MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **80** (2018) 769-791

ISSN 0340 - 6253

Partial Order and Inclusion of Stakeholder's Knowledge

Rainer Bruggemann[a], Lars Carlsen[b]

[a]Leibniz-Institute of Freshwater Ecology and Inland Fisheries, Department Ecohydology;

brg_home@web.de

[b] Awareness Center, Linkøpingvej 35, Trekroner, DK-4000 Roskilde, Denmark

LC@AwarenessCenter.dk

(Received March 14, 2017)

Abstract

Evaluation, for example of chemicals, results mainly in a data matrix, where appropriate indicators are forming the columns and the chemicals the rows. Such a data matrix can be represented by a partial order and from a statistical point of view many insights are possible. However, if the data matrix should support decisions, then there appears the requirement to transform a poset until a linear order is obtained. This transformation often is performed just by calculating weighted sums of the (normalized) indicators, with weights, representing some knowledge beyond the data matrix. The question thus arises, i.e., as to how far weights can be determined as well- defined numbers? In the present study the condition of finding weights as unique numbers is declined. An alternative method is suggested and applied to a set of chemicals, where three indicators describe the environmental hazard of 13 substances. Consequences, concerning the environmental hazard of these chemicals are discussed. A result is that a systematic evolution of uncertainty about weights can be defined and that there appears to be a useful equation, which relates uncertainty (in weight determination) with the resulting ambiguity, measured as a posetic quantity, namely as number of resulting incomparabilities.

1 Introduction

Partial order theory is one of the general structures of mathematics and appears in many applications, without being explicitly denoted (see other contributions to this special issue of MATCH). Partial order can also be applied within an evaluative, statistical context, i.e., to evaluate and explore objects, which are characterized by a multi-indicator system (MIS).

Hereto a deep theoretical analysis can be found in [1] and more practically oriented analyses in [2] and in many other papers, previously published in MATCH (for example [3,4]). The main idea in application of partial order for evaluation is its potential to avoid subjective knowledge. Orders relations can be derived, without any knowledge beyond the data matrix, which realizes the MIS for a certain set of objects, *X*. However, often there is knowledge or even assumptions, worth to include into the evaluation analysis, albeit often pretty vague. Such information could, e.g., be assumptions about the mutual importance of the single indicators. In that context Owsinski [5] writes "…we would like to go beyond the poset 'skeleton' and endow it with 'flesh'…". However, to agree, or at least reach consensus, e.g., about specific weights will often be a troublesome exercise, and maybe even impossible, whereas to agree about some intervals for the mutual weights may be a significantly easier task. An approach, how for example blurred knowledge can be included within the framework of partial order is described in [6,7] and will be deepened in this paper. The paper is organized as follows:

- Section 2 presents the data basis, from which we take the example, the basics of partial order and the procedure for a systematic investigation of weight intervals.
- Section 3 shows some results concerning some environmental properties of the 12 chemicals and
- Section 4 discusses the results in order to define future tasks.

2 Materials and Methods

2.1 Environmental Chemistry

The example is based on environmental chemistry data, recently reported by Sailaukhanuly et. al., [8] (Table 1). The main objective is estimating the mutual hazard these chemicals pose to the environment. In general there is no measured quantity that uniquely order chemicals according to their environmental hazard. Instead a series of indicators are used as proxy [8].

Table 1. [0,1]-normalized indicators ‘persistence’, P, ‘bioaccumulation’, B, and ‘toxicity’, T.

chemicals	**P**	**B**	**T**
DDT	0.084	1	1
DDE	0.009	0.856	0.16
DDD	0	0.679	0.171
MEC	0.027	0.339	0.101
ALD	0.264	0.852	0.627
DIE	0.293	0.383	0.041
HCL	0.428	0.48	0.104
CHL	1	0.751	0.212
LIN	0.027	0	0
HCB	0.057	0.574	0.187
PCN	0.054	0.18	0.028
PCP	0.012	0.354	0.01

2.2 Basics of partial order applied in the statistical context

Let x, y be objects $\in X$, the set X for example being the set of the 12 chemicals given in Table 1. Let further the set of any quantity, $q(j)$ which will be used to find an order among the objects (here: among the chemicals) be called the information base. In the present example the information base would be {P, B, T}. When a set of consensus functions, $\Gamma(1), \Gamma(2),\ldots$based on the $q(j)$ are used to construct a partial order, then the information base would consist of just these consensus functions. The consensus functions $\Gamma(j)$ are considered as order preserving functions of $q(j)$. In that context Fattore [9] uses the concept of a functional. The consensus function terminology is outlined in sect. 2.3.

We define a partial order as follows:

$$x \le y :\Leftrightarrow q(j,x) \le q(j,y) \text{ for all } q(j) \in IB \qquad (1)$$

It is useful, to write $x \le_{IB} y$, to stress the role of the set IB. A set X, equipped with a partial order is a partially ordered set, a poset.

As mentioned above, in eqn. 1 *q*(*j*) can be either the original indicators as taken from an empirical data matrix or elements of the set of consensus functions. Eqn. 1 leads to an often very useful visualization by a Hasse diagram (see for details [2]). An object pair x, y which does not follow eqn.1, meaning that x and y are incomparable, is denoted as x ‖ y, the phenomenon itself being called "incomparability". Hence, an important subset within the framework of partial order is the set of (x_{i1}, x_{i2}), with $x_{i1}, x_{i2} \in X$ and $x_{i1} || x_{i2}$:

$$U(IB) := \{(x_{i1}, x_{i2}) \in X^2 : x_{i1} \parallel_{IB} x_{i2},\ i1 < i2\} \qquad (2)$$

The cardinality of this finite and discrete set is called TUC_{IB}.

$$TUC_{IB} := |U(IB)| \qquad (3)$$

The subscript "*IB*" of the quantity *TUC* indicates that the incomparability of the objects x_{i1} and x_{i2} is related to the values of the elements of the information base, either taken from the data matrix in which case *IB* consists of the original indicators of the problem and will be called *IB0*, or being some consensus functions, where *IB* consists of consensus functions (see below). In order to avoid troublesome notations, we specify:

$U0 := TUC_{set\ \text{of original indicators of the data matrix}}$ and (4a)

IB0 to stress that the set of original indicators is applied (4b)

Consequently, we can write TUC_{IB0} for *U0*, i.e., for the number of incomparabilities, related to the poset based on the original indicators.

Furthermore we set:

$$m = |IB0| \qquad (5)$$

m being the number of indicators.

Another set *C* is equally important:

$$C(IB):=\{(x_{i1}, x_{i2}) \in X^2, \text{ with } x_{i1}< x_{i2}, i1 <_{IB} i2\} \quad (6)$$

This set is finite and discrete and its cardinality is called TC_{IB}.

$$TC_{IB} = |C(IB)| \quad (7)$$

Given the cardinality of the set of objects, *n*, the quantities TUC_{IB} and TC_{IB} are related as follows:

$$TC_{IB} + TUC_{IB} = n*(n-1)/2 \quad (8)$$

A decrease in *TUC* due to change *IB0* (original indicators) to *IB* (consensus functions as order preserving mappings) un ambiguously implies an increase in TC according to eqn. 8.

Through the Hasse diagram the hazards of the chemicals given in Table 1 are visualized (Fig. 1) under the set of indicators {P,B.T}.

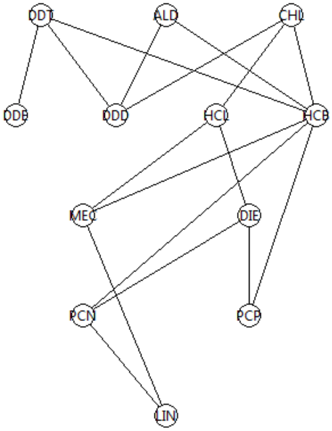


Figure 1. Hasse diagram of the 12 pesticides with simultaneous use of all three indicators, P, B and T (cf. Table 1) (software: PyHasse, see [12]).

There can be found many interesting details about the 12 chemicals, which, however, is not the focus of the present study. For the partial order visualized in Fig. 1 with $IB0 = \{P, B, T\}$ the number of incomparabilities $U0 = 31$. In a decision process most stakeholders would feel rather uncomfortable with such a high degree of incomparability as it can be regarded as an ambiguity or uncertainty in the ranking.

2.3 The general procedure to include knowledge beyond the data matrix

The general procedure can be described in 4 steps:

1*) Consensus - functions*: The very idea is to start with a linear combination of [0,1] - normalized indicator values, $q_j(x)$, for an object $x \in X$, together with scalars *g(*j*)*, called "weights" in order to derive consensus-functions (see [10] , [11] and [12]).

$$\Gamma(g, x) = \Sigma\, g(j) \cdot q(j,x) \tag{9}$$

with $\Sigma g(j) = 1$. (10)

We call $\Gamma(g, X)$ the consensus functions with all its values, corresponding to all $x \in X$.

It should be noted

a) that the approach expressed in eqn.'s (9) and (10) clearly excludes ordinal indicators as normalization and scalar multiplication with weights demands for data implying to be continuous in concept

b) that the weighted sum of indicator values is most common in Multi Criteria Decision Aids (MCDA) and is preferable in comparison to more sophisticated approaches such as PROMETHEE [13] or ELECTRE [14] because of its transparency.

c) a main disadvantage is that weighted sums (eqn. 9) have a high degree of compensation [11]. A "good" value for a certain indicator can compensate a "bad" value with respect to another indicator. Decisions should however take care for such kind of conflicts among indicators.

2) *Intervals for the weights*: If the weights, by which different indicator values for an object $x \in X$ are combined, are not exactly known (vague or blurred knowledge) then for each weight $g(j)$, intervals can be suggested. Varying the weights within these intervals by a Monte-Carlo

simulation leads to a set of weight tuples. Each single weight tuple of this set causes its own consensus function, hence a set of consensus-functions is obtained, which are considered as generating a new *IB* and therefore a new single poset.

3*) posets formed by the new IB*: This new set of consensus-functions is the basis for a new partial order of the objects, which in general is enriched in comparison to the partial order by the original indicators, as any single consensus function must preserve the order relations $x \leq_{IB} y$ based on the original data matrix, for details, see [6].

Hence, we have:

$$TUC_{IB0} \geq TUC_{IB} \quad (11)$$

with *IB0* formed by the original indicators and *IB* formed by consensus functions due to eqn.'s 9 and 10.

Clearly the selection of different sets of intervals for the weights for *m* indicators allows a great variety of partially ordered sets. Hence, it appears necessary to provide a measure to evaluate the different posets.

4) *A need of a controlling function*: It appears most appropriate for an evaluation of the set of posets (of step 3) to select the number of all incomparabilities, TUC_{IB} as a controlling quantity. In continuation and extension of the approach, described in [6], a systematic procedure is suggested and described in the following section.

2.4 The U(s-cone) - approach

The following 7 steps are introduced

1) initially start weights are defined:

$$g(\mathrm{D}) := [g(\mathrm{D},1), g(\mathrm{D},2), \ldots, g(\mathrm{D},m)] \text{ with } g(\mathrm{D},j) \in [0,1],\ j=1,\ldots,m \quad (12)$$

The *m*-tuple *g*(D) represents the sharp knowledge of all *m* weights (corresponding to *m* indicators of the original data matrix). The specification "D" indicates that the values may be related to the knowledge of the decision makers. For the sake of convenience we call *g*(D,*j*) (j = 1,…,*m*), the "start weights". The naming becomes clear, when the next steps are introduced. In most cases weights are **not** sharply known and instead of randomly, arbitrarily selecting intervals around each *g*(D,*j*) as described in [6] we define an evolution of uncertainty, with the parameter *s*, the evolution parameter (see below).

2) Weight intervals *I*(*s*,*j*)

In eqn. 13 the weight interval for each single weight *g*(*j*) is specified. Note that now we are no more speaking of the start weights, but of weights randomly taken from an interval, which is specified now:

$$I(s,j) = [g(\mathrm{LL},s,j), g(\mathrm{UL},s,j)] \tag{13}$$

where LL and UL are abbreviations for "Lower limit" and "Upper limit", respectively. With $g(\mathrm{LL},s,j) \leq g(\mathrm{UL},s,j)$ the interval) *I*(*s*,*j*) is a closed interval for each weight component *g*(*j*) for a fixed $s \in [0,1]$.

3) The concept of an s-evolution

The lower and upper limits *g*(LL,*s*,*j*) and *g*(UL,*s*,*j*) are considered as varying with the evolution parameter s, which expresses the increasing uncertainty about the weights, because the lengths of the intervals |*I*(*s*,*j*)| are increasing with *s* (eqn. 14).

$$|I(s,j)| = g(\mathrm{UL},s,j) - g(\mathrm{LL},s,j) \tag{14}$$

4) The s-evolution by a **linear** approach

For each *g*(LL,*s*,*j*) , *g*(UL,*s*,*j*) (*j*=1,…,*m*), $s \in [0,1]$ we set:

$$g(\mathrm{LL},s,j) = a(\mathrm{LL},j) \cdot s + b(\mathrm{LL},j) \text{ and } g(\mathrm{UL},s,j) = a(\mathrm{UL},j) \cdot s + b(\mathrm{UL},j) \text{ with } s \in [0,1] \tag{15}$$

with $a(\ldots)$ and $b(\ldots)$ being the coefficients of the linear dependence of the upper and lower interval limits with s. The symbols UL and LL indicate that any generalization beyond the linear evolution by s used here, can be described by other terms for instance by the. 1st and 3rd quartiles of a normal distribution for $g(j)$.

5) Specifications

An initial condition, i.e., for $s = 0$, is required, which represents a sharp knowledge, the interval length $|I(s,j)| = 0$. Consequently

$$g(\mathrm{LL},0,j) = g(\mathrm{D},j) \quad \text{for } s = 0 \quad \text{and} \quad g(\mathrm{LL},1,j) = 0 \quad \text{for } s = 1 \text{ and}$$
$$g(\mathrm{UL},0,j) = g(\mathrm{D},j) \quad \text{for } s = 0 \quad \text{and} \quad g(\mathrm{UL},1,j) = 1 \quad \text{for } s = 1. \qquad (16)$$

Equation. 16 together with the boundary conditions (eqn. 15) describe the s-evolution. Thus with an increasing s, the value of $g(\mathrm{LL},s,j)$ is decreasing whereas that of $g(\mathrm{UL},s,j)$ is increasing resulting in an enhancing of the intervals-length, $|I(s,j)|$. By eqn. 16 the coefficients of the eqn. 15 are uniquely defined:

$$a(\mathrm{LL},j) = -\, g(\mathrm{D,j}) \qquad a(\mathrm{UL},j) = 1 - g(\mathrm{D},j)$$
$$b(\mathrm{LL,j}) = g(\mathrm{D},j) \qquad b(\mathrm{UL},j) = g(\mathrm{D},j) \qquad (17)$$

Consequently, the pair of lines (cf. Fig. 2), which can be defined for each component of the tuple g(D) describes the decreasing knowledge about the weights with increasing s from 0, corresponding to sharply known weights to 1 where all weight - values are possible; thus, all possible consensus functions should be generated which should lead to a poset corresponding to *IB0* (see below).

6) The s-cone

The series of increasing intervals for each individual weight component is called the s-cone. For a demonstration, in Fig. 2 the setting for $m = 2$ and $g(\mathrm{D},1) = 0.7$ and $g(\mathrm{D},2) = 0.3$ is depicted. In section 3.4 two sets of starting weights will be used: One with equal weights for P, B and T (i.e: (= 0.333, 0.333, 0.333)) then only two lines describing the s-cone would appear. In the second set, where we used (0.2, 0.6, 0.2) as starting weights a similar picture like that shown in Figure 2 would result. At *s* = 0 the s-cones would start at 0.2 (for P and T) and at 0.6 (for the indicator B).

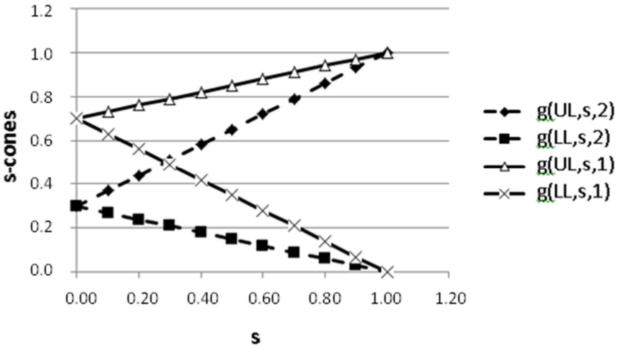


Figure 2. An example for the settings of the s-evolution of weight-intervals. For more details, see text.

In Fig. 2 at a certain *s*, (abscissa), intervals are found (cf. step 4 above). These intervals describe the values the weights *g*(*j*) can obtain by an random selection process. In case *s* = 0, the weights are coinciding with the start weights. Towards *s* = 1 the intervals are increasing and begin to overlap and some care is needed to guarantee that nevertheless the consensus-functions are made of normalized weights. Examples referring to the 12 chemicals (Table 1) are shown in the results-section.

7) Monte Carlo simulation

For every $s \in I(s,j)$ $m*MC$ weights $g(s,j) \in I(s,j)$ (j=1,..,m) are calculated, *MC* being the number of Monte Carlo simulations. The result can be thought of as a matrix with $g(j)_{mc}$ indicating the jth weight (of the jth indicator) at the mc_{th} Monte Carlo simulation (mc = 1,…,MC). Following this scheme at a certain value of *s* we obtain *MC* consensus functions $\Gamma(g,X)$ (i.e. $\Gamma(g,x)$ for all $x \in X$) (eqn. 9) and their partial order can be determined, following eqn. 1 with *IB* the set of consensus functions. The poset for this one selected *s* can be visualized by a Hasse diagram. Selecting different *s*-values, say *ns*-values $\in$ [0,1] the two-dimensional field as shown above becomes a three-dimensional one, where the third dimension corresponds to *ns* *s*-values (Fig. 3).

Table 2. Scheme of the Monte Carlo simulation

	Weights for indicators			
No of *MC*-simulation	q_1	q_2	…	q_m
1	$g(1)_1$	$g(2)_1$	…	$g(m)_1$
2	$g(1)_2$	$g(2)_2$	…	$g(m)_2$
…	…	…	…	…
MC	$g(1)_{MC}$	$g(2)_{MC}$	…	$g(m)_{MC}$

At a certain *mc*- and *s*-value one consensus-function, according to eqn. 9 is obtained leading to a weak order. As *MC* Monte Carlo simulations are performed, *MC* different consensus functions are resulting. As only the ranking due to each single consensus function is important and as each single consensus function must preserve eqn. 1, based on *IB0*, eqn. 11 holds. Therefore one one and only one enriched partial order with a correspondingly reduced TUC_{IB} is obtained.

Consequently, for all *I*(*s*,*j*) (*j*=1,…,*m*), the number of incomparabilities TUC_{IB} can be obtained as a function of *s*. To avoid cumbersome notation we call this enumeration of incomparabilities based on a systematic evolution, simply *U*(s-cone) and *U*(s-cone, s) when the dependence on *s* is to be indicated.

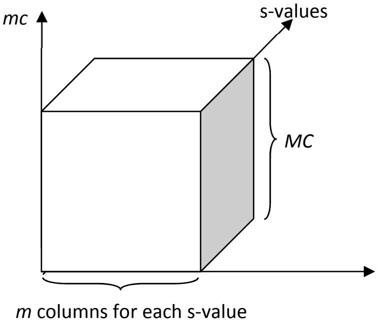


Figure 3. Cube, representing the evolution by *s* (*ns* discrete values assumed), *MC* Monte Carlo simulations and *m* original indicators due to *IB0.* A slice of the cube for a g*iven s* corresponds to a set of *MC* consensus functions and this set generates one poset.

2.5 Software

The calculations are performed, using the software package PyHasse [15], from which a simplified version is available in the Internet, see www.pyhasse.org. The main module, based on user defined intervals is HDweightMC8_7. Some more modules were developed, which allow experimenting with starting weights, which however are still far away from a status, where they can be offered to users. Under these preliminary modules the most important module is “UMC_vs_Us_4” (version 3), which is specifically developed corresponding to the s-cone concept. It delivers among others the incomparabilities at each s based on the Monte Carlo simulations.

In Fig. 4 the main new PyHasse module is shown

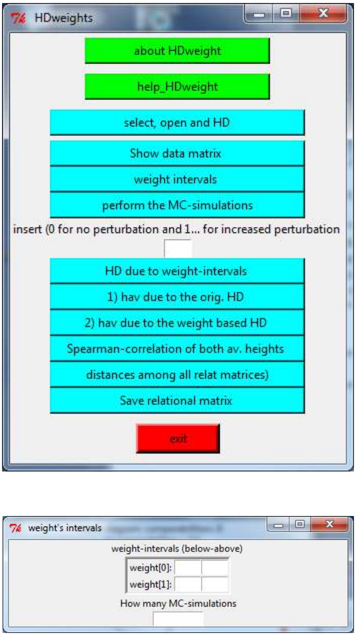

HDweights
about HDweight
help_HDweight
select, open and HD
Show data matrix
weight intervals
perform the MC-simulations
insert (0 for no perturbation and 1... for increased perturbation
HD due to weight-intervals
1) hav due to the orig. HD
2) hav due to the weight based HD
Spearman-correlation of both av. heights
distances among all relat matrices)
Save relational matrix
exit
weight's intervals
weight-intervals (below-above)
weight[0]:
weight[1]:
How many MC-simulations

Figure 4. Top: User - interface of HDweightMC8_7. Below: The mask to enter the weight intervals (*m* = 2) and *MC*

HDweightMC8_7 allows any selection of weight intervals, not only those following the s-cone concept. The selection of weight intervals can be done after pressing the button “weight intervals”. This causes mask, which is shown in Fig. 4 at the bottom. After inserting the values for the intervals of each weight $g(j)$, $j = 1,..,m$ a selection of MC has to be done. After performing the Monte Carlo simulations the Hasse diagram based on the set of consensus functions can be obtained (Button: “HD due to weight intervals”). The five buttons below are not directly related to the s-cone concept.

3 Results

3.1 Incomparabilities evolving along the s-cone in general

Corresponding to the setting (eqn.’s 14 - 16) the variety of the consensus -functions is enhanced, when s is increasing. The stability of the result should be checked by different runs with different MC) with $s \in [0,1]$. To take into account that different incomparabilities are obtained when s is varying, we write U(s-cone, s). As any new indicator will not reduce the number of incomparabilities, we can immediately deduce eqn. 18, which is nothing else than eqn. 11, but now written in terms of the s-cone concept:

$$U(\text{s-cone}, s(1)) \leq U(\text{s-cone}, s(2)) \text{ with } s(1) \leq s(2) \text{ in } [0,1] \quad (18)$$

$$U(\text{s-cone}, 0) = 0 \quad (19)$$

and theoretically:

$$U(\text{s-cone}, 1) = U0 \quad (20)$$

If $s = 1$, the Monte Carlo simulation should reproduce the poset, based on X and $IB0$, i.e.U(s-cone,1) should be $TUC_{IB0} = U0$. In practice however not all consensus functions will be obtained, even with large MC because the set of linear extensions of a poset defined by eqn. 1 is not represented by all linear functions due to eqn. 9.

3.2 Uncertainty volume

An “uncertainty volume” *V* is defined as the length ≠ 0 of each interval, according to *s* and the selected weight component:

$$V(m,s) := \prod_{j=1}^{m} |\,(I(s,j)\,| \tag{21}$$

In [7] *U*(s-cone) was already introduced and it was hypothesized that the functional form would be as follows:

$$U(V(m,s)) = U0 \;\cdot\; V(m,s)^{(1/m)} \tag{22}$$

Eqn. 22 can be analyzed in two manners:

1) As *V*(*m*,*s*) is by eqn. 21 a function of *s* (and of *m* if different data matrices with different number of indicators are explicitly accounted for) *U* becomes a function of *s* (and *m*). The uncertainty intervals are increasing with increasing *s* (eqn.’s 14 - 16) and *V*(*m*.*s*) is a fixed quantity for each *m* and *s*.

2) A given uncertainty volume is not necessarily calculated assuming the systematic evolution of *I*(*s*,j) due to eqn. 15. Any interval for the weights *g*(*j*), $j \in J$, can be used to calculate an uncertainty volume. The index set *J* encompasses those indices *j*, for which the interval *I*(*j*) has a length ≠ 0. Then instead of eqn. 22 another equation could be written:

$$U(V) = U0 * V^{(1/m)} \tag{23}$$

where *V* is calculated by:

$$V = \prod_{j \in J} I(j), \tag{24}$$

In [7] eqn. 23 was applied on the pesticide data (Table 1) to discuss several specified volumes of weight’s uncertainty. Whether or not eqn. 23 is a good representation of TUC_{IB} can be disclosed, using the data in Table 1 and calculating arbitrarily uncertainty volumes by applying eqn. 24 (Fig. 5). Along the ordinate the values of *U* are shown; the abscissa corresponds to the values of uncertainty volumes (vide supra).

The points are due to some arbitrarily selected weight intervals. The maximal and minimal values of *TUC_{IB}*, called *Urealmin* and *Urealmax*, according to each *V* are shown too. The result seems to confirm that the use of a power law for *U*(*V*) is justified.

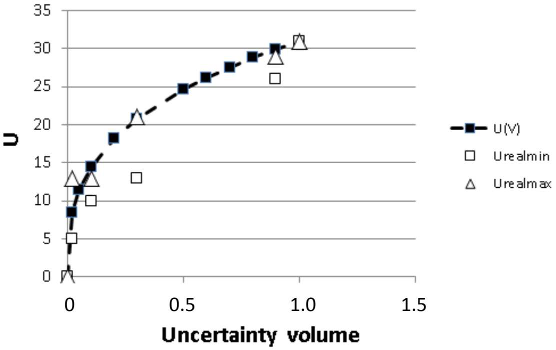


Figure 5. The drawing is based on data of Table 1. The uncertainty volumes are determined by arbitrarily selected intervals *I*(,*j*) (eqn. 24) and *U*(*V*) is calculated (broken line). More details: see text.

3.3 *U*(*V*(*m*,*s*)) as a consequence of the s-cone concept

It is reasonable to consider *U* within the systematic evolution as a linear function of *s*, which we call *U*(*s*). Regarding eqn. 19 and 20 it is reasonable to write

$$U(s) = s \cdot U0 \qquad (25).$$

U0 is (implicitly a function of *m*, but is suppressed here). Applying the eqn’s 13 to 16 it is a simple calculation to show that the length of an interval *I*(*s*,*j*) depends on *s* as follows:

$$|\mathrm{I}(s,j)| = s \qquad (26)$$

By eqn. 26 the uncertainty volume *V*(*m*,*s*) can be rewritten as follows:

$$V(m,s) = s^m \qquad (27)$$

and by eqn. 25 the power law $U(s) = V(m,s)^{(1/m)} \cdot U0$ results.

So far, an approximation for *U*(s-cone, *s*) can be established as a function of *s* via eqn. 25. Nevertheless, it should be clear, that eqn. 25, or the power law in terms of an uncertainty volume (eqn. 23) cannot completely represent the truth, because the values *q*(*j*,x) influence the enrichment process due to the consensus function (eqn. 9) as well. Thus, whether or not the set of consensus functions indicate for instance x < y or x > y, must depend on the data values of x and y. Taking this into account there are three consequences:

1. TUC_{IB} will in general deviate from *U*(*s*), exemplified by Fig. 5, because the selection of weight intervals does not follow the systematic evolution
2. Just because *U*(*s*) depends only on *U0* and *s*, the role of the indicator values (the real data) along the evolution by *s* is suppressed.
3. Given a certain value for *s* (and *m*) and *MC* Monte Carlo simulations one obtains exactly one poset and a value for TUC_{IB} = *U*(s-cone, *s*) can be directly calculated. Even then *U*(*s*) (eqn. 23) will not necessarily coincide with *U*(s-cone, *s*). Calculations are performed with the preliminary module “UMC_vs_Us” (see also section 2.5), where the values of U(s) are rendered as “Us” and the values of U(s-cone, s) as “UtotMC. .

3.4 *U*(*s*) compared with *U*(s-cone, *s*)

As can be expected, the comparison of *U*(s-cone, *s*) and *U*(*s*) is only possible by performing real calculations, i.e. checking for each selected s the resulting poset and by determining its TUC_{IB} (eqn’s 2 and 3).

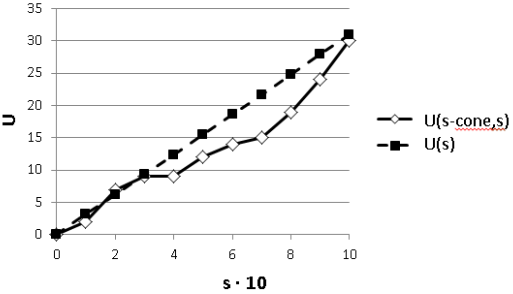


Figure 6. The number of indicators is 3 (*m* = 3); *U*(*s*) (broken line) deviates from *U*(s-cone, *s*), the abscissa corresponds to 10**s*. Number of *MC*-simulations: 10 000

The result, supposing equal start weights (i.e. *g*(D) = (1/3, 1/3, 1/3)), corresponding to the three indicators P, B and T) is shown in Fig. 6.

By further trials, i.e. further data sets it can be verified that the deviations are possible in both directions, i.e. *U*(*s*) cannot considered as an upper or lower limit of *U*(s-cone, *s*). This can be clearly demonstrated (Fig. 7), when only P and T of Table 1 are used (equal starting weights assumed, i.e. g(D) = (0.5, 0.5)):

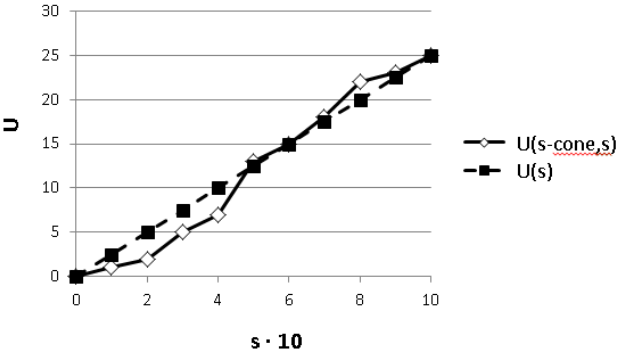


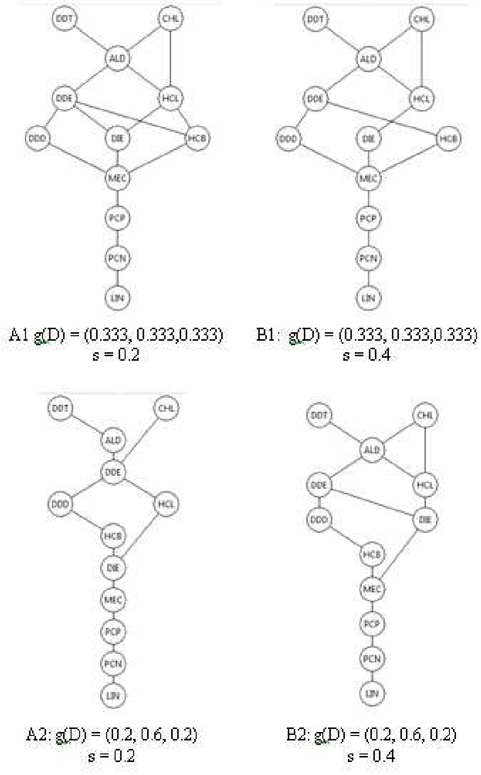
Figure 7. Number of indicators is 2 (*m* = 2); demonstration that *U*(s-cone, *s*) can deviate from *U*(*s*) in both directions. Ordinate: the incomparabilities, based on different approaches, abscissa: 10**s*

3.4 Examples, referring to the 12 chemicals

In order to demonstrate the applicability of the new PyHasse module, we studied the 12 obsolete pesticides shown in Table 1.

As exemplary cases we selected two tuples of starting weights: A: *g*(D) = (0.333,0.333,0.333) and B: *g*(D) = (0.2, 0.6,0.2), respectively. The Hasse diagrams of these four cases are shown in Figure 8.

In case A no indicator of *IB0* = {P, B, T} is preferred, whereas in case B there is a preference of indicator B, biodegradation, assuming that this indicator is somewhat more important. It is immediately seen that even with these uncertainties in the weighting, DDT and CHL are stable located as maximal elements. Further, in all four cases LIN < PCN < PCP < MEC < DIE., The remaining five chemicals are then differently arranged according to the selected g(D) and the uncertainty level, described by parameter *s*. It should be clear that any other Monte Carlo simulation will lead to slightly varied results. The main message is the reduction of TUC_{IB0} to values, which allow a better analysis of the Hasse diagrams



A1 g(D) = (0.333, 0.333,0.333)
s = 0.2

B1: g(D) = (0.333, 0.333,0.333)
s = 0.4

A2: g(D) = (0.2, 0.6, 0.2)
s = 0.2

B2: g(D) = (0.2, 0.6, 0.2)
s = 0.4

Figure 8. Hasse diagrams for four case with uncertainties, *s* = 0.2 and 0.4, respectively

4 Discussion

Here we deduce a phenomenological view from figures such as Fig. 6 or 7, where *U*(s-cone, *s*) are to be compared with *U*(*s*) as functions of the uncertainty parameter *s*:

a. Parts where *U*(*s*) and *U*(s-cone,*s*) approximately coincide: normal behavior
b. deviations where *U*(s-cone,*s*) < U(*s*): inertia of the posets generated by consensus functions against variability of weights. We interpret this as a generalized stability
c. deviations where *U*(s-cone,*s*) > U(*s*): there are weights, for which the selection of intervals will have severe influence on the posets, based on the different sets of consensus functions.

Clearly a real application of the three behaviors needs a definition as to how far a deviation of *U*(*s*) from *U*(s-cone, *s*) can nevertheless be accepted as an approximation. This and other aspects open a wide field for future tasks:

Many consequences of the approach shown in this paper are still to be investigated in more detail (for example defining appropriate limits for considering U(s) as an approximation of U(s-cone, s), however still more conceptual extensions are necessary and should be solved. The following list gives an idea, what is seen as next tasks:

1. The weakest point in this paper is for sure the problem of the characterization of the Monte Carlo simulations. Usually Monte Carlo simulations lead to distributions, which can be explored in statistical terms. Here, given a certain *s*, *MC* consensus functions are obtained, describing the values of n objects in terms of $\Gamma(g,\mathrm{x})$ $\mathrm{x} \in X$ (eqn. 9). So in principle for each object *MC* values of $\Gamma(g,\mathrm{x})$ are available. This could be a basis for a object-related distribution; however this distribution is not of primary interest, because the set of all relations x < y) is to be studied, which needs the simultaneous consideration of all consensus functions. Even if a probability can be calculated as to how far x < y is found, there is no direct access to TUC_{IB}. This is a problem which needs much more conceptual work in the future.
2. A similar problem is to find out how many Monte Carlo simulations are needed in terms of number of objects and indicators. Up to now only a trial-and-error-procedure is possible.
3. In case $m = |IB0| = 2$ crucial weights $g(\mathrm{c},j)$ can be found and an analysis could be performed, similar to that performed in [16], using the theoretical framework, outlined already in [6]. However, as the need of going beyond the partial order based on original

indicators, i.e., “endowing the skeleton of the original poset with flesh” [5] is most urgent, when $m > 2$, the question arises as to how far a concept can be developed to predict deviations of U(s-cone, s) from U(s) in terms of the original indicator values $q(j,\mathrm{x})$.

4. A possible answer could be a projection in to spaces, defined by indicator pairs (similar to [16]). Then, however, another question arises: How to project a set of starting weights in case m>2 into pairs of starting weights in two-dimensional approaches?
5. Up to now every weight obeys the evolution of the s-cone. What happens however, if the knowledge about weights is different for different indicators $q(j) \in IB0$. Within the s-cone concept this would require to discuss not one parameter s but being specific for each single indicator $q(j)$.
6. The decision, which chemical is hazardous for the environment seems to be facilitated by the new approach, because now the knowledge of decision makers can be used to enrich the partial order. The consequence is that less incomparabilities appear, so that hazardous chemicals can be more easily inspected.

Appendix. Symbols used throughout the paper

Γ	Consensus-function
$\Gamma( )$	Consensus function without specifying the object
$\Gamma(1)$, $\Gamma(2)$	Several consensus functions
X	Set of objects, $n= \|X\|$
$\Gamma(g,X)$	Consensus function with a certain weight tuple g and with values for all x $\in X$
$\Gamma(g,\mathrm{x})$	Value of a consensus function with a certain weight tuple g for an object x $\in X$.
$a(*,\mathrm{j})$, $b(*,j)$	Coefficients of a linear evolution of s referring to the jth indicator $q(j) \in IB0$. $*$ either upper or lower limit (UL or LL) of the interval.
g	Weights, not specified
$g(*,s,j)$	Weight of the jth indicator considered as function of s. $*$ either upper “UL” or lower limit, “LL”,of the interval

g(j)	jth weight , j=1,2,…,m, not specified
g(D)	Tuple of starting weights (may be given by a decision maker)
g(D,j)	jth weight given by e.g. a decision maker, component of g(D).
$I(s,j)$	Interval for the weight of the jth indicator, depending on s
m	Number of indicators in the original data matrix (=|*IB0*|)
n	= |X|
MC	Number of Monte-Carlo-simulations
IB	information basis. Set of indicators or of consensus functions
IB0	information basis, specifically referring to the indicators $q(j)$ of the original data matrix
$q(j)$	jth indicator
$q(j$,x)	Value of $q(j)$ for object x
s	Uncertainty evolution parameter
$s(*,j)$	=$a(*,j)*s + b(*,j)$, $*$ either upper or lower limit of the interval $I(s,j)$.
s-cone	Evolution of the intervals for the weights as function of s, unspecified
s-cone($*,j$)	Evolution of the interval or the weight of the jth indicator. $*$ either upper or lower limit of the interval
U	set of pairs x||y, not specified, x,y ∈ X
U(IB)	set of pairs x || y , x,y ∈ X, stressing now the dependence on *IB*.
TUC_{IB}	Number of incomparabilities obtained by constructing a poset, based on *IB*
TUC_{IB0}	Number of incomparabilities, found in the original poset, based on *IB0*.
TC_{IB}	Number of comparabilities for a certain *IB*.
U0	= TUC_{IB0}, *IB0* the set of the original indicators taken from the data matrix
U(s-cone)	=TUC_{IB} but now related to a systematic evolution of intervals by the s-cones (see below)

U(s-cone,*s*)	Like U(s-cone) but now the dependence on s is specifically indicated
U(*s*)	Number of incomparabilities as a function of *s*, calculating after *s*U0*
U(*V*)	*U* as a function of the uncertainty volume *V*, the intervals will not necessarily selected according to the s-cone-concept
V(m,s)	$\prod_{j=1}^{m}\|(I(s,j)\|$, only intervals whose length is unequal 0, following the s.- cone concept
V	$\prod_{j=J}\|(I(j)\|$, *J* the set of indices with the length of intervals for the weights unequal 0.

References

[1] A. Kerber, R. Bruggemann, Problem driven evaluation of chemical compounds and its exploration, *MATCH Commun. Math. Comput. Chem.* **73** (2015) 577–618.

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

[3] R. Bruggemann, K. Voigt, A new tool to analyze partially ordered sets – Application: Ranking of polychlorinated biphenyls and alkanes/alkenes in river Main, Germany, *MATCH Commun. Math. Comput. Chem.* **66** (2011) 231–251.

[4] R. Bruggemann, K. Voigt, Antichains in partial order, example: pollution in a German region by lead, cadmium, zinc and sulfur in the herb layer, *MATCH Commun. Math. Comput.Chem.* **67** (2012) 731–744.

[5] J. W. Owsinski, Endowing posets with flesh: If, why and how? in: M. Fattore, R. Bruggemann (Eds.), *Partial Order Concepts in Applied Sciences*, Springer, Cham, 2017, pp 3–20.

[6] R. Bruggemann, G. Restrepo, K. Voigt, P. Annoni, Weighting intervals and ranking. Exemplified by leaching potential of pesticides, *MATCH Commun. Math. Comput. Chem.* **69** (2013) 413–432.

[7] R. Bruggemann, L. Carlsen, Incomparable: What now, IV. Incomparabilities: A modelling challenge, in: M. Fattore , R. Bruggemann (Eds.). *Partial Order Concepts in Applied Sciences*, Springer, Cham, 2017, pp. 35–47.

[8] 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.

[9] M. Fattore, Functionals and synthetic indicators over finite posets, in: M. Fattore, R. Bruggemann (Eds.). *Partial Order Concepts in Applied Sciences*, Springer, Cham, 2017, pp. 71–86.

[10] G. Strassert, Das Abwägungsproblem bei multikriteriellen Entscheidungen – Grundlagen und Lösungsansatz unter besonderer Berücksichtigung der Regionalplanung. Peter Lang, Europäischer Verlag der Wissenschaften, Frankfurt am Main, 1995.

[11] G. Munda, *Social Multi–Criteria Evaluation for a Sustainable Economy*, Springer, Berlin, 2008.

[12] P. C. Fishburn, *Utility theory for Decision Making*, Robert E. Krieger Pub. Company, 1979.

[13] J. P. Brans, P. H. Vincke, A preference ranking organisation method (The PROMETHEE method for multiple criteria decision - making), *Manag. Sci.* **31** (1985) 647–656.

[14] B. Roy, The outranking approach and the foundations of the ELECTRE methods, in: B. e Costa (Ed.), *Readings in Multiple Criteria Decision Aid*, Springer, Berlin, 1990, pp. 155–183.

[15] R. Bruggemann, L. Carlsen, K. Voigt, R. Wieland, pyhasse software for partial order analysis, in: R. Bruggemann, L. Carlsen, J. Wittmann (Eds.), *Multi–Indicator Systems and Modelling in Partial Order*, Springer, New York, 2014, pp. 389–423.

[16] R. Bruggemann, K. Voigt, G. Restrepo, U. Simon, The concept of stability fields and hot spots in ranking of environmental chemicals, *Environ. Modell. Softw.* **23** (2008) 1000–1012.