T-NORM-BASED ORDERINGS ON $\bigcup_{n\in N}[0,1]^n$

Darina Kyselová

Department of Mathematics and Descriptive Geometry, Faculty of Civil Engineering
Slovak University of Technology, Radlinského 11, 813 68 Bratislava, Slovakia
e-mail: kyselova@math.sk

Summary In this paper we recall orderings based on t-norms, which could help to select the best suited decision procedure in many practical situations. We discuss several refinements of t-norm based orderings which break some of possible ties.

1. INTRODUCTION

In multi-criteria decision making we need to select the procedure of finding an optimal alternative. Alternatives are described by some vector from S^n, $n \in N$, where S is a given scale. We identify the alternatives with their score vectors and fix the number of criteria $n > 1$. Moreover, we fix $S = [0,1]$.

A widely applied method of comparing alternatives is based on aggregation operators. A mapping A: $\bigcup_{n\in N}[0,1]^n \rightarrow [0,1]$ is called an aggregation operator [1,9] on $[0,1]$ if

(i) $\forall\, n \in N$: $A(0,...,0) = 0$, $A(1,...,1) = 1$

(ii) $\forall\, n \in N$, $\forall (a_1,..., a_n), (b_1,..., b_n) \in [0,1]^n$:

$a_1 \le b_1,..., a_n \le b_n \Rightarrow A(a_1,....,a_n) \le A(b_1,...., b_n)$

(iii) $\forall\, s \in [0,1]$: $A(s) = s$.

An aggregation operator A summarizes the profile $(a_1,....,a_n)$ of **a** in a single numerical value A(**a**). This operator A induces a weak order relation $\succeq_A$ on $\bigcup_{n\in N}[0,1]^n$ by **a** $\succeq_A$ **b** $\Leftrightarrow A(\mathbf{a}) \ge A(\mathbf{b})$ [2,5] (*i.* e., $\succeq$ is a complete preorder).

$\succeq_A$ is a preference relation which characterizes the decision maker and may be split into strict (**a** $>$ **b**), and indifferent (**a** $\sim$ **b**) preference relation.

Triangular norms, known as t-norms, are important classes of aggregation operators. Triangular norm T is an associative symmetric aggregation operator with neutral element 1[10].

In the next we use t-norm T to compare two alternatives **a** and **b**. If T(**a**) > T(**b**) we say that **a** is strictly preferred to **b**. An important issue that arises in this application is the adjudication of ties. That means if T(**a**) = T(**b**) we need a way to select between the alternatives **a** and **b**.

2. MIN ORDERING

The Min-procedure $\succeq_{\min}$ compares alternatives on the basis of their worst performance

$$\mathbf{a} \succeq_{\min} \mathbf{b} \Leftrightarrow \min_i a_i \ge \min_i b_i$$

The Min is well suited for aggregating ordinal evaluations when meaningful recording into cardinal scales seem to be out of reach. Note that all aggregation-based procedures fitting to the ordinal evaluation (i.e., not depended on score transformations) were recently described in [6, 7, 8]. Recall that an ordinal scale we deal with is a finite totally ordered set S = { x_1,…, x_n }(for example, S ={weak, medium, good}), which will be isomorphically represented by a fixed scale I = {1, … , n}. On the other hand, cardinal scales are usually real intervals expressing the possible measurement outputs.

Concerning the Min-procedure, many couples (**a**, **b**) of alternatives remain indifferent. In what follows, we recall two refined comparison procedures not based directly on aggregation but invariant under common increasing transformation.

3. DISCRIMIN ORDERING

The Discrimin ordering compares alternatives on the basis of their worst performance, like the Min, but taking into account only the points of view (criteria) on which they differ.

For a fixed $n \in N$, denote I ={1,…, n }.

For **a**, **b** $\in [0,1]^n$, let D(**a, b**) = { $i \in$ I, $a_i \ne b_i$ } be the difference set, i.e. the set of the attributes on which the evaluations of alternatives **a** and **b** differ.

Then $\mathbf{a} \succeq_{\text{discrimin}} \mathbf{b} \;\Leftrightarrow\; \min_{i\in D(a,b)} a_i \ge \min_{i\in D(a,b)} b_i$

The Discrimin procedure may take into account their values on several coordinates, contrary to the Min procedure which bases the comparison on the sole smallest value.

The Discrimin ordering refines the Min ordering in the sense that:

a $\succ_{\min}$ **b** $\Rightarrow$ **a** $\succ_{\text{discrimin}}$ **b** and **a** $\succ_{\text{discrimin}}$ **b** $\Rightarrow$ **a** $\succeq_{\min}$ **b.**

Moreover, **a** $\sim_{\text{discrimin}}$ **b** $\Rightarrow$ **a** $\sim_{\min}$ **b**. Observe that Discrimin is also a complete preorder. Similarly, for any t-norm T, DiscriT ordering can be used.

Example 1: The alternatives **a** = (0.4, 0.5, 0.2) and

b = (0.4, 0.4, 0.2) are compared with Min ordering procedure:

min{0.4, 0.5, 0.2 } = 0.2 and min{0.4, 0.4, 0.2 }= 0.2 . That means **a** $\sim_{min}$ **b**. If Discrimin procedure is applied, D(**a, b**)= {2 } and min{0.5} > min {0.4} hence:

$\mathbf{a}\ \phi_{\text{discrimin}}\ \mathbf{b}$.

Note that similarly we can introduce Discrimin weak order for score vectors with different dimensions.

4. LEXIMIN ORDERING

In [2] Leximin procedure is applied by putting the components of the vectors **a** and **b** in increasing order using ≤ and then use a lexicographic comparison (or, equivalently, Discrimin ordering).

The standard notation for this set of ordered n-tuples is $[0,1]^{[n]}$.

If S is any linearly ordered set, then lexicographic order $/_L$ on S^n = {($a_1, a_2,\dots, a_n$) ∈ S ×S ×…× S } is:

- For **a, b** ∈ S^n $\mathbf{a} /_L \mathbf{b}$ if and only if either **a** = **b**, or for the least index *i* for which they differ, $a_i > b_i$.
- The **Leximin** ordering $/_{lexmin}$ is induced by the lexicographic ordering $/_L$ on $[0,1]^{[n]}$, where $[0,1]^{[n]}$ = {($a_1, a_2,\dots, a_n$) ∈ $[0,1]^n$, $a_1 \le a_2 \le \dots \le a_n$ }

Observe that **a** $\sim_L$ **b** if and only if ($a_1,\dots, a_n$) is a permutation of ($b_1,\dots, b_n$). The orderings described above have been presented in increasing order of refinement, alternatives being indifferent for an order could be distinguished by the next one.

$\mathbf{a} /_{lexmin} \mathbf{b} \Rightarrow \mathbf{a} /_{discrimin} \mathbf{b} \Rightarrow \mathbf{a} /_{min} \mathbf{b}$

Example 2: Let **a** = (0.3, 0.7, 0.5), **b** = (0.4, 0.3, 0.5).

Then **a′** = (0.3, 0.5, 0.7), **b′** = (0.3, 0.4, 0.5) and

$\mathbf{a}' /_L \mathbf{b}' \Rightarrow \mathbf{a} /_{lexmin} \mathbf{b}$

Recall that in [3] we have proposed two extending of Leximin to $\Upsilon_{n\in N}[0,1]^n$ based on occurrence (normed occurrence) function.

5. LEXIT ORDERING

In [4] the Leximin ordering is generalized to a LexiT ordering for any t-norm T.

Let $[0,1]^{[n]}$ = {($a_1, a_2,\dots, a_n$) ∈ $[0,1]^n$, $a_1 \le a_2 \le \dots \le a_n$} . Let **a** = ($a_1, a_2,\dots, a_n$) ∈ $[0,1]^{[n]}$ and T be any t-norm. Let P_a be the power set of {$a_1, a_2,\dots, a_n$}. This set contains 2^n elements. For any $F \in P_a$ let T(*F*) denotes the value of t-norm T applied to the all elements of *F*. Let $\hat{a} = (\hat{a}_1,\dots,\hat{a}_{2^n})$ be the 2^n – tuple of the family {T(*F*), $F \in P_a$ } put into ascending order, where T({a_i}) = a_i and T(∅) = 1. Thus we have $\hat{a}_1 \le \hat{a}_2 \le \dots \le \hat{a}_{2^n}$. On $[0,1]^{[2^n]}$ we have the lexicographic ordering $/_L$, which is a linear ordering. On $[0,1]^{[n]}$ we define: **a** $/_{lexiT}$ **b** if and only if $\hat{a} \ge_L \hat{b}$.

Example 3: Take the product t-norm, T(x, y) = x·y. Let **a** = (0.1, 0.4, 0.6) and **b** = (0.2, 0.2, 0.6). In this case P_a and P_b each have 8 elements.

P_a = {{0.1, 0.4, 0.6}, {0.1, 0.4}, {0.1, 0.6}, {0.4, 0.6}, {0.1}, {0.4}, {0.6}, ∅},

P_b = {{0.2, 0.2, 0.6}, {0.2, 0.2}, {0.2, 0.6}, {0.2, 0.6}, {0.2}, {0.2}, {0.6}, ∅}.

From this, $\hat{a}$ = (0.024, 0.04, 0.06, 0.1, 0.24, 0.4, 0.6, 1) and $\hat{b}$ = (0.024, 0.04, 0.12, 0.12, 0.2, 0.2, 0.6, 1). Comparing $\hat{a}$ and $\hat{b}$ lexicographically, we see that $\hat{a}_i = \hat{b}_i$ for *i* = 1, 2 and $\hat{a}_3\ \pi\ \hat{b}_3$. Thus $\hat{a} <_L \hat{b}$ and therefore **a** $<_{LexiT}$ **b**.

6. LEXIT ORDERING FOR STRICT T

The above LexiT comparison of alternatives requires 2^n steps, where one step means computing the two products $\hat{a}_i, \hat{b}_i$ and comparing them.

Let $[0,1]^{[n]}$ = {($a_1, a_2,\dots, a_n$) ∈ $[0,1]^n$, $a_1 \ge a_2 \ge \dots \ge a_n$}. Let **a** = ($a_1, a_2,\dots, a_n$) ∈ $[0,1]^{[n]}$ and T be a strict t-norm. We compute n values T($a_1, a_2,\dots, a_n$), T($a_2,\dots, a_n$) , T($a_3, a_4,\dots, a_n$), …, T(a_n) and T($b_1, b_2,\dots, b_n$), T($b_2,\dots, b_n$) , T($b_3, b_4,\dots, b_n$), …, T(b_n).

Then **a** $/_{lexiT}$ **b** if and only if (T($a_1, a_2,\dots, a_n$), T($a_2,\dots, a_n$) , T($a_3, a_4,\dots, a_n$), …, T(a_n)) $/_L$ (T($b_1, b_2,\dots, b_n$), T($b_2,\dots, b_n$), …, T(b_n)).

Example 4: Take the product t-norm, T(x, y) = x·y. Let **a** = (0.6, 0.4, 0.1) and **b** = (0.6, 0.2, 0.2). We compute (0.024, 0.04, 0.1) $<_L$ (0.024, 0.04, 0.2) ⇒ **a** $<_{LexiT}$ **b.**

Note that the above method can be applied for any weakly cancellative t-norm T (i.e., T(x, x) = T(y, y) ⇒ x = y) and that while by Yager et al. approach we should compute (and compare) 2^n values, here the computational complexity reduces to compute (and compare) n values only.

Acknowledgment

This work was supported by Science and Technology Assistance Agency under the contract No. APVT-20- 003204. A partial support of the grant VEGA 1/1145/04 is also kindly announced.

REFERENCES

[1] T. Calvo, A. Kolesárová, M. Komorníková, R. Mesiar: Aggregation Operators: Properties, Classes and Construction Methods. In: *Studies in Fuzziness and Soft Computing – Aggregation Operators, New Trends and Applications.* Physica – Verlag, Heidelberg (2002), pp. 3 – 106.

[2] D. Dubois, H. Fargier, H. Prade: *Refinements of the maximin approach to decision making in fuzzy environment,* Fuzzy set and Systems 81 (1996), pp. 103 - 122.

[3] D. Dubois, D. Kyselová, R. Mesiar: *Limit refinements of aggregation based orderings on continuous scales*. Proc. AGOP' 2005, Lugano, (2005), pp. 47 – 50

[4] R. Yager, C. Walker and E. Walker: *Generalizing Leximin to t-norms and t-conorms: the LexiT and LexiS Orderings*, Fuzzy Sets and Systems 151 (2005), pp. 327 – 340.

[5] P. Fortemps, M. Pirlot : *Conjoint axiomatization of Min, DiscriMin and LexiMin,* Fuzzy Sets and Systems 148 (2004), pp. 211 – 229.

[6] R. Mesiar, T.Rückschlossová: *Characterization of invariant aggregation operators,* Fuzzy Sets and Systems 142 (2004), pp. 63 – 73.

[7] T. Rückschlossová: *Invariant aggregation operators,* J. El. Eng 52 (2001).

[8] T.Rückschlossová: *Aggregation operators and invariantness,* PhD Thesis, STU Bratislava, (2003) .

[9] M. Komorníková, R. Mesiar, *Triangular norm-based aggregation of evidence under fuzziness*. In: B. Bouchon - Meunier, ed., Aggregation and Fusion of Imperfect Information, Physica-Verlag, Heidelberg, (1998), pp. 11 – 35.

[10] E. P. Klement, R. Mesiar, E. Pap, *Triangular norm.* Kluwer Acad. Publ., Dordrecht, (2000).