Preference Networks for the Combination of Qualitative and Quantitative Preferences

Panos Vassiliadis
Dept. of Computer Science
University of Ioannina
Ioannina 45110, Hellas
pvassil@cs.uoi.gr, www.cs.uoi.gr/~pvassil

March 28, 2010

WORK IN PROGRESS

Abstract

Preference management in databases is handled via profiles that allow the ranking of the tuples of a query's answer. A profile is a set of expressions, typically over the attributes of a relation. Preference management and profiles typically come in two flavors: (a) *quantitative* preferences, that assign a score to a profile's expressions and subsequently to each tuple that satisfies an expression, and (b) *qualitative* preferences which state a precedence relationship between two expressions and subsequently result in expressing when a tuple is preferred over another. Related literature has elaborated each of the above categories in isolation only. In this paper, we present a method to map quantitative to qualitative preferences and vice versa. We map profiles of both kinds to graphs called preference networks that formally capture the precedence relationship of a profile's expressions for both of the above categories. To fully support the management of preferences via preference networks we introduce weighted qualitative preferences that annotate the precedence relationship with a degree of preference (i.e., be able to say 'I prefer `IM` to `JP` *a lot*'). We provide methods to map qualitative to quantitative profiles as well as different ways to consolidate two profiles in one.

1 Introduction

A preference-based querying system receives the query of a user over a relation R and returns the results of the query ranked on the basis of a set of preferences that have previously been expressed by the user. This set of preferences is also referred to as the profile of the user. The research community so far has followed two paths in the research of preference-based querying systems. The first class of approaches is based on *quantitative* preferences, in the context of which a user annotates an item with a score that characterizes the degree of a preference. *Qualitative* preferences, on the other hand, are the basis for the second class of approaches; qualitative preferences are expressed via the comparison of items. Assume a relation where songs composed by different bands are stored. In a quantitative preference setting, a user might for example declare in his profile 'I like songs composed by `Saxon` with a score of 0.7'. Comparison of items with respect to a user's profile of preferences is done via the scores. For example, if the user also declares 'I like `Savatage` with a score of 0.8', then the system will rank all songs composed by `Savatage` higher than the ones composed by `Saxon`. In the context of a qualitative setting, the user might, for example, declare 'I prefer `Iron Maiden` to `Metallica`' and then, all songs composed by `Iron Maiden` will be scored higher than the ones composed by `Metallica`.

Related work in database querying has pursued both paths for the management of preferences. On the one hand, there are efforts like [Cho03], [KK02], [Kie02], [GKC^{+}08] who base their approach on qualitative relationships. On the other hand, there are efforts like [KI05b], [KI05a],[SPV07] who follow a quantitative approach. Still, to the best of our knowledge, there is no work that combines both worlds in a clear way.

Clearly, it is very interesting to come up with a way to combine both methods. First, this can give us a generic way to abstract both quantitative and qualitative preferences in a common formalism. Second, this allows the expression of more complicated profiles, where both qualitative and quantitative preferences coexist. Third, this allows the integration of heterogeneous profiles (possibly from different systems). Finally, more complicated operations can also be supported: for example, when a person rates an album that could be a quantitative mode, whereas when she clicks on some search results that could be translated in a qualitative way.

In this paper, we present a method to map quantitative to qualitative preferences and vice versa as well as the means to consolidate two profiles in one. We map profiles of both kinds to graphs called preference networks that formally capture the precedence relationship of a profile's expressions for both of the above categories. To fully support the management of preferences via preference networks we introduce weighted qualitative preferences that annotate the precedence relationship with a degree of preference. *So, instead of expressing preferences of the form 'I prefer* `Iron Maiden` *to* `Metallica`', *we are interested in expressing preferences of the form 'I prefer* `Iron Maiden` *to* `Metallica` **a lot**', *or, 'I prefer* `Savatage` *to* `Accept` **slightly**'. In this paper, we build upon this kind of preferences and provide the following results.

- We identify the different domains that can exist for scores and preference weights. Different types of score allow different ranges of actions, like for example difference or division of scores. Having clearly set the foundations for the nature of the domain equips us with the knowledge of the potential of the specified profiles.

- We formalize the definition of profiles and map profiles to graphs, which we call *preference networks*. This graph representation is essential for the management of preferences in the sequel.

- We formalize *weighted qualitative preferences* and explore their properties. As already mentioned, weighted qualitative preferences capture not only preferences, but degrees of preference, too. A weight annotates a preference expression and allows us to infer further properties of the interrelationships between the preferences within a profile. We explore properties and possible operations for weighted qualitative preferences and spend significant attention to the case of transitive derivation of such interrelationships.

- We provide methods to map qualitative to quantitative profiles and vice versa and explore the properties and consistency requirements for this kind of transformations.

- We provide different types of profile consolidation between a qualitative and quantitative profile. We introduce *naive consolidation* to allow us to set the basis for the subsequent discussion and define consistency checks and well-formedness properties for profiles that will be consolidated. We also provide *shallow* consolidation that practically extends a profile with information from the other profile. Finally, we present *deep consolidation* during which the two profiles are intermingled into a new one.

- Due to the indefinite nature of some types of weight domains, it is possible to introduce some degree of uncertainty in the reasoning about the interrelationship of two preferences (typically, when this is derived via some transitivity rule). As a side-effect of the main effort of this paper, we introduce range-belief preferences and explore their properties, too.

The roadmap for the rest of the paper is as follows. In section 2 we introduce the various kinds of domains that can be used for scores and weights. In section 3 we formalize profile definition and give some reference examples; also, we introduce preference networks. In section 4 we introduce quantitative preferences and in section 5 we introduce weighted qualitative preferences; moreover we discuss transitivity and range-belief preferences. In section 6 we explore operations for these kinds of preferences. In section 7 we present a first method to consolidate qualitative and quantitative profiles and discuss properties and consistency considerations for the consolidated profiles. In section 8 we map quantitative to qualitative domains and vice versa. In section 9 we present a shallow method of consolidating profiles that respects the individual nature of the original profiles. Finally, in section 10 we explore the deep consolidation of quantitative and qualitative profiles.

2 Domains and characteristics for degrees of preferences

As already mentioned in the introduction, quantitative preference expressions are annotated with a degree of preference, which we call score. We also plan to use degrees of preference when we introduce weighted qualitative preferences later in this paper. Before proceeding with the formal definition of such preferences, we need to take a deeper look to the fundamental nature of the characteristics of these degrees of preferences; this is the topic of this section.

Measurement theory typically divides measurement domains on the basis of the operations they support. Specifically, the following domains are valid in measurement theory [FP97]:

1. *Nominal scale domains.* These domains simply list constants that convey a certain meaning without any sense of order or applicable operations on them. For example, such a domain concerning software faults [FP97] could be $\{design\ fault, specification\ fault, code\ fault\}$. We can only test the members of nominal scale domains for equality or not($=, \neq$).

2. *Ordinal scale domains.* These domains support a notion of ordering for their members, but no other applicable operations on them. For example, such a domain concerning temperatures could be $\{cold, mild, hot\}$: there is a strict order of these values, but we cannot define distances or arithmetic operations to them. We can test the members of ordinal scale domains for (i) equality ($=, \neq$), and (ii) higher position in the domain's order ($<, >, \leq, \geq$).

3. *Interval scale domains.* These domains order the constants that are members of the domain and capture information for the distances among them. Therefore, addition and subtraction can be defined for them (although multiplication and division cannot). For example, a Celsius scale for temperature is an interval scale: we can say that two temperatures, say 10C and 20C, have a distance of 10 degrees, but not that the latter is twice as hot as the former. We can, however, say that the distance of 10C to 5C is half the distance of 10C to 20C. We can test the members of interval scale domains for (i) equality ($=, \neq$), (ii) higher position in the domain's order ($<, >, \leq, \geq$), (iii) division of their intervals (but not of their values per se).

4. *Ratio scale domains.* These domains are characterized by an ordering of their members, the intervals among them, and, ratios among them. Moreover, they have a zero element (meaning absolute absence of the measured quality) which is the first member of the domain. The

domain increases at regular intervals (a.k.a. units) and supports all arithmetic operations. For example, *Lines of Code* constitute a measurement domain in the ratio scale. We can test the members of ratio scale domains for (i) equality ($=, \neq$), (ii) higher position in the domain's order ($<, >, \leq, \geq$), (iii) division of their intervals, (iv) division of their values per se.

5. *Absolute scale domains.* These domains are characterized by the fact that the only possible members of the domain are countings of elements of a set under measurement (e.g., 'how many times a bug was inspected in module X'). Any arithmetic operations to the results of the counting is applicable.

In the sequel we will provide the formal foundation for degrees of preferences for any kind of degrees (be it quantitative scores, or weights for weighed qualitative preferences), on the basis of the aforementioned taxonomy. Formally, assume a domain of degrees of preference $\mathbf{W}^*$. Unless otherwise specified, in the rest of our deliberations we will by default assume that (a) $\mathbf{W}^*$ is infinitely countable and (b) $\mathbf{W}^*$ is at least in the ordinal scale and, therefore, its members respect a strict total order (i.e., $\forall\, x, y$, one of the three holds $x > y, y > x, x = y$). In special occasions where specific assumptions must be made for the scale type of $\mathbf{W}^*$, these will be explicitly listed.

For the special cases when $\mathbf{W}^*$ is *finite*, it is clear that due to the strict partial order of its members there are two special values: (a) the one with the smallest rank, denoted as $\perp$ (i.e., $\forall\, w \in \mathbf{W}^*$, $w > \perp$) and (b) the one with the highest rank, denoted as $\top$ (i.e., $\forall\, w \in \mathbf{W}^*$, $w < \top$).

We also assume a "zero-knowledge element", w_0. In the context of weighted qualitative preferences, this element has the semantics of "I prefer t_1 to t_2, but I do not know the weight of my preference". We will use the term $\mathbf{W}$ to denote $\mathbf{W}^* \bigcup \{w_0\}$. A very important discussion concerns the place of w_0 in the total order of $\mathbf{W}$. We *do not* include w_0 in the total order of $\mathbf{W}^*$. This requires the redefinition of several properties concerning the operators $\{=, \neq, <, >, \leq, \geq\}$ since, given a weight w, the result of comparing it to w_0 is *unknown*. We require the following:

- All comparisons and binary arithmetic operations among two members of $\mathbf{W}^*$ retain their value in $\mathbf{W}$

- Assuming $w \in \mathbf{W}$ the comparisons $w\ \vartheta\ w_0$, $\vartheta \in \{=, \neq, <, >, \leq, \geq\}$, result in the special value `UNKNOWN`.Observe that w could be w_0, too.

- Assuming $w \in \mathbf{W}$ the comparisons $w\ \varphi\ w_0$, $\varphi \in \{+, -, *, /\}$, also result in the special value `UNKNOWN`. Observe that w could be w_0, too.

Practically, we treat the value w_0 as the *unk* value of [GZ88].

Clearly, if *all* preferences in a preference relation are defined with w_0 as the employed weight we fall back to the traditional models of qualitative preferences [Cho03], [Kie02].

3 User profiles

A user's profile is a finite set of preferences. In this section, we discuss a simple formalism for expressing user preferences. Then, we move on to formally define profiles and then we map them to preference networks. Finally, we cover issues of tuple ranking with respect to a profile.

Several proposals for the specification of user profiles exist in the literature.In [KK02], Kießling and Köstler introduce Preference SQL, an SQL extension aimed towards supporting the expression of user preferences in conjunction with regular SQL queries. Preference SQL supports the user in expressing simple preferences that prioritize tuples that satisfy one of the following conditions: an attribute is *around* a certain value, or *between* two values, or has the *highest/lowest* value in the relation. Moreover, the user can declare his positive or negative preference for specific values of an attributes. Finally, Preference SQL supports the user in expressing composite preferences via (a) *Pareto composition* (which treats all simple preferences as equal), or, (b) *cascading composition* (which means that the preference expressions are ordered, each having a decreasing priority, and the order of the tuples returned to the user reflects this order of preferences). The work of Chomicki [Cho03], assumes that preferences are expressed as first order formulae; a formula f is in disjunctive normal form, quantifier-free, closed under negation and uses only built-ins.

In our case, we do not assume a particular way via which user preferences are collected (e.g., submitted by the users via a GUI, or collected from a reasoner that deduces there preferences from user transactions), nor do we intend to introduce a specific language for that purpose. Our main concern in this section is to present a formalism for simple expressions that allows formulae to be evaluated on the tuples of an underlying relation R in $o(1)$ (i.e., we determine whether a preference is relevant for a tuple in isolation with reference only to the attribute values of the tuple).

3.1 Preference formulation

We will start with the introduction of general purpose (i.e, applicable in both the qualitative and the quantitative context) preference expressions over the tuples of a relation. Then, we will see how quantitative and qualitative preferences can be expressed via preference expressions.

In the sequel, we will assume an infinite domain of names and a single relation R whose schema $R.S$ comprises attributes $A_1, \ldots A_n$. Each attribute A_i is related to a domain $dom(A_i)$ which is isomorphic to the integers. All expressions are defined over R; that said we avoid repeating the term 'over R' in every definition. We assume a domain $\mathbf{W}_s$ for the scores of quantitative preferences and a domain $\mathbf{W}_w$ for the weights of qualitative preferences.

Definition 1 *(Atoms). An atomic expression, or atom, is an expression of the form $A\ \theta\ v$, with A being an attribute of R, v being a value in $dom(A)$ and θ is an operator in the set $(<, >, \leq, \geq, =, \neq)$.*

Definition 2 *(Expressions). An expression in conjunctive normal form (CNF) involves the conjunctions of a finite set of atoms over R.*

In our setting, we define quantitative preferences as formulae in conjunctive normal form that are annotated with a strength and qualitative preference expressions as pairs of formulae in conjunctive normal form that are connected with a preference weight. Observe that we use a different term for the degrees of preference, depending on their quantitative or qualitative nature: this is done for reasons of clarity of the context. The respective profiles are defined as finite sets of preferences.

Definition 3 *(Quantitative Preference expression). A quantitative preference expression is a pair $p_s(\phi,\ s)$, with ϕ being an expression and $s \in \mathbf{W}_s$.*

Definition 4 *(Qualitative Preference expression). A qualitative preference expression is a triplet $p_w(\phi_1, \phi_2,\ w)$, with ϕ_1 and ϕ_2 being expressions and $w \in \mathbf{W}_w$.*

Definition 5 *(Profile). A* profile *P is a finite set of qualitative and quantitative preference expressions expressed over the same relation R.*

In other words, a profile is a finite set of expressions of the following two kinds:

1. (qualitative) $\bigwedge_{i\ \in\ 1\ldots\ N} (A_i\ \theta_i\ v_i) \succ^w \bigwedge_{j\ \in\ 1\ldots\ M} (A_j\ \theta_j\ v_j)$,
2. (quantitative) $\bigwedge_{i\ \in\ 1\ldots\ N} (A_i\ \theta_i\ v_i)$ with strength s, for a finite N

with N,M denoting positive integers, A denoting attributes of relation R, $\theta \in (<, >, \leq, \geq, =, \neq)$ and v denoting a value in $dom(A)$.

We will call the former part of the profile as its *qualitative part* and the latter as its *quantitative part*. A profile consisting only of qualitative part is called a *qualitative profile*; similarly, a profile consisting only of a quantitative part is called a *quantitative profile*.

It is possible to extend the representation of formulae (both quantitative and qualitative) and include disjunctions in the same expression. In other words, we could also consider preferences in disjunctive normal form. Still, we opt for a simpler class of expressions, bearing in mind that if a user wishes to express a preference that comprises a disjunction, the disjunction appears as two different preferences, with the rest of the preference expression intact; this can be recursively generalized for arbitrary disjunctions.

The problems that we have to deal with now are (a) the precise definition of the semantics of the above language constructs (and their implications) and (b) the combination of quantitative and qualitative preferences in a unique 'construct' that will allow us to order the tuples of the relation R in a unique way. So, in the next two sections we discuss quantitative and qualitative preferences in isolation and then, in section 5 we move on to unify profiles that comprise preferences of both kinds.

3.2 Reference Example

We will employ an exemplary database of music albums with data taken from Wikipedia. Each album is characterized by the band that produced it, its date of release, its duration and the musical genre to which its belongs. All the preferences and profiles that will be introduced later in this section and throughout this paper refer to this reference relation of albums.

Band	Album	Date	Duration	Genre
Iron Maiden	*Piece Of Mind*	1983	45:50	Heavy
	Fear of the Dark	1992	58:29	Heavy
	A Matter Of Life And Death	2006	71:54	Heavy
Savatage	*Hall Of The Mountain King*	1987	39:29	Power, progressive
	Dead Winter Dead	1995	52:06	Progressive, symphonic
Gamma Ray	*Land Of The Free*	1995	56:43	Power
Iced Earth	*Iced Earth*	1991	44:14	Thrash
	Something Wicked This Way Comes	1998	61:56	Power, heavy
	Alive in Athens	1999	180:06	Thrash, Power
	The Reckoning	2003	17:41	Thrash, power
Accept	*Restless and Wild*	1983	43:52	Heavy, speed
	Russian Roulette	1986	43:21	Heavy
Helloween	*Helloween*	1985	26:22	Speed
	Walls Of Jericho	1985	40:32	Power, speed
Saxon	*Crusader*	1983	39:10	Heavy
	The Inner Sanctum	2007	47:48	Heavy
Stratovarius	*Fright night*	1989	40:33	Power
	Elements part I	2002	61:56	Power, neo-classical
Metallica	*Kill 'em all*	1983	51:!3	Trash, speed
	Ride the lightning	1984	47:30	Thrash
	Load	1996	78:59	Heavy, hard rock
	S & M	1999	131:40	Heavy

Figure 1: A database of heavy metal albums

3.3 Preference Networks

In this section, we will map qualitative and quantitative profiles to directed graphs which we call *preference networks*. We will discuss the notion of the *covers* relationship for both cases. The proposed graphs are practically the Hasse diagrams of the profiles at the intentional level.

Assume a relation R whose schema $R.S$ comprises attributes $A_1, \ldots A_n$. All preference expressions for all the profiles we will discuss are defined over R. Assume a domain $\mathbf{W}_s$ for the scores of quantitative preferences and a domain $\mathbf{W}_w$ for the weights of qualitative preferences.

Definition 6 *(Qualitative Preference Network). An* intentional qualitative preference network L_I^w *of a qualitative profile* P *over a relation* R *is a graph* $G(V, E)$ *constructed as follows:*

1. *Each expression participating in any kind of preference is mapped to a node* $v \in V$*. If an expression appears more than once, exactly one node is used.*
2. *Given two expressions* ϕ_1 *and* ϕ_2*, a directed edge* $(\phi_2 \rightarrow \phi_1)$ *connects the nodes of the two expressions with a weight* w *if* ϕ_1 covers ϕ_2*. The* covers *relationship is defined as follows:*
   - *Assuming two qualitative preference expressions* ϕ_1 *and* ϕ_2*,* ϕ_1 covers ϕ_2 *if and only if* $\phi_1 \succ^w \phi_2$ *and* $\nexists$ *expression* ϕ_3 *such that* $\phi_1 \succ \phi_3 \succ \phi_2$*. In this case, a directed edge* (ϕ_2, ϕ_1) *is added to the graph and it is annotated with* w*.*

Example. Ingo has the qualitative profile depicted in Figure 2. Ingo is a sophisticated person with a complicated profile: in summary, Ingo likes `Gamma Ray` more than any other band and then his preferences are divided to the band `Helloween` and the genre *speed metal* that both follow `Gamma Ray` with a difference *little*. In terms of genres, Ingo prefers *speedmetal* to *heavy and speed*, which in turn, is preferred to *heavy metal* with a weight *little*. At the same time, `Helloween` is preferred *a lot* to both *thrash metal* and very long albums with duration more than 45 minutes. *Thrash metal* is also preferred to *thrash and speed* which in turn is preferred *a lot* to old songs with $date < 1980$. The latter are also surpassed by the preference on long albums by *little*. Old albums are *little* preferred to any albums of genre *heavy metal*.

Ingo's preferences can be formally stated as follows:

- $Band = Gamma\ Ray \succ^{little} Genre = speed \succ Genre = heavy\&speed \succ^{little} Genre = heavy$
- $Band = Gamma\ Ray \succ^{little} Band = Helloween \succ^{a\ lot} duration > 45 \succ^{little} Date < 1980 \succ^{little} Genre = heavy$

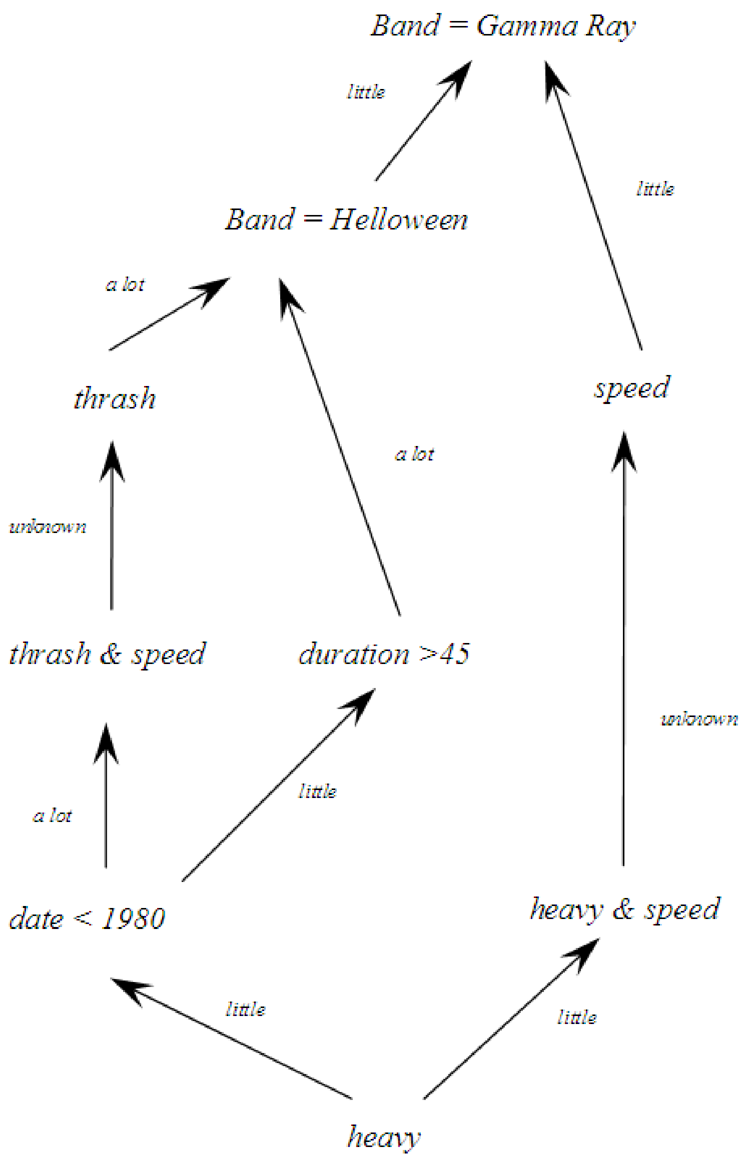

Figure 2: Ingo's profile

- $Band = Helloween \succ^{a\ lot}\ Genre = thrash \succ Genre = thrash\&speed \succ^{a\ lot}\ Date < 1980$

Definition 7 *(Quantitative Preference Network). An* intentional quantitative preference network L_I^s *of a quantitative profile* P *over a relation* R *is a graph* $G(V, E)$ *constructed as follows:*

1. *Each expression participating in any kind of preference is mapped to a node* $v \in V$*. If an expression appears more than once, exactly one*

node is used.

2. *The strength of a quantitative preference annotates the node of the respective expression.*

3. *Given two expressions ϕ_1 and ϕ_2, a directed edge ($\phi_2 \rightarrow \phi_1$) connects the nodes of the two expressions if ϕ_1* covers *ϕ_2. The* covers *relationship is defined as follows:*

    - *Assuming two quantitative preference expressions (ϕ_1,s_1) and (ϕ_2,s_2), ϕ_1* covers *ϕ_2 if and only if $s_1 > s_2$ and $\nexists$ expression ϕ_3, s_3, such that $s_1 \geq s_3 > s_2$.In this case, a directed edge (ϕ_2, ϕ_1) is added to the graph.*

Example. Assume John has a quantitative profile defined over the albums relation of the reference example with the domain of scores having the following 10 values with the obvious semantics $\mathbf{W}_s = \{0.1, \ \ldots \ , 1.0\}$. John's preferences are: [1]

- John likes albums with *Date* $\geq$ 1990 and *Date* $\leq$ 2000 with a score of 0.6
- John likes albums with *Date* $<$ 1990 with a score of 0.7
- John likes albums with *Date* $>$ 2000 with a score of 0.7

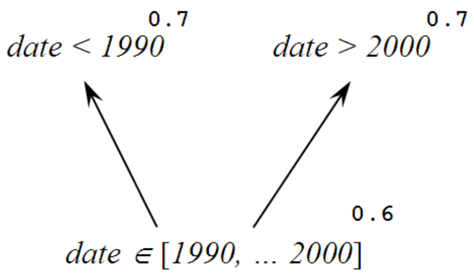

Figure 3: John's profile

3.4 Tuple classification over a profile

How are the tuples of a relation R related to the preferences of a profile P? It is possible that more than one preferences are related to a single tuple. The relationship of tuples to preferences is covered by the following definition:

[1] We intentionally restrict preferences over the Date attribute. This is not a limitation of the model but a choice to have an example where each tuple can be related to a single preference expression; as we shall see in the sequel, this is not always the case with profiles.

Definition 8 *(Extent of a preference) Each expression (also: node of a network) is related to a (possibly empty) subset of relation R (which we call the extent of the preference) that consists of the tuples that satisfy the expression. We denote the extent of a preference p with $Ext(p)$.*

We consider two types for the classification of tuples to preference extents:

- A *monothetic tuple classification scheme* assigns each tuple to at most one preference extent
- A *polythetic tuple classification scheme* allows a tuple to be related to more than one preference extents

A quantitative profile is *intentionally monothetic* (or simply, *monothetic*) with R if a single score (no score included) is assigned to each tuple of the relation R, independently of the instance of R. If this property holds but it is subject to the current instance of R, then P is *extensionally monothetic* with R. The intuition behind this separation has to do with the ability to determine a-priori whether there is a possibility for a tuple to be related with more than one preferences. In the case of intentional monothecy, the profile guarantees this property independently of the underlying extent of R (for example, the profile involves exactly one attribute and the ranges of the preference expressions are disjoint). Still, it is quite easy to have situations where no consistency can be attained.

4 Quantitative Preferences

In this section we will discuss quantitative preferences in isolation and investigate their properties.

4.1 Intentional level

As already mentioned, quantitative preferences express do not assess different tuples with respect to which is most preferred over the other, but rather they express the user's preference over a single tuple. Scores indicate the degree of preference (for example "I **like** `Iron Maiden` very much". Observe the usage of the verb (predicate) "like" as opposed to the previously used verb "prefer".

Assume a relation R (over which preferences will be expressed) and a domain $\mathbf{W}_s$ (which will serve as the domain of preference scores) as described in section 2. Remember that $\mathbf{W}_s$ is infinitely countable (frequently, finite) and at least in the ordinal scale. Assume a profile P comprising only quantitative preferences, too.

A set of quantitative preferences defined over the relation R divides the extent of R to subsets of tuples annotated by the scores of the preferences.

Assume a finite set of quantitative preferences p_1, p_2, ..., p_m all defined over R. Each preference is of the form $p_i(\phi_i, s_i)$, involving an expression ϕ_i and a score s_i. The scores of the preferences impose an *ordering* to the preferences, i.e., assuming two preferences p_i and p_j, $p_i \neq p_j$,

$p_i \succ p_j \Leftrightarrow w_i > w_j$

Example. Let us come back to the example depicted in Figure 3. Observe that there is partial order of John's preferences (and consequently of their extents). Observe also that the fact that all preferences are defined over the same attribute combined with the fact that the ranges of values for the preferences are disjoint guarantees that each tuple of the relation will fall in exactly one preference extent.

4.2 Consistency considerations

What happens, if we cannot guarantee any kind of monothetic classification for a quantitative profile over a relation? For example, assume that a user submits the following preferences:

- I like traditional metal albums ($Genre{=}'Heavy'$) with score 0.8
- I like albums of the early 80's ($Date \leq 1985$) with score 0.9

Clearly, since the two preferences are orthogonal to each other (i.e., expressed over different attributes) it is quite possible that an album satisfies both of them – in fact, the album $Piece\ of\ Mind$ (first line of Figure 1) satisfies both criteria. There are various reactions to this situation, out of which we retain the two most practical:

Preference prioritization. The first possible reaction is to prioritize preferences in a total order. In this case, if a tuple falls in the extent of more than one preferences, it will assume the score of the one with the highest priority. This is what qualitative preferences do with lexicographic composition of preferences [Cho03] (also referred to as prioritized composition [Kie02]).

Score combination. The second possible reaction is to combine the scores of the individual preferences which are related to a tuple into a single score via a function (see for example, [SPV07]); examples include a weighted sum, or the maximum value of all the scores.

4.3 Extensional level

The mapping of a quantitative expression over a relation R to a quantitative relationship is straightforward. Assume a quantitative preference $p(\phi, s)$

over R, involving an expression ϕ and a score s[2]. Then, for all tuples $t \in R$, if $\phi(t)$ then t in $Ext(p)$. We say that the *score of a tuple* t, $score(t)$, belonging to the extent of a preference $p(\phi, s)$ is s.

The real problem now is whether the preference extents are disjoint, i.e., we should be able to guarantee that a profile assigns each tuple to the extent of a single preference. If such a property is obtained, then the profile produces equivalence classes, determined on the basis of the user preferences, that provide a unique ordering of the tuples of a relation R.

Proposition 4.1 *A quantitative profile that is not accompanied by a conflict resolution mechanism does not guarantee equivalence classes for the tuples of the relation over which the profile is defined.*

An example for the above proposition is mentioned in the previous subsection.

Theoretically speaking, a solution that alleviates the problem is to compute all the 2^n combinations of the different preferences (possibly excluding the ones that naturally result in $false$) as well as the appropriate score for each of the combinations. Assume a finite set of atomic preferences $p_1, p_2, \ldots, p_n$. Without loss of generality assume also the composite preference $p = p_1 \ \wedge \ p_2$. A tuple t belongs in the extent of p if both $p_1(t)$ and $p_2(t)$ hold, and at the same time, $p_i(t)$ does not hold for every other $i \ \neq \ 1, 2$. In this case it is clear that every tuple belongs to exactly one of these composite preferences and the extended profile that comprises the 2^n combinations produces an equivalence relation over the tuples.

Definition 9 *Given a profile P over a relation R comprising a finite set of quantitative preferences, P=$\{p_1, p_2, \ldots, p_n\}$, a* composite preference *p is a combination of a subset of the preferences of the profile P. A tuple t belongs to the extent of p if and only if (i) all the expressions of the preferences of p hold for this tuple and (ii) there is no preference in P - p for which t satisfies its expression. The score for the new composite quantitative preference is a function of the scores of the atomic preferences that belong to p.*

Definition 10 *Given a profile P over a relation R comprising a finite set of quantitative preferences, P=$\{p_1, p_2, \ldots, p_n\}$, an extended profile P^+ is the set of all the 2^n composite preferences of P.*

It is clear given a profile composed of a finite set of preferences, an extended profile can always be built. In this case, even if a tuple would originally be related to more than one preferences, it will eventually resort in a single composite preference and obtain the appropriate score given from the strict prioritization or the score combination of the simple preferences.

[2]We remind the reader that all the notation and definitions are found in Section sec:profiles

Theorem 4.2 *Both preference prioritization and score combination guarantee that a profile P acts as an equivalence relation for the tuples of a relation R.*

5 Weighted Qualitative Preference Relations

Having defined preliminary concepts concerning the domains of scores and preference weights, now, we can define weighted preference relations. A weighted qualitative preference relation states that a tuple t_1 is *strictly* preferred over another tuple t_2 (as opposed to the possibility of being *at least* preferred to t_2 – see [GKC+08] for an interesting discussion on the topic). Moreover, this preference has a weight taken from a domain of at least ordinal nature and the preference states that t_1 is preferred over t_2 with an *exact* weight w (as opposed to the possibility of *at least* w, or, *between the range of* w_1 *and* w_2).

Definition 11 *(Weighted Qualitative Preference Relation). A relation $\mathcal{P}$ is a weighted qualitative preference relation over a database relation R if it is a subset of $(dom(A_1) \times \ldots \times dom(A_k)) \times (dom(A_1) \times \ldots \times dom(A_k)) \times \mathbf{W}_w$, with all $(dom(A_i)$ being isomorphic to the integers and $\mathbf{W}_w$ being of at least ordinal scale.*

For two arbitrary tuples of R, say t_1 and t_2 and an arbitrary weight w, if $\mathcal{P}(t_1, t_2, w)$ holds, then we write $t_1 \succ_{\mathcal{P}}^{w} t_2$ and read it as "t_1 is ***strictly*** *preferred over t_2 with a weight of* ***exactly*** *w".*

When the context is clear, we will simplify notation and write $t_1 \succ^w t_2$ only. Following Chomicki [Cho03], we say that t_1 *dominates* t_2 by w. As usual, whenever $\nexists\ w \in \mathbf{W}_w$ such that $t_1 \succ^w t_2$, we denote it as $t_1 \nsucc^w t_2$.

The example of Figure 2 graphically depicts a qualitative profile as a preference network.

5.1 Indifference

It is also possible that two items are not comparable at all. For example, if the only knowledge of my preference profile is that I prefer `Deep Purple` more than `Led Zeppelin` with weight *fair*, this tells me nothing about the relationship of these items with `Metallica` (or any other member of the domain of the attribute *Band*).

Definition 12 *(Indifference). Formally, two items are indifferent to each other, denoted as $t_1 \ \| \ t_2$ if neither is preferred over the other:*
$t_1 \ \| \ t_2 \Leftrightarrow \forall w \in \mathbf{W}_w, t_1 \nsucc^w t_2 \bigwedge t_2 \nsucc^w t_1$

5.2 Properties of weighted qualitative preference relations

In this subsection, we will explore properties of weighted preference relations. Following the structure of [Cho03] in our deliberations, we will discuss possible properties that can hold for a preference relation. In our deliberations in this section, we will assume a weighted qualitative preference relation $\mathcal{P}$ defined over a database relation R.

Definition 13 *(irreflexivity). For each* $t \in R$, $t \not\succ^{w} t$, *for every* $w \in \mathbf{W}_w$

In other words, an item cannot be preferred to itself, for *any* possible weights.

Definition 14 *(antisymmetry). For each* t_1,$t_2 \in R$, *and* w_1,$w_2 \in \mathbf{W}_w$ *the following holds:*

$t_1 \succ^{w_1} t_2 \Rightarrow t_2 \not\succ^{w_2} t_1$

Observe that this property is defined for any weights w_1, w_2.

Definition 15 *(negative transitivity). For each* $t_1, t_2, t_3 \in R$, *and,* w_1,w_2 $\in \mathbf{W}_w$ *the following holds:*

$t_1 \not\succ^{w_1} t_2$ *and* $t_2 \not\succ^{w_2} t_3 \Rightarrow \nexists\ w$ *in* $\mathbf{W}_w$ *s.t.* $t_1 \succ^{w} t_3$

In other words, negative transitivity gives guarantees in the absence of preference: 'if t_1 is not preferred to t_2 and t_2 not preferred to t_3, then how could t_1 be possibly preferred to t_3?' (or, else, if t_1 was preferred to t_3, wouldn't it be reasonable that at least one of the other relations holds?).

Definition 16 *(chain). For each* t_1,t_2, *exactly one of the following holds:*

- $t_1 \succ^{w} t_2$ *for some* $w \in \mathbf{W}_w$, *or,*
- $t_2 \succ^{w} t_1$ *for some* $w \in \mathbf{W}_w$, *or,*
- $t_1 = t_2$

In other words, when the chain property holds, it is impossible to have two items t_1, t_2 that are not comparable with each other (i.e., $t_1 \parallel t_2$).

Remember that the above are *possible* properties and the following combinations of properties give characterizations for the preference relation:

- strict partial order: irreflexive and transitive

- weak order: strict partial order and negative transitivity
- total order: strict partial order and chain

Clearly, transitivity is a central issue in the characterization of the nature of weighted preference relations and needs to be explored in detail. The main problems with transitivity are due to the fact that we need to compute a weight for the transitively deduced relation; this is not always straightforward, due to the delicate nature of the type of $\mathbf{W}_w^*$.

5.3 Transitivity and its Implications

The main idea in transitivity has to do with the nature of the domain. If `Rainbow` is preferred to `Deep Purple` with weight *low* and `Deep Purple` to `Led Zeppelin` with weight *fair*, what can we deduce for the relationship of `Rainbow` and `Led Zeppelin`, and its weight? One possibility for the answer is *fair*, i.e., the maximum of the two weights. Still, this is a matter of taste; in fact it is also quite possible to say *a little bit more than fair*. Assuming that the natural choice is max, though, allows to reflect again on the possibility of defining preferences with an 'at-least' modality in the definition of the preference relation (as opposed to 'exact') for the preference's weight (I prefer `Rainbow` at least with a weight of *fair* over `Led Zeppelin`): as we shall see, we can do even better; to this end, we introduce range-belief preferences in the following section.

The situation is possibly different if the domain of $\mathbf{W}_w$ allows us to define difference over the weights. So, if instead of *low*, the weight is 0.3 and instead of *fair*, the weight is 0.5, can we assume that `Rainbow` are preferred to `Led Zeppelin` with a score of 0.8? What if the two scores were 0.8 and 0.9 and the upper limit is 1.0?

In the sequel, we will explore alternative implications for transitivity and discuss their semantics and properties. In our deliberations we need to enrich the domain of weights with a function that assigns the score of the transitively computed relationship. Formally, we assume a computable function f_{sit}, f_{sit}: $\mathbf{W}_w \times \mathbf{W}_w \rightarrow \mathbf{W}_w$ that takes the original weights as input and produces the new one as output.

Definition 17 *(Simple Item Transitivity). For each* t_1,t_2,$t_3 \in R$, *and* w_1,w_2 $\in \mathbf{W}_w$ *the following holds:*

$$t_1 \succ^{w_1} t_2 \text{ and } t_2 \succ^{w_2} t_3 \Rightarrow t_1 \succ^{w} t_3,\ w = f_{sit}(w_1, w_2),\ w \in \mathbf{W}_w$$

where f_{sit} *is a computable function* f_{sit}: $\mathbf{W}_w \times \mathbf{W}_w \rightarrow \mathbf{W}_w$.

For ordinal domains supporting only order among their members, function f_{sit} could be *min*, *max*, a constant (e.g., the 'I don't know the score' constant) or any other applicable function. In case $\mathbf{W}_w$ is of ratio scale and it supports difference, then a variant of addition can be defined as long as it is closed for the domain $\mathbf{W}_w$ (which is straightforward for domains isomorphic to the integers, but requires special care for finite domains).

A possible example of function f_{sit} for an ordinal domain is the following:

$$w = \begin{cases} w_1 & if\ w_2 = w_0 \\ w_2 & if\ w_1 = w_0 \\ max(w_1, w_2) & otherwise \end{cases} \tag{1}$$

A second example for function f_{sit} is $w = w_0$ for any w_1, w_2, projecting the computed transitive relationship of two tuples into a weightless qualitative relationship.

A third example of function f_{sit} for an interval scale domain is the following:

$$w = \begin{cases} w_1 & if\ w_2 = w_0 \\ w_2 & if\ w_1 = w_0 \\ max(w_1, w_2) + |w_1 - w_2| & if\ this\ sum\ doesn't\ exceed\ \top \\ \top & otherwise \end{cases} \tag{2}$$

If the domain is infinitely countable, the last case is useless; on the other hand, if the domain is finite, it is important to guarantee that f_{sit} is closed for $\mathbf{W}_w$.

Another possible consideration has to do with the case of two values that are both preferred over a third one. What is the relationship of the two first values, then?

Definition 18 *(Simple Weight Transitivity). For each t_1,t_2,t_3 $\in R$, and w_a,w_b $\in \mathbf{W}_w$ the following holds:*

$t_1 \succ^{w_a} t_2$ *and* $t_1 \succ^{w_b} t_3$, $w_b > w_a \Rightarrow t_2 \succ^{w} t_3$, $w = f_w(w_a, w_b)$, $w \in \mathbf{W}_w$

where $f_w$is a computable function f_w: $\mathbf{W}_w \times \mathbf{W}_w \to \mathbf{W}_w$.

Observe that the implication holds only when $w_b > w_a$ and therefore, the cases where w_0 could possibly be involved are excluded. Simple weight transitivity can be defined both for domains in any of the ordinal, ratio and interval scales. Still, it is much more interesting when weight differences can be defined – i.e., when the domain of $\mathbf{W}_w$ is of ratio or interval scale. Assume I prefer `Savatage` to (a) `Saxon` with a score of 0.3 and (b) `AC/DC` with a score of 0.7. Then, if simple weight transitivity is supported, we can infer that I prefer `Saxon` to `AC/DC` with a score of 0.4, assuming $f_w = w_b - w_a$.

5.4 Semantics and Consistency Considerations

So far we have covered how preferences can be expressed as well as what their properties are. So, when a user poses a query over the relation R, how are the results going to be ranked according to the user's profile P? To deal with this problem we need to relate the user's preferences to the ranking of tuples in the database.

The mapping of a qualitative expression over a relation R to a qualitative relation is straightforward. Assume a qualitative preference expression $p(\phi_1, \phi_2, w)$ over R. Then, for all tuples t_1, $t_2 \in R$, $t_1 \neq t_2$, if

(i) $\phi_1(t_1)$ and $\phi_2(t_2)$,

(ii) NOT ($\phi_1(t_2)$ and $\phi_2(t_1)$)

then $t_1 \succ_L^w t_2$.

Whereas the above discussion resolves the problem of the introduction of semantics for a single expression, things are more complicated when it comes to the introduction of semantics for a whole profile. Specifically, the following problems have to be considered:

- Is it possible that two expressions in a qualitative profile have a conflict?
- Can we determine a-priori that a profiles is polythetic or monothetic?
- In the case of a polythetic profile, where a tuple can be assigned to the extent of more than one expressions, how do we ultimately decide its final extent?

5.4.1 Conflicts in qualitative profiles

Transitivity allows us to deduce inconsistencies for a profile. We can find problems of conflicting preferences concerning the precedence between preference expressions as well as inconsistencies concerning the degree of precedence.

Cyclic preferences. The first case of inconsistency has to do with *cyclic preferences*. Assume the case of a profile P with the following three preferences: $p_a(\phi_1, \phi_2, w_a)$, $p_b(\phi_2, \phi_3, w_b)$, $p_c(\phi_3, \phi_1, w_c)$. Clearly this presents a problem since a simple application of transitivity will result in every expression being preferred to everyone else. We need to avoid this kind of situations and to this end we introduce *acyclic profiles*[3]. To keep the discussion concise and to the point, in the discussions of this section, we will extend the notions of preference and transitivity from tuples to preferences.

[3]Observe that scores in quantitative profiles relieve us from this kind of problems

Definition 19 *(Acyclic profiles) A qualitative profile is acyclic if for every preference expression $p(\phi_1, \phi_2, w)$ of the profile it is impossible to derive an expression $p'(\phi_2, \phi_1, w')$, either directly or transitively.*

Weight Inconsistencies. Given a function to compute simple item transitivity, is it always possible to compute exactly one value for the transitive distance between two tuples? Assume a domain $\mathbf{W}_w = \{little, fairly, a\,lot\}$ and a function f_{sit} producing the maximum of the two values as the derived weight for the transitive relationship. Assume also the profile of Figure 4 that is pictorially represented as a Hasse diagram. Then, it is clear that (a) the left path in the figure indicates the albums of `Accept` are preferred to the ones of `Metallica` with a weight *a lot*, whereas (b) the right path indicates that they are preferred with a weight *little*.

Can we assume that this is a possibly rare case for real-world users? Actually, there exist reports (see [SJ99]) that contradict this intuitive assumption; in fact, even simple properties of distance functions (e.g., $distance(A, B)$ $> distance(A, A)$, $A \neq B$) appear to be violated by user responses. Therefore, we believe it is necessary to require preference management systems to be tolerant to inconsistencies caused by user input.

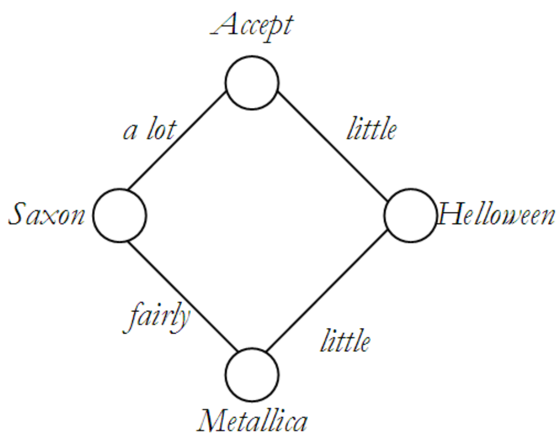

Figure 4: Inconsistency for the distance of two preferences via multiple paths

Proposition 5.1 *It is still possible to have two different evaluations for the distance of two preference expressions via transitivity.*

Given a qualitative profile P defined over a domain $\mathbf{W}_w$ and a relation R, as well as a simple item transitivity function f_{sit} we introduce the following two definitions:

- The profile is *extensionally transitively consistent*, if for every pair of tuples t_1 and t_2 for whom we can compute a preference relation, there is a single weight w that characterizes this relationship– i.e., either $t_1 \succ^w t_2$ or $t_2 \succ^w t_1$.

- The profile is *intentionally transitively consistent*, if for every pair of preferences p_1 and p_2 for whom we can compute a preference relationship there is a single weight that characterizes this relationship.

5.4.2 Polythetic and monothetic profile identification

As with quantitative profiles, qualitative profiles may also cause polythetic classifications. Again, we will define two notions of monothetic classifications:

- *Intentional monothetic classification*, refers to the property of a profile to allow us to determine that every tuple will be related to at most one preference expression
- *Extensional monothetic classification* refers to the property of a specific instance of R to relate each of its tuples to at most one preference expression.

Assume a profile P over a relation R, $R = A_1, \ldots, A_n$. Without loss of generality, assume that the preference expressions of the profile involve only the attributes $A_1, \ldots, A_k$, $k \leq n$. Assume also that every expression of the profile is in conjunctive normal form and, consequently involves atoms each of which defines a range of values over the domain of an attribute A, $dom(A)$. Then, given a tuple t, all the possible values that define the preference(s) to the extent(s) of which the tuple t will eventually be classified are within the space $\Omega_k{=}dom(A_1) \times \ldots \times dom(A_k)$. Then, a profile is intentionally monothetic if every preference expression defines a subspace of the space Ω_k that is disjoint to every other subspace derived from the preference expressions.

Theorem 5.2 *Assume a qualitative profile P defined over a relation R, $R = \{A_1, \ldots, A_n\}$. P involves a finite set of preference expressions. The profile P is intentionally monothetic if the following hold:*

1. *There is a* core *of attributes $A_1, \ldots, A_k$, $k \leq n$, all of which are involved in all the expressions of P.*
2. *Every preference expression is of the form $\bigwedge_{i \in 1 \ldots k*} (A_i\ \theta_i\ v_i) \succ^w \bigwedge_{j \in 1 \ldots k*} (A_j\ \theta_j\ v_j)$, $k* \geq k$, with $\theta \in (<, >, \leq, \geq, =, \neq)$ and v denoting a value in $dom(A)$.*
3. *For the core attributes of each expression ϕ, we can derive a range-aware preference expression ϕ^* of the form $\bigwedge_{i \in 1 \ldots k} (A_i \in range_i)$, with $range_i$ being the union of disjoint value ranges in $dom(A_i)$.*

4. For every two expressions ϕ and ϕ' there exists at least one core attribute A whose range expressions in ϕ and ϕ' are completely disjoint.

Proof. Assume that the conditions of the theorem are all met and a tuple t is classified in more than one extents; i.e., it is involved with preference p_1 and p_2. Then, the two preferences share a common subspace within Ω_k. Still, this is impossible: by definition, even if $k-1$ out of the k ranges are identical, there must exist at least one core attribute A where the two preference expressions have disjoint ranges. In this case, the subspaces of the two preferences in the k-space are necessarily disjoint, too. ∎

The second requirement specifies the structure of the involved profiles. $k*$ is greater or equal than k, since (a) *all* the k attributes must be covered (and therefore we need at least k atoms per expression), (b) expressions of the form $A > v_{low} \wedge A < v_{high}$ can be specified by the user, and (c) expressions on non-core attributes are allowed. Here, we assume that the user provides meaningful ranges of values (although we do not cover meaningless cases as $sex =' Female', illness =' Prostate'$ – see next too).

Is it possible to use one attribute in only one preference? Yes, if it is not part of the core! Fundamentally, there need to be k common attributes in the core of the profile whose subspaces must be completely disjoint. Now, if the extent of a preference is actually a subset of the k-space that the core attributes assign to the preference, by using a range on another attribute outside the core, this is fine, too.

Observe that the third requirement calls for the union of ranges. This is due to the presence of atomic expressions of the form $A \neq v$ in the profiles, resulting in ranges of the form $(-\infty \ldots v-1] \cup [v+1 \ldots \infty)$ for domains isomorphic to the integers (or generally, $(-\infty \ldots v) \cup (v \ldots \infty)$ for the rest of the domains). Expressions without $\neq$ produce single ranges.

Clearly, the essence of the theorem lies in the ability to deduce that the regions in the k-space of the core of the preferences are mutually disjoint. The fourth requirement is hard but effective –especially, in the absence of any other information (e.g., functional dependencies, business constraints, etc). Therefore, relaxed variants of the theorem can be devised too when auxiliary information is available. Moreover, it is easy to see that extensionally monothetic profiles are profiles whose expressions have common subspaces that are empty of tuples.

5.4.3 Resolution of multiple extents in polythetic profiles

Monothetic profiles have no problems of semantics, since each tuple refers to at most one preference. What happens with polythetic profiles, though, when we must resolve a single extent to place a tuple that is related to one or more preferences? As the reader might recall, quantitative profiles resolve problems by combining scores or prioritizing preferences. Score combination is not an option for qualitative profiles so preference prioritization seems to be the only path tho follow.

Traditional composition schemes include Pareto and lexicographic composition. Pareto composition treats all preferences as equals and postulates that for tuple t_1 to be preferred over tuple t_2, t_1 must be better in at least one attribute and not worse in all the others than t_2. Lexicographic composition requires an a-priori ordering of preferences. By combining *all* preferences of a profile via lexicographic composition it is possible to derive a unique preference extent for every tuple.

Is it possible to exploit the preference networks to rank tuples? The answer to the question is not always positive. Let us revisit Ingo's profile in Figure 2. The album *Helloween* in Figure 1 satisfies two preferences (a) *Band* = `Helloween`, and (b) *Genre* = '*speed*'. Both preferences appear to be second best in Ingo's profile. So, unless a strict priotization of preferences is introduced, the profile remains polythetic. At the same time, one cannot avoid the thought that in some (but not all) cases like this where all the preference extents of a tuple are equally preferred, it is not really important to have a monothetic profile (since the result presented to the user will be the same).

Based on all the above, there are the following cases that we consider:

- The profile is defined as the lexicographic composition of all its preferences; in this case, a tuple belongs to the extent of the preference with the highest priority

- As a last resort, we can have a (possibly partial) specification of the prioritization of the preferences made by the user. Under this approach, when a tuple belongs to two preference extents, it is ranked according to a conflict resolution function f_{conf} with f_{conf} taking as input the two preference expressions and returning one of them who is the winner. For example,

$$f_{conf}(p_1, p_2) = \begin{cases} p_1 & if\ p_1 \succ p_2\ can\ be\ established \\ p_2 & if\ p_2 \succ p_1\ can\ be\ established \\ \oslash & otherwise \end{cases} \quad (3)$$

  Observe the last clause: if neither precedence can be established, then no preference wins and the tuple is not considered at all (!).

5.5 Range-Belief relations

Apart from the simple cases of transitivity, we can explore more elaborate ones. Consider the following example:

- Assume the finite domain $\mathbf{W}_w^* = \{very\ little, little, fair, a\ lot, strongly\}$.

- I like `Rainbow` more than `Deep Purple` with weight *very little*.
- I also like `Deep Purple` more than `Led Zeppelin` with weight *fair*.

Assume now that the domain is of interval type and we can compute the following distance, based on two user-defined variants of value addition and value distances (not listed here)

$$fair + (fair \text{ - } \textit{very little}) = \textit{a lot}$$

Then, we can claim that I prefer `Rainbow` to `Led Zeppelin` with strength *a lot*, based on simple item transitivity. Still, this is quite an unorthodox approach and requires an interval scale assumption and some non-intuitive definitions of arithmetic operations over values and their intervals. On the contrary, the most reasonable answer that we can deduce for the relationship of `Rainbow` and `Led Zeppelin` is that I prefer `Rainbow` to `Led Zeppelin` with strength (i) at least *fair* (i.e., the maximum of $fair, \textit{very little}$) and (ii) no more than *strongly*, which is the maximum of the domain.

To produce such relationships among tuples, we need to revisit our assumption that a preference weight is **exactly** w. To this end, we introduce a range belief preference relation as follows:

Definition 20 *(Range-Belief Preference Relation). A relation $\mathcal{P}$ is a range-belief qualitative preference relation over a database relation R if it is a subset of $(dom(A_1) \times \ldots \times dom(A_k) \times (dom(A_1) \times \ldots \times dom(A_k) \times \mathbf{W}_w \times \mathbf{W}_w$.*

*For two arbitrary tuples of R, say t_1 and t_2 and an arbitrary weight w, if $\mathcal{P}(t_1, t_2, w_{low}, w_{high})$ holds, it is necessary that $w_{low} \leq w_{high}$. Then, we write $t_1 \succ_{\mathcal{P}}^{[w_{low}, w_{high}]} t_2$ and read it as "t_1 is **strictly** preferred over t_2 with a weight that **belongs to the range** $[w_{low}, w_{high}]$".*

When the context is clear, we will simplify notation and write $t_1 \succ^{[w_{low}, w_{high}]} t_2$ only.

Obviously, $t_1 \succ_{\mathcal{P}}^{[w,w]} t_2 \equiv t_1 \succ_{\mathcal{P}}^{w} t_2$

In other words, a range-belief preference relation says that although we do not know the exact value for the weight of the relationship, we know that it falls within a closed range of values.

For infinitely countable domains, we can also slightly abuse the terminology of the closed interval of weights for preference relations and extend the notation by including ∞ and $-\infty$. This way, we can also support semantics of **at least** ($t_1 \succ^{[w_{low},\ \infty]} t_2$) and **at most** ($t_1 \succ^{[-\infty,\ w_{high}]} t_2$) weights.

Definition 21 *(Range Item Transitivity). For each t_1,t_2,$t_3 \in R$, and w_1,w_2 $\in \mathbf{W}_w$ the following holds:*

$$t_1 \succ^{w_1} t_2 \text{ and } t_2 \succ^{w_2} t_3 \Rightarrow t_1 \succ^{[w_{low}, w_{high}]} t_3,\ [w_{low}, w_{high}] = f_{range}(w_1, w_2),$$

where f_{range} *is a computable function* $f_{range} : \mathbf{W}_w^2 \to \mathbf{W}_w^2$.

5.6 Transitive Closure

Definition 22 *(Transitive Closure). The transitive closure of a weighted qualitative preference relation can be recursively computed by applying the transitivity rules until no results can be computed further.*

Assume a weighted preference relation P *over a database relation* R. *Then, the transitive closure of* P*, denoted as* P^**, is defined as follows:*

$$t_1 \succ_{P^*}^{w} t_2 \equiv \exists\ k \text{ in } \mathbb{N} \text{ s.t. } t_1 \succ_{Pk}^{w} t_2$$

where P^k*, is recursively defined as follows:*

1. $t_1 \succ_{P1}^{w} t_2 \equiv t_1 \succ_{P}^{w} t_2$
2. $t_1 \succ_{Pn}^{w} t_2 \equiv \exists\ t_3, w_{13}, w_{32}$ *s.t.* $t_1 \succ_{Pn-1}^{w13} t_3$ *and* $t_3 \succ_{P}^{w32} t_2$ *and* $w = f_{sit}(w_{13}, w_{32})$

An open issue for research is the conditions under which the transitive closure is consistent. In other words, assuming there are more than one ways to compute the distance between two preferences (i.e., via two different intermediate preferences), how can we guarantee that the outcome of the application of the transitive computations is the same?

6 Operations

In this section, we discuss operations concerning two weighted preference relations. Specifically, we will discuss the intersection, difference, union, and prioritized composition of two weighted preference relations. For each of these operations we will explore different possibilities concerning variants that have to do with the treatment of weights.

We will first define operations for plain weighted preference relations and then, we will extend our definitions for range-belief preference relations. We will employ the shorthand notation $\overline{w}$ to express a range $[w_{low}, w_{high}]$ and $\overline{w}^a$ to express a range $[w_l^a, w_h^a]$.

6.1 Projection

Assume a preference relation $\mathcal{P}_1$ defined over the database relation R with a domain of weight $\mathbf{W}_1$. Assume now a second domain for weights $\mathbf{W}_2$ and a total function $M_W\colon W_1 \to W_2$. The projection of a weighted qualitative preference $\mathcal{P}_1$ defined over the domain W_1 to a preference relation $\mathcal{P}_2$ defined over a domain W_2 is straightforward:

$$\forall\ (t_1, t_2, w) \in \mathcal{P}_1,\ (t_1, t_2, M_W(w)) \in \mathcal{P}_2$$

Interestingly, the mapping function $M_W\colon W_1 \to \{w_0\}$ projects the weighted qualitative preferences of $\mathcal{P}_1$ to simple qualitative preferences of traditional models [Cho03], [Kie02]. In other words, weights are eliminated from the expression of the preferences (e.g., instead of 'I prefer `Iron Maiden` to `Metallica` **a lot**' the new preference states 'I prefer `Iron Maiden` to `Metallica`').[4]

6.2 Intersection

Assume two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ defined over the same database relation R. The *plain intersection* of two weighted preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ is defined as the set of preferences that are common to both of them.

$$\mathcal{P}_1 \bigcap \mathcal{P}_2 : \{t_1 \succ^w t_2 \mid t_1 \succ^w t_2 \in \mathcal{P}_1, t_1 \succ^w t_2 \in \mathcal{P}_2\}$$

Obviously, for a preference to hold in the intersection of two preference relations, the same precedence and the same weight must be present in the two preference relations. We can relax the requirement on the equality of

[4] W_1 and W_2 should be of the same type and the properties of W_1 should be guaranteed by the mapping: for example, if it is of ratio type, then distances must be preserved under the mapping M_W. Still, this is a matter of measurement theory and outside the scope of this paper.

weights and provide a customizable intersection operator. Take for example the case where two friends are organizing a party and they must decide what music they will play. So, they compare their profiles and try to come up with a list of songs that they both like. Assume that Steve prefers `Saxon` to `Judas Priest` **a lot**, whereas Adrian prefers `Saxon` to `Judas Priest` **fairly**. In the plain definition of intersection, the preference of `Saxon` to `Judas Priest` would be ignored. Still, we can do better and decide that we have a rule that given the weights of preference assigns an "aggregate" value to the final weight. This can be either the minimum, or the maximum, or any other value belonging to the domain of weights $\mathbf{W}_s$.

Formally, we assume a function $f_\cap$, $f_\cap$: $\mathbf{W}_s \times \mathbf{W}_s \rightarrow \mathbf{W}_s$ that resolves conflicting weights for the same tuples of the database relation R. Then, we can define the *shallow intersection* of two weighted preference relations as follows:

$$\mathcal{P}_1 \bigcap^{f_\cap} \mathcal{P}_2 : \{t_1 \succ^{w} t_2 \mid t_1 \succ^{w_1} t_2 \in \mathcal{P}_1, t_1 \succ^{w_2} t_2 \in \mathcal{P}_2, w = f_\cap(w_1, w_2)\}$$

`Note`: An interesting case is when $f_\cap$ is $f_\cap$: $\mathbf{W}_s \times \mathbf{W}_s \rightarrow \{\oslash\}$, i.e., it removes any weight information from the preference.

Intersection for Range-Belief Preferences. Assume now that preferences are expressed as range-belief preferences. We can define the intersection of two range-belief preference relations in a similar manner to simple relations. First, we define the *plain intersection* of two range-belief preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ is defined as the set of preferences that are common to both of them.

$$\mathcal{P}_1 \bigcap \mathcal{P}_2 : \{t_1 \succ^{\overline{w}} t_2 \mid t_1 \succ^{\overline{w}} t_2 \in \mathcal{P}_1, t_1 \succ^{\overline{w}} t_2 \in \mathcal{P}_2\}$$

Next, we can define *shallow intersection* via a conflict resolution function. Formally, we assume a function $f_\cap$, $f_\cap$: $\mathbf{W}_s^4 \rightarrow \mathbf{W}_s^2$ that resolves conflicting weights for the same tuples of the database relation R. Then, we can define the shallow intersection of two range-belief preference relations as follows:

$$\mathcal{P}_1 \bigcap^{f_\cap} \mathcal{P}_2 : \{t_1 \succ^{\overline{w}} t_2 \mid t_1 \succ^{\overline{w}^a} t_2 \in \mathcal{P}_1, t_1 \succ^{\overline{w}^b} t_2 \in \mathcal{P}_2, \overline{w} = f_\cap(\overline{w}^a, \overline{w}^b)\}$$

Obviously, several functions could be candidates for $f_\cap$. To our point of view, the most reasonable choice for $f_\cap$ would be *common interval*. For example, assume that Adrian prefers `Saxon` to `Judas Priest` in the range [*fair*,*uttermost*] and Steve prefers `Saxon` to `Judas Priest` in the range [*a lot*,*very much*] (Figure 5). Then, it appears reasonable to select the common interval of the two ranges the new range-belief preference. In other words,
$\overline{w} = [max(fair, a\ lot),\ min(uttermost, very\ much)] = [a\ lot, very\ much]$.

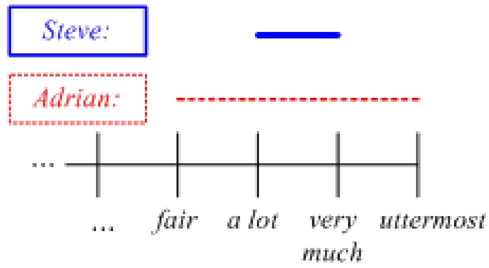

Figure 5: Intersection of Steve's and Adrian's profiles

6.3 Difference

Assume two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ defined over the same database relation R. *Plain difference* concerns the subtraction of the elements of a weighted preference relation $\mathcal{P}_2$ from a weighted preference relation $\mathcal{P}_1$.

$$\mathcal{P}_1 \; - \; \mathcal{P}_2 : \{t_1 \succ^w t_2 \mid t_1 \succ^w t_2 \in \mathcal{P}_1, t_1 \succ^w t_2 \notin \mathcal{P}_2\}$$

Similarly to intersection, this does not eliminate preferences over the same items but with different weights. So, again, we assume a *shallow difference* variant, which exploits a function f_-, f_-: $\mathbf{W}_s \times \mathbf{W}_s \rightarrow \mathbf{W}_s$ that resolves conflicting weights for the same tuples of the database relation R and is defined as follows:

$$\mathcal{P}_1 \; -^{f_-} \; \mathcal{P}_2 : \{t_1 \succ^w t_2 \mid t_1 \succ^w t_2 \in \mathcal{P}_1, \nexists w' \in \mathbf{W}_s, s.t., t_1 \succ^{w'} t_2 \in \mathcal{P}_2\}$$
$$\bigcup \{t_1 \succ^w t_2 \mid t_1 \succ^{w_1} t_2 \in \mathcal{P}_1, t_1 \succ^{w_2} t_2 \in \mathcal{P}_2, w = f_-(w_1, w_2)\}$$

Difference for Range-Belief Preferences. Difference is defined in a similar way to the previous definitions for both the plain and the shallow case.

The interesting question in the case of all kinds of preferences is the choice of the function f_-. If difference is allowed for the members of $\mathbf{W}_s$, then it quite possible to consider the difference of the weights (for simple weighted preferences), or the respective parts of the involved ranges (for range-belief preferences). A good example is to pick the complement of the common range of the two preferences, if a single such complement is defined. If we assume the aforementioned example with Adrian and Steve (Figure 5) then such a complement cannot be defined as there are two complements of the common range. Assuming now that Adrian prefers `Saxon` to `Judas Priest` in the range [*fair*,*uttermost*] and Steve prefers `Saxon` to `Judas Priest` in the range [*very much*,*uttermost*] (Figure 6) we can express the shallow difference of their preferences, as

$\overline{w} = single\, complement((fair, uttermost), (very\, much, uttermost)) = [fair,$ $a\ lot]$.

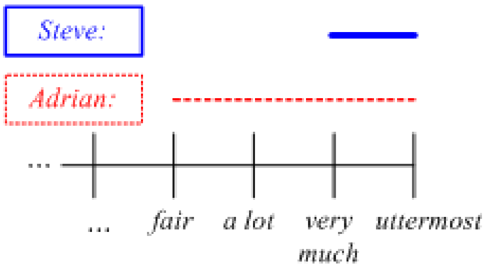

Figure 6: Difference of Steve's and Adrian's profiles

6.4 Union

There is no notion of plain union, since we need to take care of potential conflicts in advance. Coming back to the example of Steve and Adrian we can have the following two possible conflicts, along with a third non-conflict:

- they both prefer `Saxon` to `Judas Priest` but Steve with weight *a lot*, and Adrian with weight *fairly*,
- Steve prefers `Deep Purple` to `Black Sabbath` *fairly* and Adrian prefers `Black Sabbath` to `Deep Purple` *a lot*,
- Steve prefers `Wishbone Ash` to `AC\DC` and Adrian prefers `The Who` to `Deep Purple`, both with *fair* weight.

Then, a union result comprises three parts: (a) preferences belonging to $\mathcal{P}_1$ with no counterpart in $\mathcal{P}_2$, (b) preferences belonging to $\mathcal{P}_2$ with no counterpart in $\mathcal{P}_1$ and preferences belonging to both preference relations, either (c) with the same, or, (d) possibly with a different weight. Contradicting preferences are excluded. Again, for the last case (d), we assume a function $f_\cup$, $f_\cup$: $\mathbf{W}_s \times \mathbf{W}_s \rightarrow \mathbf{W}_s$ that resolves conflicting weights for the same tuples of the database relation R.

Formally, assume two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ defined over the same database relation R. The union of the two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ via the function $f_\cup$ for weight resolution is defined as follows:

$$\mathcal{P}_1 \overset{f_\cup}{\bigcup} \mathcal{P}_2 = \begin{cases} \{t_1 \succ^w t_2 \mid t_1 \succ^w t_2 \in \mathcal{P}_1, \nexists w' \in \mathbf{W}_s, s.t., t_1 \succ^{w'} t_2 \in \mathcal{P}_2 \bigvee t_2 \succ^{w'} t_1 \in \mathcal{P}_2\} \\ \{t_1 \succ^w t_2 \mid t_1 \succ^w t_2 \in \mathcal{P}_2, \nexists w' \in \mathbf{W}_s, s.t., t_1 \succ^{w'} t_2 \in \mathcal{P}_1 \bigvee t_2 \succ^{w'} t_1 \in \mathcal{P}_1\} \\ \{t_1 \succ^w t_2 \mid t_1 \succ^w t_2 \in \mathcal{P}_1, t_1 \succ^w t_2 \in \mathcal{P}_2\} \\ \{t_1 \succ^w t_2 \mid t_1 \succ^{w_1} t_2 \in \mathcal{P}_1, t_1 \succ^{w_2} t_2 \in \mathcal{P}_2, w = f_\cup(w_1, w_2)\} \end{cases} \tag{4}$$

Interestingly, the union operation treats both preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ equally (and symmetrically). So, when Steve and Adrian disagree on the issue of `Deep Purple` and `Black Sabbath`, nobody's preference wins. Now assume that the party takes place for Adrian's birthday and Steve gives in whenever a conflict occurs. This brings us to the prioritized composition of two preference relations which is dealt with in the following subsection.

Union for Range-Belief Preferences. Before proceeding to prioritized composition, we need to consider what happens with the union of range-belief preferences. The main idea is quite similar to the case of simple weighted preferences; in fact, the three first cases are the same. The only difference of simple and range-belief preference relations has to do with the case when the two preferences have a different weight. Again, a function $f_{\cup}$ defined over ranges this time must be defined. A simple example is the function that returns the union of two ranges, if such a union can produce a *single* interval. Assume the following situation (Figure 7):

- Bruce prefers `Saxon` to `Judas Priest` with a range *little* to *fairly*,
- Steve prefers `Saxon` to `Judas Priest` with a range *very much* to *uttermost*,
- Adrian prefers `Saxon` to `Judas Priest` with a range of *fair* to *very much*,.

Then, we cannot define a union for the preference of Bruce and Steve. Still, Bruce and Adrian can produce a union for their preferences with a range of *little* to *very much*, and Adrian with Steve can have a range of *fair* to *uttermost*.

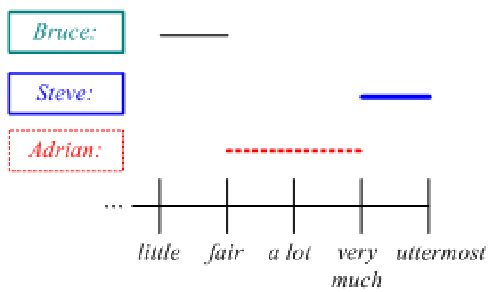

Figure 7: Union for Bruce's, Steve's and Adrian's profiles

6.5 Prioritized Composition

The prioritized composition of two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ favors the former over the latter. So, it comprises the elements of $\mathcal{P}_1$ and any elements of $\mathcal{P}_2$ that are not covered due to identity or conflict by the elements of $\mathcal{P}_1$.

Formally, assume two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ defined over the same database relation R. The prioritized composition of the two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ is defined as follows:

$$\mathcal{P}_1 \,\&\, \mathcal{P}_2 : \{t_1 \succ^w t_2 \mid t_1 \succ^w t_2 \in \mathcal{P}_1 \bigvee (t_1 || t_2 \in \mathcal{P}_1 \wedge t_1 \succ^w t_2 \in \mathcal{P}_2\}$$

Observe how conflicts are resolved in favor of $\mathcal{P}_1$. Observe also the indifference operator in the case where we incorporate elements of $\mathcal{P}_2$ in the result.

The definition is the same when we consider the case of range-belief preferences. If there is an overlap in the ranges of two preferences with the same expression, then the first preference wins.

In both union and prioritized composition it is a modality whether we want to deal with preferences that concern the same items but with different weight. We believe that the definitions we give cover the most practical case; still, there are other possibilities that we leave as unexplored ground for the future.

6.6 Open Issues

There are several issues not explored in this paper. We try to give a brief presentation of these issues that are open for future research.

Distance. What is the distance of two preference relations? What is the distance of two profiles?Is it possible to resort to techniques measuring graph distance to the rescue?

Composition of preferences over different database relations So far, in all the operations concerning two preference relations, we have assumed two preference relations $\mathcal{P}_1$ and $\mathcal{P}_2$ defined over the *same* database relation R. What happens if we need to combine preferences from *different* database relations?

Assume the case where James wants to form a band. James is responsible for the vocals of the band and he has a list of candidates for the band's lead guitar players as well as his preferences about them. The same applies for his preferences over bass guitar candidates. So, he inputs his data into the relations $LeadGuitarHero$ and $BassGuitarHero$ respectively. Now, James must decide which combination of lead and bass guitar players will be the best and this has to come up via a combination of the individual preferences over lead and bass guitarists.

There are two possible ways to combine the preferences over different database relations. The Pareto composition treats both preference relations as equal. The Lexicographic composition treats one preference relation as more important than the other (and resembles a lot the aforementioned prioritized composition). The product of the two preference relations is a result not intuitively clear. The precise intuition and semantics of the above notions are open to definition and further exploration.

7 Preference Networks and Naive Profile Consolidation

In this section, we will present a preliminary way to consolidate quantitative and qualitative profiles. We will impose specific assumptions to guarantee the well-formedness of the involved profiles as well as the resulting profile and we will also explore risks and implications of this naive consolidation.

7.1 Assumptions

In the rest of our deliberations, we will make several necessary assumptions that will guarantee the consistency of the involved profiles and the validity of the transformations that we will define. For reasons of intuitive and algorithmic consistency we require that all profiles are acyclic; moreover, we need to make several other assumptions that pertain to the structure of the graphs of the involved profiles.

- Clearly, a cycle in the network of a profile indicates an intuitive problem; therefore, we need to verify that all the profiles result in directed acyclic graphs. In the sequel, *we will assume that all profiles are acyclic.*

- A profile that is disconnected presents a challenging case where the user has specified that there are two disjoint areas of interest for him. Dealing with each component in isolation is a possible work-around for the subsequent operations that we will discuss; still, it is possible that some kind of profile consolidations will unite the disconnected components. For the moment, *we restrict the scope of our investigation to profile networks with exactly one component.*

- It is possible for qualitative profiles to have edges with unknown weights (equivalently, with the value *unknown*). Unless otherwise specified, we avoid these cases since they unnecessarily complicate the functions we need to define in the sequel and *assume that all the edge weights on a qualitative profile are known.* We conjecture that extensions to cope with unknown values are also feasible.

- In the rest of our deliberations, we will assume that the quantitative profiles that we will use are *lines* (all the scores are different). We conjecture that an extension of this assumption to lattices is feasible but not within the scope of this paper.

- We need to assume that $\mathbf{W}_w$ *is at least ordinal* (and therefore, its members can be ordered). This also allows us to "order" the nodes of a quantitative profile according to their score.

7.2 Naive Consolidation without Transformations

A simple way to merge a quantitative and a qualitative profile is to simply identify the common preference expressions of the two profiles and connect the respective networks at these places.

Definition 23 *(Naive Consolidation). An* intentional preference network *L_I of a qualitative profile P_w and a quantitative profile P_s over a relation R is a graph $G(V, E)$ constructed as follows:*

1. *Each expression participating in any kind of preference of any of the two profiles is mapped to a node $v \in V$. If an expression appears more than once, exactly one node is used. If an expression is found in both profiles it is called a* junction expression *and its corresponding node a* junction node*.*

2. *The strength of a quantitative preference annotates the node of the respective expression.*

3. *Given two expressions ϕ_1 and ϕ_2, a directed edge ($\phi_2 \rightarrow \phi_1$) connects the nodes of the two expressions if ϕ_1* covers *ϕ_2. If the two expressions belong to the qualitative profile and $\phi_1 \succ^w \phi_2$, the edge is annotated with the appropriate weight w.*

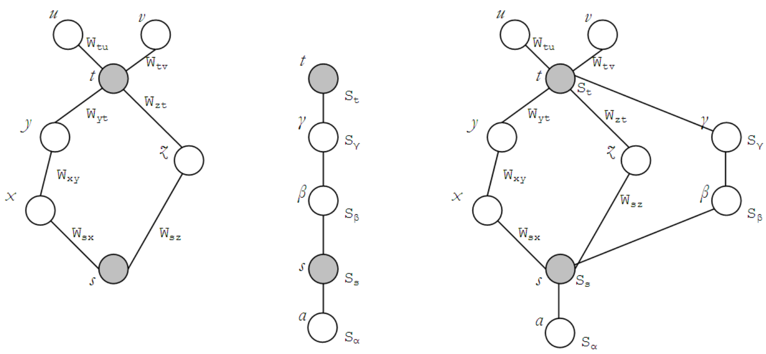

Figure 8: A qualitative and a quantitative profile and their naive consolidation

Observe Figure 8 where a qualitative and a quantitative profile are depicted in the left and middle of the figure. The two profiles have two common nodes s and t and each is well specified in its own purview: all the nodes of the quantitative profile have scores $(s_\alpha, s_s, s_\beta, s_\gamma, s_t)$ and all the edges of the qualitative profile have weights $(w_{sx}, w_{xy}, w_{yt}, w_{sz}, w_{zt}, w_{tu}, w_{tv})$. If we consolidate the two profiles into one with the method of this section, then the resulting network is depicted on the right-hand side of the figure.

Consistency considerations. Let us make the following assumptions for the original qualitative and quantitative profiles:

- Both profiles are acyclic, and with exactly one component.
- The quantitative profile is a line.
- For every path $x \to \ \ldots\ \to y$ in the qualitative profile, there is no path $y \to \ \ldots\ \to x$ in the quantitative profile and vice versa.

The last assumption guarantees that the ordering of nodes in the two profiles is analogous (it is similar to state that we can find an topological ordering of the nodes of the qualitative profile that respects the order of the nodes of the quantitative line). Observe that the restriction is of existential and not universal nature (i.e., we do not require that every path in one profile has a homologous path in the other profile, but rather, that a path of the opposite direction does not exist).

Can we be assured that the resulting network is well formed? We can state the following proposition:

Proposition 7.1 *Given a quantitative and a qualitative profile that respect the aforementioned assumptions, the network that results from their naive consolidation is acyclic.*

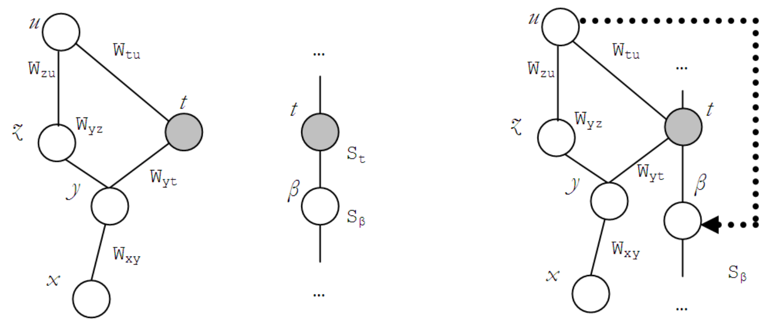

Figure 9: Counter-example for the possibility of cycles in the consolidated profile

Proof. There are two possibilities for a cycle to exist in the consolidated profile. The first possibility is that a path $x \to \ \ldots\ \to y$ in one profile, is combined with a path $y \to \ \ldots\ \to x$ in the other profile. This is ruled out by the preconditions we have set. The second case is that a new edge is added that creates such a path. This is not possible either. Observe Figure 9. On the left hand side we depict a qualitative and a quantitative profile; on the right hand side their consolidation is depicted along with an edge that generates a cycle. Assuming a junction node t is it possible to have an edge from a node u that follows t (i.e., there exists a path $t \rightarrow \ \ldots \ \rightarrow u$) to a node β that precedes t (i.e., there is a path $\beta \rightarrow \ \ldots \ \rightarrow t$)? Observe that the consolidation does not generate any new edges, therefore an edge from a node in the qualitative profile towards a node in the quantitative profile can only exist if both these nodes are junction nodes. In this case, though, at least one of the two profiles already has a cycle, which is not possible based on the assumptions we have already made. Therefore, a cycle is not possible. ∎

Can we always perform a consolidation of two profiles? Observe the case of Figure 10. On the right hand side the user has specified a qualitative profile. The profile states that the user prefers tuples fulfilling predicate A to the ones fulfilling predicate C. Similarly, the user prefers tuples fulfilling predicate z to the ones fulfilling predicate B. The most preferred tuples, however, are the ones fulfilling predicate x. Observe that B is not related to A or C at all (although the user knows the existence of these predicates since he uses them in his profile). Then, the system makes a quantitative recommendation which is depicted as a line in the middle of Figure 10. In this case, as far as the junction nodes of the two profiles are concerned, A is preferred to B which, in turn, is preferred to C. Some other predicates (all in lowercase) are also part of the two profiles (named with latin characters in the qualitative, and greek characters in the quantitative profile, respectively). Can we consolidate the two profiles? All the necessary requirements are met; so, one could argue that in principle there is no problem. However, the resulting consolidated profile (depicted in the right hand side of the figure) extends the user's profile with new information (depicted as dotted edges) that is possible violating the original relative positions of the user's preferences: now, B is explicitly preferred to C, although the user had no such specification in his original profile.

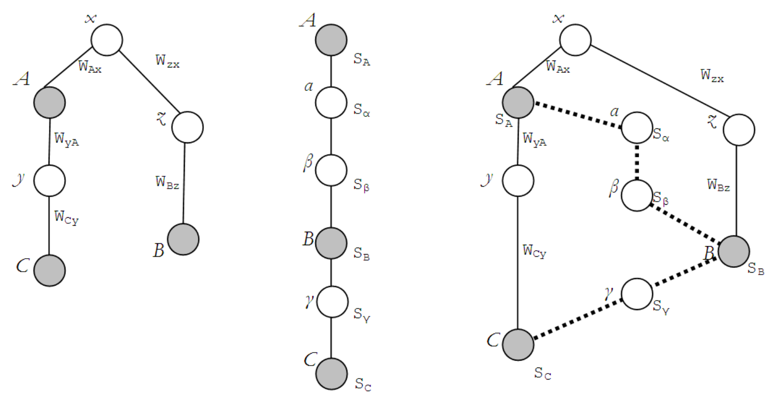

Figure 10: A legal consolidation with side-effects

There are several paths to handle this kind of situations. We could possibly disallow the merging of this kind of profiles, requiring that no new information is added for the junction points. Intuitively, we would like to have profiles such that, for every pair of junction nodes A,B for which a path $A \rightarrow \ \ldots \ \rightarrow B$ exists in the quantitative profile (necessarily, due to the linear nature of the quantitative profile), there is also a path $A \rightarrow \ \ldots \ \rightarrow B$ in the qualitative profile, too.

Note here that one could define a conservative consolidation policy that respects the quantitative profile, in a similar fashion.

We will formalize the notions of suitability of two profiles for consolidation as follows:

Definition 24 *(Loosely Homologous profiles) A quantitative profile P_s and a qualitative profile P_w are* loosely homologous *if for every two junction points in the line of the quantitative profile having no other junction node between them (but maybe other non-junction nodes), there is no path in the qualitative profile that passes from the higher node without passing from the lower one.*

For example, assuming two junction nodes A and B with A being higher than B, there is no path $Z \rightarrow \ \ldots \ \rightarrow A$ starting from another junction point Z that does not pass from B. In other words, homologous profiles have the nice property that as far as junction points are concerned, every path in the qualitative profile passes exactly from the same (consequently, *all*) junction nodes as in the quantitative line.

Definition 25 *(Strongly Homologous profiles) A quantitative profile P_s and a qualitative profile P_w are* strongly homologous *if for every path in the qualitative profile, all the junction nodes are met in the same order as in the quantitative line.*

Figure 11 presents different cases of homology between a quantitative and several qualitative profiles.

We can have various levels of strictness in our consolidation policies.

1. We can follow a *conservative consolidation policy* and restrict ourselves to loosely or strongly homologous profiles. This way the number of semantic discrepancies is minimized. A conservative policy assumes that consolidated profiles are small extensions of the existing profiles (e.g., a qualitative profile specified by the user is slightly extended by a suggestions made my a machine-learning algorithm)

2. In several cases, it is possible to avoid conservative policies and adopt a more *liberal consolidation policy* that treats both profiles equally and

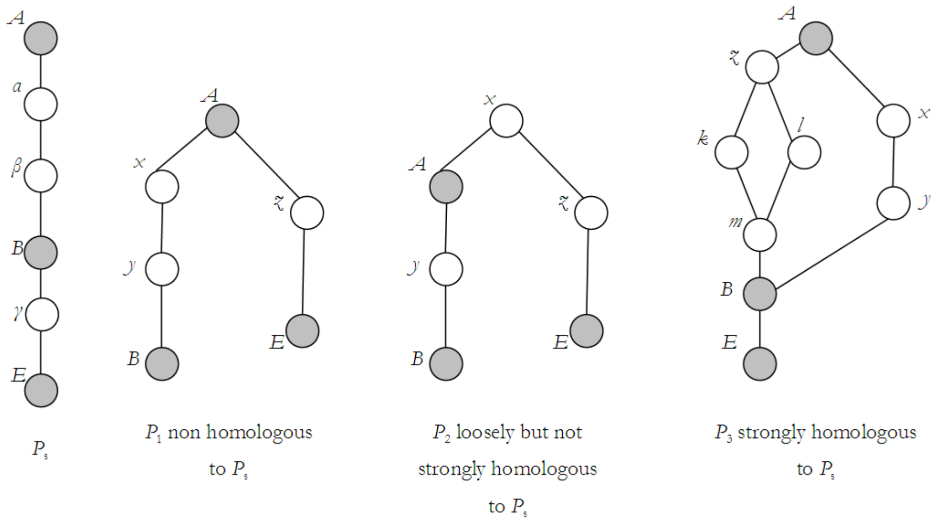

Figure 11: A quantitative profile and several qualitative profiles with different degrees of homology to it

allows such discrepancies. This way, we disallow consolidations only when cycles are not introduced in the resulting profile.

Observe Figure 8 again. Despite the fact that the originating profiles were fully specified in terms of weights and scores, the new profile is not. What we are missing from the setting of Figure 8 is (a) the weights of the edges in the quantitative graph and (b) the scores of the nodes of the qualitative graph. How can we compute the missing information? This will be the topic of the following section.

8 Mapping of quantitative to qualitative preferences

Two well formedness properties for a consolidated profile can be defined:

- A preference network is *qualitatively fully specified* if all the edges of the graph are annotated with appropriate weights w, $w \in \mathbf{W}_w$.
- A preference network is *quantitatively fully specified* if all the nodes of the graph are annotated with appropriate scores s, $s \in \mathbf{W}_s$.
- A preference network is *fully specified* if it is both quantitatively and qualitatively fully specified.

A simple way to provide fully defined networks is to map one kind of profile to another and fill the missing information (edge weights in the quantitative graphs and node scores in the qualitative graph).

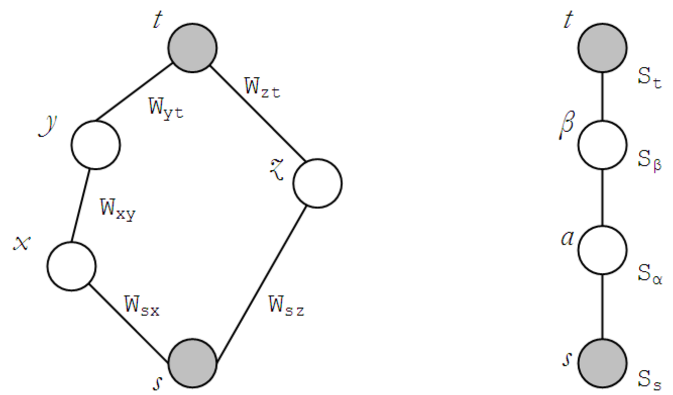

Figure 12: Reference example for the computation of distances

Intuitively, assume that we know the distance $\delta(s,t)$ between the nodes s and t for the graph of Figure 12. Then, we can base the computation of edge weights in the quantitative graph on the assumption that the weight of an edge in the qualitative path is the percentage of the distance that corresponds to this edge over the total distance between the s and t. Similarly, the score of a node v in the qualitative graph is the percentage of the distance of the path between s and v over the total distance of the path between s and t.

Then, if we want to compute the weight of the edge (β,α), $w_{\beta\alpha}$, the intuitive way to do it is:

$w_{\beta\alpha} = \frac{\delta(\beta_s,\alpha_s)}{\delta(t_s,s_s)}$, with $\delta()$ being the function that computes the distance between two scores and translates it to a weight for an edge. In other words, the weight of the edge between the node β and the node α is the ratio of the distance of scores of β and α over the total distance of the scores of s and t.

Similarly, if we want to compute the score of node x s_x, the intuitive solution is:

$$s_x = \delta(s_t, s_s) * \frac{weight(path(s,x))}{weight(path(s,t))} + s_s$$

The distance of the scores of nodes s and t multiplied by the percentage of the weight of the path from s to x over the total weight of the path from s to t signifies what part of this score distance corresponds to the node x. Then, the score of x is s_s incremented by this quantity.

Limitations. All the above intuitive translations are sensitive to the potential of the domains of weights and scores. The domain $\mathbf{W}_s$ of scores must be at least of interval scale, so that the difference of two scores can be defined. On the other hand, the domain $\mathbf{W}_w$ of edge weights must be of ratio scale so that the ratio of weights can be computed. This is quite a burdensome requirement. In the rest of our deliberations, unless otherwise specified, we will assume that the domain $\mathbf{W}_s$ of scores is of interval scale (and only). We will also assume that the domain $\mathbf{W}_w$ of weights is of ordinal scale (and only) and only the ordering of its values is naturally given. The goal of the following subsections is to provide mappings between quantitative and qualitative preferences. We will achieve this based on some external functions that complement the domains for that purpose only.

8.1 Mapping quantitative to qualitative preferences

Quantitative preference expressions can be directly mapped to qualitative expressions by exploiting the preference scores. Intuitively, if the quantitative preference score for an expression ϕ is greater than the respective score for a tuple ϕ', then, $\phi \succ \phi'$. Still this does not help us determine the weight w of the qualitative relation, and for this purpose we need a function that translates the "difference" in the quantitative scores to a qualitative preference weight.

Formally, assume the following:

- a relation R and two expressions, ϕ_1 and ϕ_2 defined over R
- an interval domain $\mathbf{W}_s$ for quantitative scores and an ordinal domain $\mathbf{W}_w$ for qualitative weights

- two preferences, p_1 and p_2 that assign a score s_1 to ϕ_1 and s_2 to ϕ_2, s_1,$s_2 \in \mathbf{W}_s$; without loss of generality assume $s_1 > s_2$ (remember that $\mathbf{W}_s$ is at least in the ordinal scale)

- a function f_{sw}, f_{sw}: $\mathbf{W}_s \times \mathbf{W}_s \to \mathbf{W}_w$

Then, a qualitative relation can de derived where ϕ_1 dominates ϕ_2 by the result of f_{sw}, i.e., $\phi_1 \succ^{f_{sw}(s_1,s_2)} \phi_2$.

Example. Assume a quantitative domain $\mathbf{W}_s = \{0.1, 0.2, \ldots, 1.0\}$ and a qualitative domain $\mathbf{W}_w = \{little, fairly, a\ lot\}$. Assuming two scores, s_1 and s_2, $s_1 > s_2$, we can define a function f_{sw} as follows:

$$f_{sw} = \begin{cases} little, & if\ diff(s_1, s_2) = 0.1 \\ fairly, & if\ diff(s_1, s_2) \in [0.2, 0.4] \\ a\ lot, & otherwise \end{cases} \tag{5}$$

There are several important observations that we can make for the mapping of quantitative to qualitative preferences.

- Observe that we make no particular assumption neither on the nature, nor on the relationship of the domains of quantitative scores and qualitative weights $\mathbf{W}_s$ and $\mathbf{W}_w$, respectively. For example, we could have a simple ordinal domain for scores "I like `Savatage` a lot and `Metallica` fairly" and a ratio scale for strengths "I prefer `Savatage` twice to `Metallica`" (or, equivalently, $t_S \succ^2 t_M$, with t_S any tuple fulfilling the predicate *band=Savatage* and t_M any tuple fulfilling the predicate *band=$Metallica$*).

- Since the qualitative relationship ultimately refers to tuples and not expressions (and each tuple is related to exactly one preference formula), *it is not necessary that all expressions are defined over the same attribute.*

- The aforementioned mechanism can be extended to map scores to range-belief weights.

A simple algorithm can be devised on the previous scheme to produce a well specified profile when given a quantitative linear profile.

Algorithm Full Specification of Quantitative Line

Input: A linear quantitative profile P; an ordinal domain of weights $\mathbf{W}_w$; an interval domain of scores $\mathbf{W}_s$; a function f_{sw} translating score differences to edge weights

Output: The well specified profile P' with all its edges annotated with weights in $\mathbf{W}_w$

begin
- *current* = node at the bottom of line P;
- **for** *every edge e=(current,current.next)* **do**
  - *e.weight*=f_{sw}(*current*,*current.next*);
  - *current* = *current.next*;

end

Observations. The following observations can be made for the previous transformation:

- Obviously, there is no alteration in the structure of the graph. The *covers* relationship is maintained.
- If one wants to derive the qualitative distance of two consecutive nodes via the transitivity functions of the qualitative weights, this will not necessarily produce the same results as with the function f_{sw}. The circumstances under which this can be guaranteed are a topic for future research.

8.2 Mapping qualitative to quantitative profiles

The mapping of qualitative to quantitative profiles is based on augmenting a given score for a node of a qualitative profile on the basis of the weights of the edges of the paths that stem from this node.

Formally, assume the following:

- a relation R and two expressions, ϕ_1 and ϕ_2 defined over R and connected via a qualitative edge with a weight w from ϕ_2 to ϕ_1 (i.e., $\phi_1 \succ^w \phi_2$)
- an interval domain $\mathbf{W}_s$ for quantitative scores and an ordinal domain $\mathbf{W}_w$ for qualitative weights
- a quantitative preference, p_2 that assigns a score s_2 to ϕ_2
- a function f_{ws}, f_{ws}: $\mathbf{W}_s \times \mathbf{W}_w \rightarrow \mathbf{W}_s$

Then, a quantitative relation can de derived where ϕ_1 dominates ϕ_2 with weight w and ϕ_1 is annotated with the score that is the result of $f_{ws}(s_2, w)$.

Example. Assume a quantitative domain $\mathbf{W}_s = \{0.1, 0.2, \ldots, 1.0\}$ and a qualitative domain $\mathbf{W}_w = \{little, fairly, a\ lot\}$. Assuming a score and a weight, w and s_2, we can define a function f_{ws} as follows:

$$f_{ws} = \begin{cases} min(1.0, s_2 + 0.1), & if\ w = little, \\ min(1.0, s_2 + 0.3), & if\ w = fairly \\ min(1.0, s_2 + 0.6), & otherwise \end{cases} \tag{6}$$

The annotation of a qualitative profile with scores can be produced via a DFS exploration of the graph, assuming scores are available for the fountains of the network. Each time a new node is visited, its score can be computed on the basis of the score of the previous node and the weight that connects them. As already mentioned in previous sections, it is quite possible that the passing of the algorithm from the same node for a second time might produce a different score. Given a set of scores for the same node ϕ, we need a reconciliation function f_{rec} that reduces the list to a single score. Then, since a profile network can be topologically ordered it is easy to devise an algorithm that (a) topologically sorts the profile and computes the individual scores and dependencies each time, and (b) reconciles the individual scores of the same node into a single result.

The issue of being able to determine a-priori whether a quantitative profile suffers from the problem of assigning multiple scores to a single expression given a set of original scores for its fountains is left open for the moment.

Things get even more complicated when we have more information than a single node. Observe the slightly changed Cliff's profile of Figure 13 and assume we know the scores for two nodes: $Genre = heavy$, with a score of 0.2 and $Band = Iced\ Earth$ with a score of 0.8. Assume also a quantitative domain $\mathbf{W}_s = \{0.1, 0.2, \ldots, 1.0\}$ and a qualitative domain $\mathbf{W}_w = \{little, fairly, a\ lot\}$ as well as the aforementioned function for the derivation of scores for the nodes of a qualitative profile.

The main problems here are two: (a) node *thrash and speed* has a score of 0.9, which is already higher than the known score of node $Band = Iced\ Earth$ and (b) node $Band = Iced\ Earth$ has three scores: a given 0.8, a score of 1.0 coming from the clearly problematic left-hand path and a score of 0.7 coming from the right-hand path to it. How should the situation be treated?

The main transcendence that we need to perform here is the barrier of the type of $\mathbf{W}_w$, which is of ordinal scale. What if we could annotate the domain with a function that transforms it to ratio scale (which allows the computation of both the difference and the ratio of edge weights)? Assume for the moment that we can make the following annotation:

$fairly = 2 * little$, $a\ lot = 3 * little$

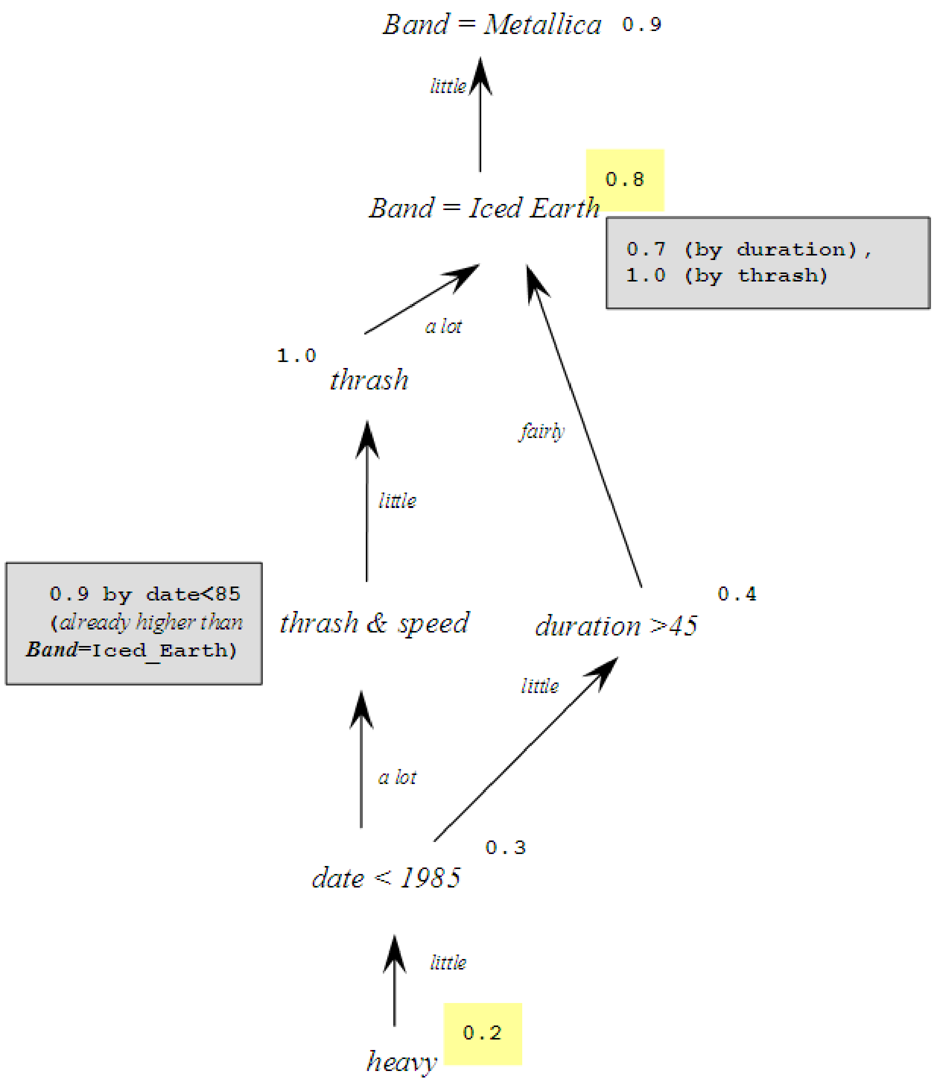

Figure 13: Cliff's qualitative profile and problems of deriving scores for its preferences

Then, all we would have to do is to divide the distance between *date* $<$ 1985 and *Band* $=$ *Iced Earth* in such a way that every path is divided appropriately to sub-distances according to the weights of the edges. To be accurate, the overall process requires (a) a transformation of the distances to the scale $[0 \ldots 1]$, (b) the computation of distances and scores for nodes, (c) the mapping of the computed scores back to values of the domain of $\mathbf{W}_w$. In our example here, we are conveniently placed already in the appropriate scaled domain and we use *round*() to map computed scores to scores in $\mathbf{W}_w$.

Node	Comp. Score	Score		Node	Comp. Score	Score
thrash-speed	0.51	0.5		duration>45	0.46	0.5
thrash	0.58	0.6		Iced Earth	0.8	0.8
Iced Earth	0.8	0.8				

Observe that things are not entirely all right, even with this trick: the distance from the left-hand path is 7 times *little* and the distance from the right-hand path is 3 times *little* (!). Still, this method gives the flexibility of adapting to user-defined preferences, even if they are not consistent.

Formally, we need the following components to be able to perform the correct calculation of scores that are bounded by an upper an lower value.

- Assume two nodes, v_{low} and v_{high} of a qualitative profile, with v_{high} preceding over v_{low}. Assume also two scores v_{low} and s_{high} assigned to the two nodes, $s_{low} < s_{high}$.
- Assume a function f_{dist}^{norm} that given the two scores, computes their distance in the normalized range $[0 \ldots 1]$
- Assume a distance granule τ and a mapping of each possible edge weight to a multiplicand of the distance granule (this way the domain of weights is transformed to the ratio scale)
- Assume also a function f_{denorm} that reverts values in the range $[0 \ldots 1]$ to actual scores

Then, the method for computing the distance of two score-annotated nodes in a qualitative preference network is simple:

1. Compute the distance of the two scores.
2. For each path that connects the two nodes, find the sum of the edge weights expressed in granules τ and compute the percentage of the distance that corresponds to the granule τ for this path.
3. Perform DFS over the nodes of the paths and assign each node with a score expressed as the sum of the score of the previously visited node augmented with the weight of the edge multiplied by the percentage of the distance that corresponds to it.

Discussion. Observe that this scheme removes the discrepancies of multiple paths. Moreover, this scheme is a possible aid to check the consistency of a profile. Nevertheless, the whole endeavor is based on a questionable assumption: we are able to map an ordinal domain to a ratio scale. Is it legitimate to do so?

Intuitively, based on the simple examples that we have used, it appears that such an extension is reasonable, especially since the user need not know its existence. It also seems reasonable to base the proposed method on ratios, in order to keep intermediate scores bounded within an upper and lower threshold. This blocks cumulative error propagation too. Nevertheless, the validity of such an assumption heavily relies on extensive user studies that corroborate the assumption or not – especially since it appears that users demonstrate unexpected behavior during such studies [SJ99]. To the best of our knowledge, there is no experimental study that assesses the quality of results after such a transformation. So, the issue is left open for further exploration in the future.

9 Shallow Consolidation of qualitative and quantitative preferences

In this section, we provide a simple extension to the naive consolidation of a qualitative and a quantitative profile. The extension is based on the assumption that we can transform a linear quantitative profile to a fully-specified profile with edge weights.

9.1 Shallow consolidation with emphasis to weights

The shallow consolidation of qualitative and quantitative preferences unifies the two profiles into one with a light degree of intermingling of the elements of the two profiles. All paths of the two profiles among the same junction nodes are placed in parallel (and therefore, their nodes are not comparable to each other).

Definition 26 *(Shallow consolidation of a quantitative and a qualitative profile with emphasis on weights). An* intentional weight-oriented preference network *L_I of a qualitative profile P_w and a quantitative profile P_s over a relation R is a graph $G(V, E)$ constructed as follows:*

1. *Each expression participating in any kind of preference is mapped to a node $v \in V$. If an expression appears more than once, exactly one node is used.*
2. *The strength of a quantitative preference annotates the node of the respective expression.*
3. *Transform the profile P_s to a fully specified profile P'_s.*
4. *Given two expressions ϕ_1 and ϕ_2, a directed edge ($\phi_2 \rightarrow \phi_1$) connects the nodes of the two expressions with a weight w if ϕ_1* covers *ϕ_2.*

The transformation is oriented towards qualitative preferences. This is quite handy in the cases where we are not able to transform a qualitative profile to a fully specified profile (i.e., we cannot compute scores for it) and thus, we are in need to work only with edge weights.

A fundamental property of the above merging is that the two profiles remain disjoint except for the matching points of common expressions. In other words, assuming an edge (v_1, v_2) in one of the two profiles, the edge will remain intact after the merging. This why we call the merging of the two profiles *shallow merging*.

Example. Assume Cliff enters a website where he can download music. Cliff has created a qualitative profile with his preferences, depicted on the left side of Figure 14. We use solid lines to refer to qualitative profiles. At the same time, based on previous searches and purchases, the site has

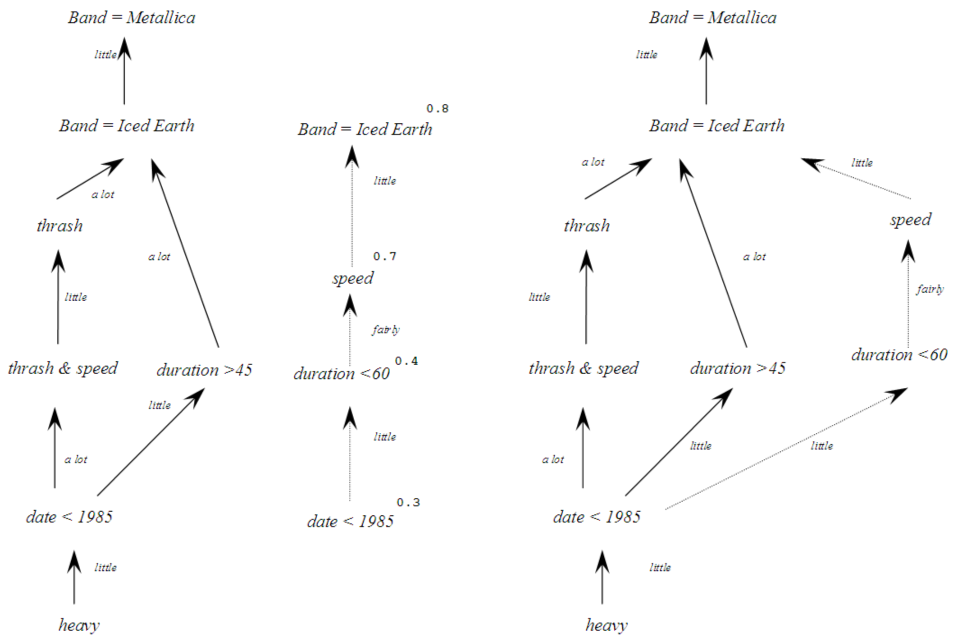

Figure 14: Shallow unification for the qualitative and the quantitative parts of Cliff's profile, with emphasis to weights

automatically created a quantitative profile for Cliff, which suggests that Cliff likes:

- albums published before 1985 with a score of 0.3,
- albums with a duration of less than 60 minutes with a score of 0.4,
- albums in the genre of `speed metal` with a score of 0.7
- albums produced by `Iced Earth` with a score of 0.8

In the middle of Figure 14, we depict how the quantitative profile mentioned above is mapped to a qualitative profile as well as to the corresponding graph. We represent the edges that result from this transformation with dotted lines.

We use function f_{sw} as defined in the previous section. Then, the shallow consolidation of the quantitative and the qualitative part of Cliff's profile is depicted on the right hand side of Figure 14.

9.2 Fully specified shallow consolidation

Assume now that we are able to convert both scores to weights in the original quantitative profile and edge weights to scores in the original qualitative profile. Then, we can produce fully specified profiles for the two input profiles, and consequently, a fully specified profile for their shallow unification.

Definition 27 *(Shallow consolidation of a quantitative and a qualitative profile). An* intentional preference network L_I *of a qualitative profile* P_w *and a quantitative profile* P_s *over a relation* R *is a graph* $G(V, E)$ *constructed as follows:*

1. *Transform the profiles* P_s *and* P_w *to fully specified profiles* P'_s *and* P'_w*.*
2. *Each expression participating in any kind of preference is mapped to a node* $v \in V$*. If an expression appears more than once, exactly one node is used.*
3. *The strength of a preference annotates the node of the respective expression.*
4. *Given two expressions* ϕ_1 *and* ϕ_2*, a directed edge* $(\phi_2 \rightarrow \phi_1)$ *connects the nodes of the two expressions with a weight* w *if* ϕ_1 covers ϕ_2 *in any of the profiles* P'_s *and* P'_w*.*

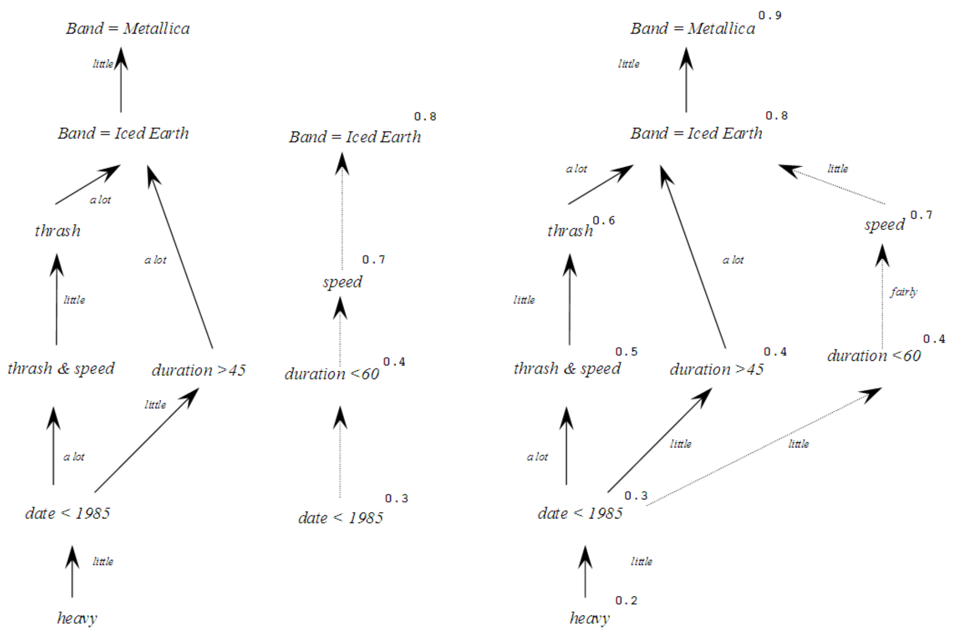

Figure 15: Shallow unification for the qualitative and the quantitative parts of Cliff's profile

The transformation of qualitative profiles requires the following iterative procedure:

- We identify all junction nodes over the qualitative profile.
- Starting from the lower junction node, the algorithm iteratively selects a pair of consecutive junction nodes in the quantitative line. These

two will serve as the nodes providing upper and lower scores for the computation of the scores of the non-junction nodes between them. All the non-junction nodes of the qualitative profile are annotated with the appropriate scores.

- All the non-junction nodes placed higher than the highest junction node or lower than the lowest junction node are annotated with scores derived by the iterative application of a function f_{ws}.

Coming back to the example with Cliff's profile (Figure 15), the result of the application of the transformations of the original profiles is shown on the right-hand side of the figure.

9.3 Consistency considerations

How can we be assured that (a) two profiles can be consolidated, in the first place and (b) their consolidated preference network is valid? As in previous sections, we make the following assumptions for the original qualitative and quantitative profiles:

1. Both profiles are acyclic, and with exactly one component.
2. The quantitative profile is a line.
3. For every path $x \to \ \ldots \ \to y$ in the qualitative profile, there is no path $y \to \ \ldots \ \to x$ in the quantitative profile and vice versa.
4. For each edge $e(x,y,w)$ of the qualitative profile that relates node x with node y via a weight w, there exists no edge $e'(x,y,w')$, $w \neq w'$, in the network that results from the transformation of the quantitative profile to a well specified network (and vice versa).

Can we be assured that the resulting network is well formed? We can state the following proposition:

Theorem 9.1 *Given a quantitative and a qualitative profile that respect the assumptions (1) - (3), the network that results from their shallow consolidation is acyclic. If assumptions (1) - (4) are respected, then each edge has a unique weight.*

Proof. The three first assumptions are the same as in the previous case of naive merging. The last assumption guarantees that once the quantitative profile is transformed to a well specified network, the weights in the two graphs that are going to be merged are consistent. It suffices to perform the check only for the edges of one of the two graphs. ∎

On the other hand, it is possible to have inconsistencies in the distance between two nodes of the resulting network, even if no such inconsistencies exist in the first place. Refer to the case of Figure 4: even if the weights are such that the transitive preference for `Accept` over `Metallica` is consistent between two paths (e.g., in the case that all edges have the weight *little*), it is possible that the consolidation with a quantitative profile creates an inconsistency in the case that both `Accept` and `Metallica` are junction points. In general, since new paths are created in the process of shallow consolidation, even if the path weights are consistent place for each pair of nodes in the original qualitative profile before the consoslidation, there is still the danger of resulting in a transitively inconsistent network.

10 Deep Consolidation

The consolidation method described in the previous section provides a single profile; nevertheless, the profile that is produced via this process is practically a loose coupling of the quantitative and the qualitative profiles. This is due to the fact that apart from the junction points of the two profiles, the remainders of the profiles remain unrelated.

In this section, we proceed in consolidating qualitative and quantitative profiles with deep consolidation semantics. The main idea behind deep consolidation is that the the resulting profile will not keep the elements coming from the originating profiles intact, but rather, it will produce new relationships for them on the basis of their junction nodes, weights and scores.

Given a qualitative and a quantitative profiles, the algorithm proceeds in the following way:

1. First, the algorithm constructs the respective graphs of the two profiles and computes all their junction nodes.

2. For each profile, we compute its fully specified counterpart. Therefore, all nodes are annotated with scores and all edges are annotated with weights in both profiles

3. The nodes of the quantitative profile's line that are higher (lower) the upper (lower) junction node are attached to the qualitative profile in the same order as in the quantitative profile.

4. Starting from the lower junction node of the line, the algorithm iteratively selects a pair of consecutive junction nodes in the quantitative line. Clearly, non-junction nodes can be placed between them. Then, the algorithm tries to merge all the nodes that lie within these two junction nodes either in the qualitative, or in the quantitative profile into a new network.

How do we merge the nodes found between the two consecutive junction nodes? We will exploit quantitative scores and a function f_{sw} to construct the consolidated subgraph. Given two junction nodes v_{low} and v_{high}, the consolidation proceeds as follows:

1. The set S of all the non-junction nodes between v_{low} and v_{high}in both profiles is constructed. The *covers* relationship on the basis of scores is computed.

2. For each pair of nodes u, v that belong to S an edge is added to the network, annotated with a score $w = f_{sw}(score(u), score(v))$.

Observe again the deep consolidation for the case of Cliff's profile. The right-hand side of the figure depicts the network that results from the deep

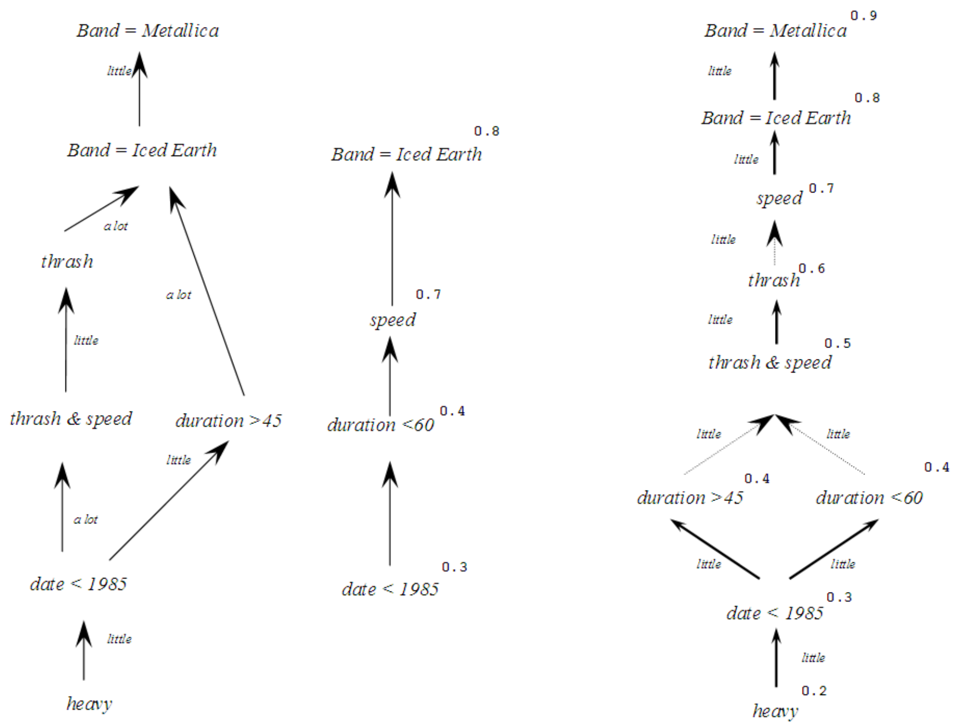

Figure 16: Deep consolidation for Cliff's quantitative and qualitative profiles

consolidation of the two profiles depicted on the left-hand side. In the resulting network, the original edges are depicted in thick solid lines and the newly introduced edges are depicted in dotted lines.

11 Conclusions

In this paper, we have addressed the combination of qualitative and quantitative profiles by introducing preference networks and weighted qualitative preferences. Preference networks formally capture precedence relationships between the expressions of a profile in any of the above categories (and subsequently of the tuples of a relation annotated with the profile) whereas weighted qualitative preferences capture not only preferences, but degrees of preference, too. A weight annotates a preference expression and allows us to infer further properties of the interrelationships between the preferences within a profile. We also explore properties and possible operations for weighted qualitative preferences with emphasis to the case of transitivity. Based on these underpinnings, we have provided methods to map qualitative to quantitative profiles and vice versa and explored the properties and consistency requirements for this kind of transformations. We have also provided different types of profile consolidation between a qualitative and quantitative profile. Throughout the paper, we pay particular attention to consistency checks and well-formedness properties for profiles that will be consolidated.

Moreover, we thoroughly explore the formal properties of the domains out of which scores and preference weights take their values, we discuss properties of quantitative preferences not explored in the past and introduce range-belief preferences to handle degrees of uncertainty about the degree of user preferences.

Future work can pursued in different paths. From a theoretical standpoint, problem like the consistent and efficient computation of transitive closure need further exploration. From an algorithmic standpoint, it is worth considering the possibility of incorporating preference networks and weighted qualitative preferences in the existing body of algorithms for query results ranking. From a modeling standpoint, open issues involve the exploration of preferences bearing degrees of uncertainty and their properties as well as the integration of both weighted and range belief preferences in other contexts, like e.g., OLAP (see for example [GR09]).

Acknowledgments. The author would like to thank T. Dalamagas and G. Koutrika as well as anonymous referees of a previous version of this paper for valuable comments that improved the manuscript's clarity.

References

[Cho03] Jan Chomicki. Preference formulas in relational queries. *ACM Trans. Database Syst.*, 28(4):427–466, 2003.

[FP97] Norman E. Fenton and Shari Lawrence Pfleeger. *Software Metrics*. International Thomson Publishing, 2nd edition, 1997.

[GKC+08] Periklis Georgiadis, Ioannis Kapantaidakis, Vassilis Christophides, Elhadji Mamadou Nguer, and Nicolas Spyratos. Efficient rewriting algorithms for preference queries. In *Proceedings of the 24th International Conference on Data Engineering, ICDE 2008, April 7-12, 2008, Cancún, México*, pages 1101–1110. IEEE, 2008.

[GR09] Matteo Golfarelli and Stefano Rizzi. Expressing OLAP preferences. In *Proceedings of Scientific and Statistical Database Management, 21st International Conference, SSDBM 2009, New Orleans, LA, USA, June 2-4, 2009*, pages 83–91, 2009.

[GZ88] Georg Gottlob and Roberto Zicari. Closed world databases opened through null values. In *Proceedings of the 14th International Conference on Very Large Data Bases, VLDB 1988, August 29 - September 1, 1988, Los Angeles, California, USA*, pages 50–61. Morgan Kaufmann, 1988.

[KI05a] Georgia Koutrika and Yannis E. Ioannidis. Constrained optimalities in query personalization. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2005, Baltimore, Maryland, USA, June 14-16, 2005*, pages 73–84, 2005.

[KI05b] Georgia Koutrika and Yannis E. Ioannidis. Personalized queries under a generalized preference model. In *Proceedings of the 21st International Conference on Data Engineering, ICDE 2005, 5-8 April 2005, Tokyo, Japan*, pages 841–852, 2005.

[Kie02] Werner Kießling. Foundations of preferences in database systems. In *Proceedings of 28th International Conference on Very Large Data Bases, VLDB 2002, August 20-23, 2002, Hong Kong, China*, pages 311–322. Morgan Kaufmann, 2002.

[KK02] Werner Kießling and Gerhard Köstler. Preference SQL - design, implementation, experiences. In *Proceedings of 28th International Conference on Very Large Data Bases, VLDB 2002, August 20-23, 2002, Hong Kong, China*, pages 990–1001. Morgan Kaufmann, 2002.

[SJ99] Simone Santini and Ramesh Jain. Similarity measures. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(9):871–883, 1999.

[SPV07] Kostas Stefanidis, Evaggelia Pitoura, and Panos Vassiliadis. Adding context to preferences. In *Proceedings of the 23rd International Conference on Data Engineering, ICDE 2007, April*

15-20, 2007, The Marmara Hotel, Constantinople, Turkey, pages 846–855, 2007.