0
ALD	0.0	0.0	1.0	0.0	0.0	0.0	0.0
DIE	0.0	0.0	0.1133	1.0	1.0	1.0	0.0
HCL	0.0	0.0	0.0	0.0	1.0	1.0	0.0
CHL	0.0	0.0	0.0	0.0	0.0	1.0	0.0
HCB	0.0	0.0	1.0	0.0	0.0	0.9267	1.0

For example *zetaav*(DDD<HCB) has now a value unequal zero but very small. Indeed if we want to check *Prob*(*DDD*n < *HCBn*) then those nc_j are relevant, where *j* is taken from *J2*: (eq. 22). *J2* ={ 2 } within the MIS, i.e. indicator BioA, hence this value is decisive for getting *Prob*(*DDDn* < *HCBn*).

Table 12. The special case DDD || HCB , determination of *Prob*(*DDDn* < *HCBn*)

DDD	Pers	BioA	Tox
DDD	0	0.679	0.171
HCB	0.057	0.574	0.187
nc	1	0.084	0.045
J1	1		3
J2		1	

Selecting now an α-*cut* in the interval (0.113, 0.637) for example α-*cut* = 0.5 will still retain DDD||HCB, DIE||ALD. However no new comparisons appear, because all the slightly reduced values of *zetaav* will not affected by the defuzzification. DDD<CHL, MEC<HCL, HCB<CHL Even *if* α-*cut* = 0.9 would be selected then DDD||CHL, MEC||HCL, but HCB < CHL (Fig.9). The only affected relations are MEC<HCL and DDD<CHL whose *zetath*-values are less 0.9. (Fig. 9)

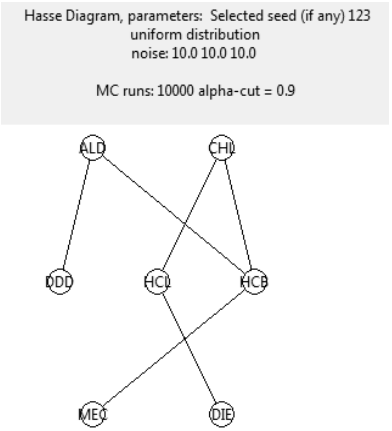

Figure 9. Hasse diagram based on data of Table 1, affected by noise (n = 0.1)

The finding up to now shows that those nc-values become important where potentially x ⊥ y is affected. By sufficient (i.e. > min nc_j) large noise, *zetaav*-values will be reduced and hence may be crucial for the defuzzification, Therefore it is useful to calculate a) a Table with all minimal nc-values describing existing <-relations and b) with those *nc*-values which are less a certain limiting value, which is here arbitrarily selected to be 0.1 (Table 13).

The module MCbasedHD8_2 of the PyHasse software (Carlsen, Bruggemann, 2015) fulfills this task.

Table 13. A: Minimal *nc*-values of all chemicals for which an <-relation can be established due to data in Table 1. B: Only those chemicals and min nc-values, which are less a limiting value (here arbitrarily selected to be 0.1)

A:

Object1	Object2	min value of *nc*
DDD	ALD	0.113
DDD	CHL	0.05
MEC	ALD	0.431
MEC	HCL	0.015
MEC	CHL	0.355
MEC	HCB	0.257
DIE	HCL	0.112

B:

Object1	Object2	min value of *nc*
DDD	CHL	0.05
MEC	HCL	0.015

There will be a probability Prob(*DDDn* < *CHLn*) which is slightly smaller than 1, at noise of less 0.1. Hence forming DDDn || CHLn has a pretty low probability. Similarly for the other pair of chemicals.

When a data driven poset is affected by noise up to a level of 10% then (in the framework of our noise model) only two couples of chemicals need a careful inspection. According to the general line of argumentation, for these two pairs a “reaction” can be written as follows

$$DDD < CHL \xrightarrow{\text{low probability}} DDD||CHL$$
$$MEC < HCL \xrightarrow{\text{low probability}} MEC||HCL$$

4. Discussion

4.1 zetaav versus zetath

The probability calculus was applied to estimate the *zetaav*-values, whereas the leading quantity to draw the Hasse diagram is *zetacrisp* which is derived from *zetath*. Nevertheless *zetaav* can be play its crucial role in the discussion, because *zetath* is *zetaav*, modified due due to fuzzy-transitivity, but in general *zetath*-values do not much differ from the *zetaav*-values. Furthermore we concentrate in the discussion of *zetaav* on a single object pair, in order not to check for the consequences the transitivity axiom would imply.

4.2 What is the win

Based on eq.’s 10, 13 the concept of crucial noise was established and can be calculated according to eq. 22. The main win is therefore the possibility of an a priori estimation as to how far a partially ordered set as that of the matrix, shown in sec. 2.1 can be obtained which is not affected by noise.

The decisive information is won by examination of the minimal value of the *nc*-tuple for each object pair and each indicator. If noise is exceeding this minimal value, then either existing order relations become available by an appropriate selection of the *α-cut*, or new comparisons can be possible. This latter effect is however most often irrelevant, because in general the zetaav-values of new comparisons are very low. As generating x ⊥ y from x || y by noise needs that noise must exceed those nc_j values, where *j* is taken from *J2*.

However, when the knowledge according to noise shows that noise exceeds the *nc*-values then the probabilities according to eq.'s 10, 13 can be calculated and graphs like those in Fig's. 2-5 may be helpful to decide which *α-cut* value could be appropriate.

4.3 What next

There are several assumptions, which must be step by step resolved and clarified.

- The nature of noise must be critically examined and other noise models must be checked.
- Note, a problem is the normalization to [0,1], either the sampling must be controlled, or a renormalization has to be done after each random selection. To keep the analysis simple, however, we do not take into regard the problems at the limits of the interval.
- The uniform distribution may be too simple for most applications; here a more general procedure must be derived, which is independent on the distribution model. It is clear that the relevant part of two distributions (one for the first and one for the second object) is the zone of their overlapping. So, if for example a normal distribution is considered and the upper tail of the one object and the lower tail of the other object have an area which is significantly greater zero then there is an overlap, and equations where ovx and ovy appear should work.
- The way how Monte Carlo simulations are performed, as exposed in [4], must be critically examined: Correlations among indicators must be adequately considered in the design of Monte Carlo simulations, as well as it would be a good idea to estimate the lower and upper limit s of the number of Monte-Carlo runs to obtain reliable data.
- Furthermore, there are some papers which enlighten the problem of noise from a combinatorial approach [1]. Can these methods be integrated to derive additional criteria?
- Although equality $xn_j = yn_j$ under noise seems to be not a relevant fact, it should be possible to include this special case into the general scheme, developed in this paper.

4.4 Other concepts

Noisy posets are of interest over many years. Some programs in the framework of PyHasse [5] allow an analysis of some aspects of noisy data sets. For example the module prob_min_max performs a Monte-Carlo simulation based on intervals the user can define, It is studied as to how far extremal elements remain extremal and as to how far non extremal elements can be considered as extremal ones. A balance is done with following aspects:

Probability of being a maximal, a minimal, an isolated element or none of them. This program is based on Monte Carlo simulations as MCbasedHD8_2 and needs a specification by the user which entries of the data matrix are considered as uncertain. An application concerning removal of Uranium will be published in the near future [13].

References

[1] P. B. Sørensen, B. B. Mogensen, S. Gyldenkaerne, A. G. Rasmussen, Pesticide leaching assessment method for ranking both single substances and scenarios of multiple substance use, *Chemosphere* **36** (1998) 2251–2276.

[2] P. P. Sørensen, B. B. Mogensen, L. Carlsen, M. Thomsen, The influence on partial order ranking from input parameter uncertainty – Definition of a robustness parameter, *Chemosphere* **41** (2000) 595–600.

[3] R. Wieland, R. Bruggemann, Hasse diagram technique and Monte Carlo simulations, *MATCH Commun. Math. Comput. Chem*. **70** (2013) 45-59

[4] L. Carlsen, R. Bruggemann, On the influence of data noise and uncertainty on ordering of objects, described by a multi-indicator system. A set of pesticides as an exemplary case, *J. Chemom*., in press.

[5] R. Brüggemann, L. Carlsen, K. Voigt, R. Wieland, PyHasse software for partial order analysis: Scientific background and description of selected modules, in: R. Brüggemann, L. Carlsen, J. Wittmann (Eds.), *Multi-Indicator Systems and Modelling in Partial Order*, Springer, New York, 2014, pp 389–423.

[6] Y. Sailaukhanuly, A. Zhakupbekova, F. Amutova, L. Carlsen, On the ranking of chemicals based on their PBT characteristics: Comparison of different ranking methodologies using selected POPs as an illustrative example, *Chemosphere* **90** (2013) 112–117.

[7] R. Bruggemann, L. Carlsen, Incomparable – What now? *MATCH Commun. Math. Comput. Chem.* **71** (2014) 699–714.

[8] R. Bruggemann, L. Carlsen, Incomparable – What now III. Incomparabilities, elucidated by a simple version of ELECTRE III and a fuzzy partial order approach, *MATCH Commun. Math. Comput. Chem*. **73** (2015) 277–302.

[9] R. Brüggemann, L. Carlsen, Introduction to partial order theory exemplified by the evaluation of sampling sites, in: R. Brüggemann, L. Carlsen (Eds.), *Partial Order in Environmental Sciences and Chemistry*, Springer, Berlin, 2006, pp 61–110.

[10] R. Brüggemann, G. P. Patil, *Ranking and Prioritization for Multi-Indicator Systems – Introduction to Partial Order Applications*, Springer, New York, 2011.

[11] B. A. Davey, H. A. Priestley, *Introduction to Lattices and Order*, Cambridge Univ. Press, Cambridge, 1990.

[12] B. De Baets, H. De Meyer, On the existence and construction of T-transitive closures, *Inf. Sci.* **152** (2003)167–179.

[13] N. Y. Quintero, G. Restrepo, R. Bruggemann, *in preparation*.