HAL
archives-ouvertes.fr

A Possibilistic Logic Approach to Conditional Preference Queries

Didier Dubois, Henri Prade, Fayçal Touazi

▶ To cite this version:

Didier Dubois, Henri Prade, Fayçal Touazi. A Possibilistic Logic Approach to Conditional Preference Queries. 10th International Conference Flexible Query Answering (FQAS 2013), Sep 2013, Granada, Spain. pp. 376-388, 2013. <hal-01212939>

HAL Id: hal-01212939
https://hal.archives-ouvertes.fr/hal-01212939

Submitted on 7 Oct 2015

A Possibilistic Logic Approach
to Conditional Preference Queries

Didier Dubois, Henri Prade, and Fayçal Touazi

IRIT, CNRS & University of Toulouse, France
{dubois,prade,faycal.touazi}@irit.fr

Abstract. The paper presents a new approach to deal with database preference queries, where preferences are represented in the style of possibilistic logic, using symbolic weights. The symbolic weights may be processed without the need of a numerical assignment of priority. Still, it is possible to introduce a partial ordering among the symbolic weights if necessary. On this basis, four methods that have an increasing discriminating power for ranking the answers to conjunctive queries, are proposed. The approach is compared to different lines of research in preference queries including skyline-based methods and fuzzy set-based queries. With the four proposed ranking methods the first group of best answers is made of non dominated items. The purely qualitative nature of the approach avoids the commensurability requirement of elementary evaluations underlying the fuzzy logic methods.

1 Introduction

One may consider that there are two main research trends in the preference queries literature, namely the fuzzy set-based approach [1,2] on the one hand, and skyline methods [3,4,5] on the other hand. Besides, in artificial intelligence, CP-nets [6] for conditional preference statements developed in the last decade have become a popular setting. It is based on a graphical representation, and obeys the ceteris paribus principle. Its potential use for dealing with preference queries has even been stressed [7]. Besides, the use of possibilistic logic for the modeling of preferences queries has been advocated more recently [8,9].

Fuzzy sets have been often proposed for the modeling of flexible queries as it provides a basis for rank-ordering the retrieved items. However, this requires the specification of membership functions, possibly of priority weights, and more importantly it is based on the implicit assumption of the commensurability of the elementary evaluations. Skyline methods single out non dominated elements, but do not rank-order them (if the query is not iterated on the remaining items), up to a few exceptions [10]. In this paper, we investigate the use of a possibilistic logic approach to the handling of data base (conditional) preference queries, which remains as symbolic as possible, but preserves a capability for rank-ordering the answers.

The paper is organized as follows. First, a short background on possibilistic logic, and the use of symbolic weights is provided in Section 2. Then, a running example is

proposed, that will be used for comparing the different methods discussed in the paper. Section 3 presents four methods for rank-ordering query answers, with an increasing discriminating power. The first method handles preferences as conditional possibility constraints. The second method, which proves to be more refined, relies on an inclusion-based ordering. In the third and fourth methods, additional constraints are added between the symbolic weights of the possibilistic logic encoding, in the spirit of CP-nets and CP-theories [11] (a generalization of CP-nets) respectively. Then Section 4 briefly compares our proposal to related work on skyline and fuzzy set approaches.

2   Technical Prerequisites and Running Example

We consider a propositional language where formulas are denoted by $p_1, ..., p_n$, and Ω denotes its set of interpretations. The logical conjunctions, disjunctions and negations are denoted by $\wedge$, $\vee$ and $\neg$, respectively.

2.1   Possibilistic Logic

Let $B^N = \{(p_j, \alpha_j) \mid j = 1, \ldots, m\}$ be a possibilistic logic base where $\alpha_j \in \mathcal{L} \subseteq [0, 1]$ is a priority level attached to formula p_i [12]. Each formula (p_j, α_j) means that $N(p_j) \geq \alpha_j$, where N is a necessity measure, i.e., a set function satisfying the property $N(p \wedge q) = \min(N(p), N(q))$. A necessity measure is associated to a possibility distribution π on the set of interpretations, as follows:

$$N(p) = \min_{\omega \notin M(p)} (1 - \pi(\omega)) = 1 - \Pi(\neg p),$$

where Π is the possibility measure associated to N and $M(p)$ is the set of models induced by the underlying propositional language for which p is true.

The base B^N is associated to the possibility distribution on interpretations:

$$\pi_B^N(\omega) = \min_{j=1,\ldots,m} \pi_{(p_j, \alpha_j)}(\omega) \tag{1}$$

where $\pi_{(p_j, \alpha_j)}(\omega) = 1$ if $\omega \in M(p_j)$, and $\pi_{(p_j, \alpha_j)}(\omega) = 1 - \alpha_j$ if $\omega \notin M(p_j)$. An interpretation ω is all the more possible as it does not violate any formula p_j having a higher priority level α_j. So, if $\omega \notin M(p_j)$, $\pi_B^N(\omega) \leq 1 - \alpha_j$, and if $\omega \in \bigcap_{j \in J} M(\neg p_j)$, then $\pi_B^N(\omega) \leq \min_{j \in J}(1 - \alpha_j)$. It is a description "from above" of π_B^N, which is the least specific possibility distribution in agreement with the knowledge base B^N.

2.2   Symbolic Weights

The weights associated to possibilistic logic formulas, which can be understood as priority or certainty levels, may be processed symbolically. By that, we mean that we are not assigning a value to the weights. So doing, we in general lose the benefit of the total ordering existing between values in a scale. Still, a partial ordering remains between symbolic expressions, e.g., we do know that $\min(\alpha, \beta) \leq \alpha$ whatever the values of α and β. Moreover, one may introduce some further constraints between symbolic

weights, when available, e.g., $\alpha > \beta$, and exploit them. This includes the particular case where one knows the complete ordering between all the symbolic weights introduced. Generally speaking, when several constraints are introduced, it is important to make sure that they are consistent.

Since one may not know precisely how imperative preferences are in general, it is convenient to handle weights in a symbolic manner, without having to assign precise values. Having symbolic weights still allows us to construct a vector for each outcome that will serve in their rank-ordering. Indeed, a query will be viewed as a (weighted) conjunction of logical formulas, and items in the database are then rank-ordered according to the level to which they satisfy this conjunction. Then, the vector components are nothing but the arguments of the $\min$ in equation (1) defining the semantics of a possibilistic base made of formulas (p_j, α_j) expressing goals and their importance. In this paper we explain how these vectors are obtained, and discuss how vectors can be ordered on this basis.

2.3   Running Example

Throughout the paper, we will use the following running example in order to illustrate the proposed approach to preference queries. This example is implemented on an experimental platform in information processing developed at IRIT in Toulouse (*http:/www.irit.fr/PRETI*) (see [13]). The data base stores pieces of information about houses to let that are described in terms of 25 attributes.

Example 1. *We want to express the following preferences:*

 – *The number of persons accommodated should be more than 10, imperatively;*
 – *It is preferred to have a house where animals are allowed,*
 – *It is preferred to be close to the sea by a distance between 1 and 20 km;*
 – *If the house is far from the sea by more than 20 km, it is preferred to have a tennis court at less than 4 km*
 – *If moreover the distance of the house to the tennis court is more than 4 km, it is desirable to have a swimming pool be at a distance less than 6 km*

These preference constraints can be encoded by the following possibilistic logic formulas. Indeed, in our approach any query is represented by a possibilistic logic base. Here there is one imperative constraint, the other constraints being soft.

  - Hard preference constraint
    - $\phi_0 = (Accomod. \geq 10, 1)$
  - Soft preference constraints
    - $\phi_1 = (Animal, \alpha_1)$
    - $\phi_2 = (1 \leq Sea \leq 20, \alpha_2)$
    - $\phi_3 = (\neg(Sea > 20) \vee Tennis \leq 4, \alpha_3)$
    - $\phi_4 = (\neg(Sea > 20) \vee \neg(Tennis \leq 4) \vee Pool \leq 5, \alpha_4)$

3   Handling Preference Queries

What makes the possibilistic logic setting particularly appealing for the representation of preferences is not only the fact that the language incorporates priority levels explicitly, but the existence of different representation formats [14,15], equally expressive

[16,17], even if more or less natural or suitable for expressing preferences. Namely, preferences can be represented as prioritized goals, i.e. possibilistic formulas, or in terms of conditionals (i.e. statements of the form if p is true then having q true is preferred to having it false), or even as a Bayesian-like networks, since a possibilistic logic base can be encoded either as a qualitative or a quantitative possibilistic network and vice-versa [18]. In the next subsection, we recall how conditional preferences are represented in possibilistic logic with symbolic weights. Then in the three next subsections, different ways of handling symbolic priorities for processing the queries are discussed.

3.1   Preference Encoding in Possibilistic Logic

The unconditional preference of the form "q is preferred to $\neg q$" may be understood in the possibility theory setting as the constraint $\Pi(q) > \Pi(\neg q)$, which expresses that at least one model of q is preferred to any interpretation that makes q false. More generally, the possibilistic encoding of conditional preferences of the form "in context p, q is preferred to $\neg q$" is a constraint of the form $\Pi(p \wedge q) > \Pi(p \wedge \neg q)$. This includes the previous case where p is a tautology. Using conditioning, this constraint is still equivalent to $\exists \alpha$ s. t. $N(q|p) \geq \alpha > 0$, where $N(q|p) = 1 - \Pi(\neg q|p)$, such that $\Pi(r|p) = 1$ if $\Pi(p \wedge r) \geq \Pi(p \wedge \neg r)$ and $\Pi(r|p) = \Pi(p \wedge r)$ otherwise.

This constraint can be encoded by the possibilistic formula $(\neg p \vee q, \alpha)$, which expresses the requirement $N(\neg p \vee q) \geq \alpha$, which is itself equivalent here to the above constraint on the conditional necessity measure $N(q|p) \geq \alpha$ (see, e.g., [12]).

More generally, if we need to consider more than two mutually exclusive alternatives, this can be encoded by means of several possibilistic formulas. For instance, the two weighted formulas $\{(\neg p \vee q \vee r, 1), (\neg p \vee q, \alpha)\}$ state that if p is true, it is imperative to have $q \vee r$, and that q is preferred to $r \wedge \neg q$ since $\alpha > 0$. This extends to n alternatives. For instance, $\{(\neg p \vee q \vee r \vee s, 1), (\neg p \vee q \vee r, \alpha), (\neg p \vee q, \beta)\}$ with $\beta < \alpha < 1$ says that in context p, one wants to have q true, or at least r true, or at least s true; see [9] for further discussions.

In the next subsections, we shall exhibit different methods to rank-order outcomes based on a possibilistic logic base encoding preferences, but we first discuss the direct exploitation of constraints of the form $\Pi(p \wedge q) > \Pi(p \wedge \neg q)$.

3.2   Weak Comparative Preferences

The handling of a set of possibilistic constraints ϕ_i of the form $\Pi(p_i \wedge q_j) > \Pi(p_i \wedge \neg q_j)$ amounts here to looking for the largest possibility distribution π compatible with these constraints applying the minimum specificity principle, see, e.g., [14]. The largest solution π, which always exists if the set of constraints is consistent, can be computed using Algorithm 1 below, and represents a preference profile that rank-orders interpretations, in agreement with the preference requirements. The minimal specificity principle ensures that all the constraints are satisfied, but only these constraints are (in other words, no extra preferences are introduced). In the algorithm, the possibility distribution π is represented as a well-ordered partition $(E_1, ..., E_m)$ of Ω, associated with the ordering $\succ_{WCP}$ such that: $\forall \omega, \omega' \in \Omega$, $\omega \succ_{WCP} \omega'$ iff $\omega \in E_i, \omega' \in E_j$ and $i < j$.

The well-ordered partition given in Algorithm 1 satisfies the minimum specificity principle. The most satisfactory set E_1 is made of the interpretations that satisfy some $L(\phi_i)$, and do not satisfy any $R(\phi_j)$. Then the set of constraints whose left part is satisfied by an interpretation of E_1 are deleted, and the procedure is iterated on the remaining constraints as long as there are some. This procedure yields a possibility distribution

---

Algorithm 1. Minimal specificity ranking algorithm

Require: Φ a set of constraints ϕ_i of the form: $\Pi(L(\phi_i)) > \Pi(R(\phi_i))$
 E_i a set of interpretations classified at the ith rank
 Ω the set of all interpretations
 m=1;
 while $\Phi \neq \emptyset$ **do**
     Put in E_m any interpretation ω_l that satisfies some $L(\phi_i)$, and does not satisfy any $R(\phi_j)$
     if $E_m = \emptyset$ **then**
         The preference base is inconsistent
     else
         Delete E_m from Ω
         Delete all pairs $(L(\phi_i), R(\phi_i))$ from Φ such that $L(\phi_i)$ is satisfied by at least one element of E_m
         m=m+1;
     end if
 end while
 return $\{E_1, \cdots, E_m\}$

---

whose number of values is at most $n+1$ where n is the number of constraints. Indeed, it is clear that at least one constraint is deleted at each iteration step.

Proposition 1. *Let a query Q composed of n preference constraints. The number of elements of the well-ordered partition $\{E_1, \cdots, E_m\}$ produced by Algorithm 1 is at most $m = n + 1$.*

Example 2. *Let 4 preference constraints be given as follows ($L(\phi_i), R(\phi_i)$ are replaced by sets of interpretations): $\phi_1 = (\{t_1, t_2, t_3\}, \{t_4, t_5, t_6, t_7, t_8\})$; $\phi_2 = (\{t_4, t_5\}, \{t_6, t_7, t_8\})$; $\phi_3 = (\{t_6\}, \{t_7\})$; $\phi_4 = (\{t_7\}, \{t_8\})$. Applying Algorithm 1 gives 5 preference levels: $E_1 = \{t_1, t_2, t_3\}$, $E_2 = \{t_4, t_5\}$, $E_3 = \{t_6\}, E_4 = \{t_7\}$ and $E_5 = \{t_8\}$.*

Example 1 (continued). The imperative preference constraint (ϕ_1) restricts the list to 15 houses. Considering the preference constraints in the running example the result of applying Algorithm 1 to constraints of the form $\Pi(\phi_i) > \Pi(\neg\phi_i), i = 1, \ldots, 4$ is given in Table 1 where $m = 2$ and $E_1 = \{539\}$. As there is no conflict detected by the algorithm, there are only two classes of outcomes.

3.3 Lexicographic Comparaison

We now consider a possibilistic logic encoding of the preference requirements, i.e., a possibilistic logic base Σ. For each interpretation ω, we can build a vector $\boldsymbol{\omega}(\Sigma)$ in the following way, for each preference constraint ϕ_i for $i = 1, \cdots, n$:

Table 1. Weak comparison preference ranking

Table 2. Lexicographic ranking

Id	Sea	Animal	Tennis	Rank
539	4.00	1	3.00	1
119	100	0	0.50	2
191	100	1	0.60	2
261	83.00	1	4.50	2
320	23.00	1	1.50	2
339	100	1	8.50	2
366	100	0	0.50	2
434	100	0	3.50	2
435	89.00	1	6.50	2
507	83.00	1	4.00	2
519	58.00	0	1.50	2
530	100	1	0.50	2
536	83.00	0	1.50	2

Id	ϕ_0	ϕ_1	ϕ_2	ϕ_3	ϕ_4	Rank
539	1	1	1	1	1	1
191	1	1	$1-\alpha_2$	1	1	2
320	1	1	$1-\alpha_2$	1	1	2
530	1	1	$1-\alpha_2$	1	1	2
119	1	$1-\alpha_1$	$1-\alpha_2$	1	1	3
261	1	1	$1-\alpha_2$	$1-\alpha_3$	1	3
339	1	1	$1-\alpha_2$	1	$1-\alpha_4$	3
366	1	$1-\alpha_1$	$1-\alpha_2$	1	1	3
434	1	$1-\alpha_1$	$1-\alpha_2$	1	1	3
435	1	1	$1-\alpha_2$	$1-\alpha_3$	1	3
507	1	1	$1-\alpha_2$	$1-\alpha_3$	1	3
519	1	$1-\alpha_1$	$1-\alpha_2$	1	1	3
536	1	$1-\alpha_1$	$1-\alpha_2$	1	1	3

- if ω satisfies ϕ_i we put '1' in the ith component of the vector;
- otherwise, we put $1 - \alpha_i$ (α_i is the weight associated to preference constraint ϕ_i).

in agreement with the minimally specific possibility distribution π associated with Σ (see Section 2). Indeed, since we are dealing with symbolic weights, we cannot compute the result of the $\min$ operator aggregation of the vector components. So, we keep the vectors as they are, and order them using the classical lexicographic ordering, see, e.g., [12], thus defining an order denoted $\succ_{leximin}$ between vectors.

In the standard case of a totally ordered scale, the $leximin$ order is defined by first reordering the vectors in an increasing way, and then applying the $\min$ order to the subparts of the reordered vectors without identical components. Since we deal with a partial order over the priority weights (at least, we know that $1 > 1 - \alpha_i, \forall i$, and $1 - \alpha_i \geq \min(1 - \alpha_i, 1 - \alpha_j)$ and so on), the reordering of vectors is no longer unique, and we have to generalize the definition in the following way:

Definition 1 (leximin). *Let v and v' be two vectors having the same number of components. First, delete all pairs (v_i, v'_j) such that $v_i = v'_j$ in v and v' (each deleted component can be used only one time in the deletion process). Thus, we get two non overlapping sets $r(v)$ and $r(v')$ of remaining components, namely $r(v) \cap r(v') = \emptyset$. Then, $v \succ_{leximin} v'$ iff $\min(r(v) \cup r(v')) \subseteq r(v')$ (where $\min$ here returns the set of minimal elements of the partial order between the priority weights).*

Example 1 (continued). When applying the possibilistic logic semantics to query evaluation, we deal not only with interpretations, but also with items (several items may correspond to the same interpretation of the requirement). Thus, considering the house with id 339, its associated vector is $v(339) = (1, 1, 1 - \alpha_2, 1, 1 - \alpha_4)$ (see Table 2). The house satisfies the two first preference constraints (number of people accommodated and animals allowance) and also satisfies the 5th preference concerning the distance to a swimming pool. But it falsifies the preference about distance to the sea (it is 10 km far), and it falsifies the preference about the distance to a tennis court. Now let us compare this house with the house with id 292 and vector $v(292) = (1, 1, 1 - \alpha_2, 1, 1)$,

applying the $leximin$ order. Then, the reduced associated vectors have one component here: $r(292) = (1)$ and $r(339) = (1 - \alpha_4)$. Then, we have $\min(r(292) \cup r(339)) = \{1 - \alpha_4\} \subseteq r(339)$. So, $v(292) \succ_{leximin} v(339)$, and by extension, we write $house_{292} \succ_{leximin} house_{339}$.

Considering all the items in the running example, the result of the lexicographic comparison over the 15 houses is given in Table 2.

One can observe that $\succ_{leximin}$ may induce up to $n + 1$ layers, since an item may violate $0, 1, \cdots,$ or n preference constraints. Indeed, items are ranked according to the number of preferences violated.

Proposition 2. *If a query Q is composed of n preference constraints, then the maximal number of levels generated by $\succ_{leximin}$ is $n + 1$.*

Contrary to what Tables 1 and 2 suggest, $\succ_{leximin}$ does not refine $\succ_{WCP}$ as the latter generally introduces constraints between weights that are not present in the method of this section. However in the running example, the WCP is equivalent to applying classical logic, ending up in two classes of interpretations only. In that special case, $\succ_{leximin}$ trivially refines $\succ_{WCP}$, since then the latter separates outcomes ω that satisfy all constraints from those that violate at least one of them, while $\succ_{leximin}$ always classifies outcomes in terms of the number of violated constraints. However, $\succ_{leximin}$ does not use priorities induced by the WCP approach (Algorithm 1). Of course it is also possible to refine the ordering of outcomes induced by WCP using $\succ_{leximin}$, or equivalently to refine the $\succ_{leximin}$ with symbolic weights, by exploiting the priorities found by Algorithm 1.

4 Adding Constraints between Symbolic Weights

In the previous subsection, the partial order between priority weights, underlying the use of the lexicographic comparison, was not requiring any information on the relative values of the symbolic weights associated with the preference requirements. It should be clear that the lexicographic ordering between vectors (and thus between interpretations, and between items) will be refined by the knowledge of some additional information on the relative importance of requirements. For instance, if being not too far from a tennis court is less important than being somewhat close to the sea, then we can enforce in comparisons that $\alpha_i > \alpha_j$, where α_i, α_j are the respective weights associated to the tennis and sea requirements. However, it is important to keep in mind that when we consider two possibilistic logic formulas (φ, α) and (ψ, β) such that $\varphi \vDash \psi$ then we should have $\beta \geq \alpha$. This is in agreement with the fact that if one requires $\varphi = \neg p \vee q$ and $\psi = \neg p \vee q \vee r$ (i.e. in context p, q must be true, or at least r), satisfying φ cannot be more important than satisfying ψ if we do not want to trivialize the latter requirement ψ (since satisfying φ entails satisfying ψ). To ensure this kind of coherence property, one may compute the degree to which each requirement is entailed by the other ones (which may result in attaching to formulas symbolic expressions involving max and min of other symbolic weights).

If no extra information is available between priorities, one may apply some general principle for introducing inequalities between symbolic weights. In the following we

discuss two options that enable us to obtain a more refined leximin-based ordering. The first option is inspired from the CP-net representation of preferences [6], and the second one from its refinement in terms of CP-theories [11].

4.1 Constraints between Weights in CP-net Style

This method is inspired from CP-nets, a well-known framework for representing preferences in AI [6]. It is a graphical representation that exploits conditional preferential independence in structuring the preferences provided by a user. These preferences take the form $u : x_i > \neg x_i$, i.e., x is preferred to $\neg x$ in context u, (u can be tautological). CP-nets are underlain by a ceteris paribus principle that amounts to giving priority to preferences attached to parent nodes over preferences attached to children nodes in the CP-net structure. Besides, it has been noticed that a CP-net ordering can be approximated by a possibilistic logic representation with symbolic weights [19,8]. The priority in favor of father nodes carries over to the possibilistic setting in the following way. For each pair of formulas of the form $(\neg u \vee x_i, \alpha_i)$ and $(\neg u \vee \neg x_i \vee x_j, \alpha_j)$, x_i plays the role of the father of x_j in a CP-net. Indeed, the first formula expresses a preference in favor of having x_i true (in context u), while in the second formula the context is refined from u to $u \wedge x_i$, which establishes a particular type of links between the two formulas where the second formula is in some sense a descendant of the first one. Then, the following constraint between the corresponding weights is applied $\alpha_i > \alpha_j$, in a CP-net spirit. These constraints between symbolic weights can be obtained systematically by Algorithm 2, which computes the partial order between symbolic weights from a possibilistic logic base. Applying this procedure allows us to add constraints among symbolic weights and to get a more refined ranking of items, as we notice in the following example.

Algorithm 2. Relative importance between possibilistic formulas in a CP-net spirit

---

Require: C a set of constraints of the form (p_i, α_i)
  CBW=∅: the set of constraints between weights
  for c_j in C **do**
    if c_j is of the form (u_i, α_j) **then**
      for c_k in C **do**
        if c_k is of the form $(\neg u_i \vee x_i, \alpha_k)$ **then**
          CBW ← CBW $\cup \{\alpha_j > \alpha_k\}$
        end if
      end for
    end if
  end for
  return CBW

Example 1 (continued). Considering the preference constraints in the running example, the result of the lexicographic comparison of vectors adding the CP-nets-like constraints between weights, namely here $\alpha_2 > \alpha_3$ and $\alpha_3 > \alpha_4$, is given in Table 3, where a more refined ranking is obtained. In particular, house 339 is preferred to house 261,

Table 3. Lexicographic ranking with additional CP-net constraints

Id	Sea	Animal	Tennis	Weights	Rank
539	4.00	1	3.00	1	1
191	100	1	0.60	$1-\alpha_2$	2
320	23.00	1	1.50	$1-\alpha_2$	2
530	100	1	0.50	$1-\alpha_2$	2
119	100	0	0.50	$1-\alpha_1,1-\alpha_2$	3
339	100	1	8.50	$1-\alpha_2,1-\alpha_4$	3
366	100	0	0.50	$1-\alpha_1,1-\alpha_2$	3
434	100	0	3.50	$1-\alpha_1,1-\alpha_2$	3
519	58.00	0	1.50	$1-\alpha_1,1-\alpha_2$	3
536	83.00	0	1.50	$1-\alpha_1,1-\alpha_2$	3
261	83.00	1	4.50	$1-\alpha_2,1-\alpha_3$	4
435	89.00	1	6.50	$1-\alpha_2,1-\alpha_3$	4
507	83.00	1	4.00	$1-\alpha_2,1-\alpha_3$	4

Table 4. Lexicographic ranking with additional CP-theory constraints

Id	Sea	Animal	Tennis	Weights	Rank
539	4.00	1	3.00	1	1
191	100	1	0.60	$1-\alpha_2$	2
320	23.00	1	1.50	$1-\alpha_2$	2
530	100	1	0.50	$1-\alpha_2$	2
339	100	1	8.50	$1-\alpha_2,1-\alpha_4$	3
261	83.00	1	4.50	$1-\alpha_2,1-\alpha_3$	4
435	89.00	1	6.50	$1-\alpha_2,1-\alpha_3$	4
507	83.00	1	4.00	$1-\alpha_2,1-\alpha_3$	4
119	100	0	0.50	$1-\alpha_1,1-\alpha_2$	4
366	100	0	0.50	$1-\alpha_1,1-\alpha_2$	5
434	100	0	3.50	$1-\alpha_1,1-\alpha_2$	5
519	58.00	0	1.50	$1-\alpha_1,1-\alpha_2$	5
536	83.00	0	1.50	$1-\alpha_1,1-\alpha_2$	5

435, and 507 since $1 - \alpha_4 > 1 - \alpha_3$. Houses 119, 366, 434, 519, 536 are clearly not as good as houses 539, 191, 320, and 530; moreover they can be compared with neither house 339, nor with houses 261, 435, and 507 (since α_1 cannot be compared with α_4 or α_3); this is why houses 119, 366, 434, 519, 536 are put in the highest possible layer: i.e., below the lowest one where houses are preferred to them and in the highest one where there is an incomparable item.

4.2 Constraints between Weights in CP-Theories Style

CP-theories as introduced in [11], are a generalization of CP-nets. Also based on a graphical representation, CP-theories offer a more expressive language where preference priority can be made explicit between the preference constraints. Thus, such constraints have the same form as in CP-nets $u : x > \neg x \; [W]$; in addition we have the set of variables (attributes) W for which it is known that the preference associated to x does not depend on any value assignment of an attribute in W (i.e., the preference attached to the concerned attribute holds *irrespective* of values of attributes in W). It has been suggested that possibilistic logic is able to approximate this representation by adding more priority constraints over the symbolic weights [20]. Formally, a possibilistic preference constraint of the form $u : x > \neg x \; [W]$, with an *irrespective* requirement w. r. t. variables in W is encoded by a possibilistic preference statement $(\neg u \vee x, \alpha_i)$, to which we shall add the constraint $\alpha_i > \alpha_j$ for any α_j over symbolic weights, such that $(\neg u \vee w, \alpha_j)$ is a possibilistic preference statement, with the same context u, over one variable (or more) $w \in W$. These constraints over weights can be obtained by Algorithm 3.

Example 1 (continued). We consider the preference constraints in the running example. In addition, it is natural to assume the preference for animals allowance holds irrespectively of the preference concerning the distance to the sea ($\top : Animals > \neg Animals \; [Sea]$). Then, Algorithm 3 yields $\alpha_1 > \alpha_2 > \alpha_3 > \alpha_4$ and the result of applying lexicographic comparison over the 15 houses, is given in Table 4, which leads

here to an even more refined ranking. We can establish that the different ranking procedures discussed so far agree on the best selected items. Besides, we can show that the maximal number of layers induced by the lexicographic ordering may be greatly increased by the presence of additional constraints:

Proposition 3. *Let E^1 denote the set of non dominated models of a consistent possibilistic base. This set remains unchanged under the weak comparative preference ordering and the lexicographic ordering (in the presence of additional constraints or not).*

$$E^1_{\succ_{WCP}} = E^1_{\succ_{leximin}}$$

Proof of Proposition 3: Let $E^1_{\succ_{WCP}}$ be the set of non dominated interpretations obtained by the weak comparative preferences method. We know that $\omega \in E^1_{\succ_{WCP}}$ if only if ω satisfies all preference constraints. Let $\boldsymbol{\omega}(\Sigma)(1, \cdots, 1)$ be the vector associated to ω. It is clear that any item that has an associated vector only made of '1' components is preferred to any other vector containing at least one component $\neq$'1' according to *leximin* order. So $\omega \in E^1_{\succ_{leximin}}$. $\square$

Algorithm 3. Relative importance between possibilistic formulas in a CP-theories spirit

---

Require: C a set of constraints of the form (P_i, α_i)
  for $c_i \in C$ a preference constraint associated with an irrespective requirement **do**
    if $W_i = \emptyset$ **then**
      CBW ← CBW ∪ **Algorithm 2**
    else
      for c_j in C **do**
        if c_j is of the form $(\neg u_i \vee \neg x_i \vee v, \alpha_j)$ or $(\neg u_i \vee z, \alpha_j)/z \in W_i, v \in \{V - U\}$
then
          CBW ← CBW ∪ $(\alpha_i > \alpha_j)$
        end if
      end for
    end if
  end for
  return CBW

4.3 Hybridizing Weak Comparative Preferences and Lexicographic Methods

As shown in the previous subsections, the above three leximin-based methods lead to different, but compatible rankings of items, with increasing discrimination. The respective complexities of the comparative preferences and lexicographic methods are $Polynomial$ and $\Pi_2^P - complete$ [21,22]. Indeed, it can be observed in practice that the lexicographic method is more costly from a computational point of view:

Proposition 4. *Let Q be a query made of n preference constraints, then the maximal number of levels generated by $\succ_{leximin}$ with additional constraints over weights is 2^n.*

Indeed, this number of layers is obtained by refining the ordering of items violating the same number of preferences in case of a total ordering of the weights. Since the

Table 5. Comparative Table of different approaches dealing with preference queries

	Formulation		Context		Ranking	
	Qualitat.	Quantitat.	Uncond. req.	Cond. req.	Skyli.	Top-k
Lacroix Lavency [24]	✓		✓			✓
Chomicki 2002 [4]	✓		✓	✓	✓	
Kießling 2002 [5]	✓	✓	✓		✓	
Fagin et al 2001 [2]		✓	✓			✓
Fuzzy logic [13]		✓	✓	✓		✓
Symbolic weight possibilistic logic	✓		✓	✓	✓	✓

lexicographic method leads to a more refined rank-ordering, one may think of first using the weak comparative preferences method to stratify items, and then, each layer (except the top one because of Proposition 3) may be refined by one of the lexicographic methods by considering each level as a new intermediate database. This hybrid method may be of interest for computing a rank-ordering for top k items (when k is larger than $|E^1|$!). However, for refining the ranking of items inside a layer, we need to process it as a whole, even if it leads to considering more than k items in the ranking.

5   Related Work

Different types of approaches for handling preference queries have been proposed in the literature. Table 5 (whose evaluation criteria are taken from [23]) provides a comparative assessment of the possibilistic approach along with a representative subset of other approaches.

As for other qualitative methods, the approach presented is capable of expressing preference between attribute values or between tuples of attribute values, since we use general logical formulas (e.g., considering only the two attributes 'Price' and 'Distance to sea' with values *low, medium* and *high* for a house to let, a query may express that one prefers *low price and low distance*, or at least *low price and medium distance* or at least *medium price and medium distance*). As can be noticed, only the Kießling approach [5] can express both qualitative and quantitative preferences. Besides, only the possibilistic and the Chomicki [4] approaches can deal with conditional preferences. It can be seen that the advantages of this approach and of the fuzzy logic one [25] are complementary, which suggests to try and hybridize them in the future. To this end, it would be necessary to compare vectors including both symbolic and numerical weights.

6   Conclusion

The interest for preference representation in the possibilistic logic framework first stems from the logical nature of the representation. Moreover, the possibilistic representation can express preferences of the form "or at least", or "and if possible"(see [20] for an introductory survey), and can handle partial orders thanks to the use of symbolic weights, without enforcing implicit preferences (as it is the case for father node preferences in CP nets). We have proposed three types of methods in order to rank-order items, which

are characterized by an increasing refinement power with manageable complexity, especially using the hybrid method explained in Subsection 4.3. Still, much remains to be done. First, the use of symbolic weights is really advantageous but we still miss some properties of numerical weights. One may think of combining these two formats to be as much expressive as possible. Moreover, this approach should be able to deal with null values, which create specific difficulties in preference queries.

References

1. Bosc, P., Pivert, O.: Some approaches for relational databases flexible querying. Journal of Intelligent Information Systems 1, 323–354 (1992)
2. Fagin, R., Lotem, A., Naor, M.: Optimal aggregation algorithms for middleware. In: Proc. 20th ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Syst. (2001)
3. Börzsönyi, S., Kossmann, D., Stocker, K.: The skyline operator. In: Proc. 17th IEEE International Conference on Data Engineering, pp. 421–430 (2001)
4. Chomicki, J.: Preference formulas in relational queries. ACM Transactions on Database Systems 28, 1–40 (2003)
5. Kiessling, W.: Foundations of preferences in database systems. In: Proc. of the 28th International Conference on Very Large Data Bases (VLDB 2002), pp. 311–322 (2002)
6. Boutilier, C., Brafman, R.I., Domshlak, C., Hoos, H., Poole, D.: CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. J. Artificial Intelligence Research (JAIR) 21, 135–191 (2004)
7. Brafman, R.I., Domshlak, C.: Database preference queries revisited. Technical Report TR2004-1934, Cornell University, Computing and Information Science (2004)
8. HadjAli, A., Kaci, S., Prade, H.: Database preference queries - A possibilistic logic approach with symbolic priorities. Ann. Math. Artif. Intell. 63, 357–383 (2011)
9. Bosc, P., Pivert, O., Prade, H.: A possibilistic logic view of preference queries to an uncertain database. In: Proc. IEEE Inter. Conf. on Fuzzy Systems (FUZZ-IEEE 2010), Barcelona, Spain, July 18-23, pp. 1–6 (2010)
10. Hadjali, A., Pivert, O., Prade, H.: On different types of fuzzy skylines. In: Kryszkiewicz, M., Rybinski, H., Skowron, A., Raś, Z.W. (eds.) ISMIS 2011. LNCS, vol. 6804, pp. 581–591. Springer, Heidelberg (2011)
11. Wilson, N.: Computational techniques for a simple theory of conditional preferences. Artif. Intell. 175, 1053–1091 (2011)
12. Dubois, D., Prade, H.: Possibilistic logic: a retrospective and prospective view. Fuzzy Sets and Systems 144, 3–23 (2004)
13. de Calmès, M., Dubois, D., Hüllermeier, E., Prade, H., Sedes, F.: Flexibility and fuzzy case-based evaluation in querying: An illustration in an experimental setting. Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 11, 43–66 (2003)
14. Benferhat, S., Dubois, D., Prade, H.: Towards a possibilistic logic handling of preferences. Applied Intelligence 14, 303–317 (2001)
15. Dubois, D., Kaci, S., Prade, H.: Representing preferences in the possibilistic setting. In: Bosi, G., Brafman, R.I., Chomicki, J., Kießling, W. (eds.) Preferences: Specification, Inference, Applications. Number 04271 in Dagstuhl Seminar Proceedings (2006)
16. Benferhat, S., Dubois, D., Kaci, S., Prade, H.: Bridging logical, comparative, and graphical possibilistic representation frameworks. In: Benferhat, S., Besnard, P. (eds.) ECSQARU 2001. LNCS (LNAI), vol. 2143, pp. 422–431. Springer, Heidelberg (2001)
17. Benferhat, S., Dubois, D., Kaci, S., Prade, H.: Graphical readings of possibilisitc logic bases. In: 17th Conf. Uncertainty in AI (UAI 2001), Seattle, August 2-5, pp. 24–31 (2001)

18. Benferhat, S., Dubois, D., Garcia, L., Prade, H.: On the transformation between possibilistic logic bases and possibilistic causal networks. Inter. J. of Approx. Reas. 29, 135–173 (2002)
19. Dubois, D., Prade, H., Touazi, F.: Conditional preference nets and possibilistic logic. In: van der Gaag, L.C. (ed.) ECSQARU 2013. LNCS, vol. 7958, pp. 181–193. Springer, Heidelberg (2013)
20. Dubois, D., Prade, H., Touazi, F.: Handling partially ordered preferences in possibilistic logic. In: ECAI 2012 Workshop on Weighted Logics for Artificial Intelligence, pp. 91–98 (2012)
21. Benferhat, S., Yahi, S.: Complexity and cautiousness results for reasoning from partially pre-ordered belief bases. In: Sossai, C., Chemello, G. (eds.) ECSQARU 2009. LNCS, vol. 5590, pp. 817–828. Springer, Heidelberg (2009)
22. Yahi-Mechouche, S.: Raisonnement en présence d'incohérence: de la compilation de bases de croyances stratifiées à l'inférence à partir de bases de croyances partiellement préordonnées. Université d'Artois Faculté des Sciences Jean Perrin, Lens (2009)
23. Stefanidis, K., Koutrika, G., Pitoura, E.: A survey on representation, composition and application of preferences in database systems. ACM Trans. Database Syst. 36, 19:1–19:45 (2011)
24. Lacroix, M., Lavency, P.: Preferences: Putting more knowledge into queries. In: Proc. of the 13th Inter. Conference on Very Large Databases (VLDB 1987), pp. 217–225 (1987)
25. Bosc, P., Pivert, O.: Fuzzy Preference Queries to Relational Databases. Imperial College Press (2012)