# An algebra of database preferences

**Bernhard Möller, Patrick Roocks**

**Angaben zur Veröffentlichung / Publication details:**

Möller, Bernhard, and Patrick Roocks. 2015. "An algebra of database preferences." *Journal of Logical and Algebraic Methods in Programming* 84 (3): 456–81.
https://doi.org/10.1016/j.jlamp.2015.01.001.

An Algebra of Database Preferences

Bernhard Möller[a], Patrick Roocks[a]

[a] *Institut für Informatik, Universität Augsburg, D-86135 Augsburg, Germany*

**Abstract**

Preferences allow more flexible and personalised queries in database systems. Evaluation of such a query means to select the maximal elements from the respective database w.r.t. to the preference, which is a partial strict-order. We present a point-free calculus of such preferences and exemplify its use in proving algebraic laws about preferences that can be used in query optimisation. We show that this calculus can be mechanised using off-the-shelf automated first-order theorem provers.

*Keywords:* relational algebra, complex preferences, preference algebra

## 1. Introduction

In the field of databases, the relational calculus is a well established discipline which, among other applications, is used in algebraic query optimisation. The classical operations there are union, difference, cartesian product, selection and projection. The queries treated with this calculus mostly pose so-called *hard constraints*, by which the objects sought in the database are clearly and sharply characterised. Hence, if there are no exact matches the empty result set is returned, which is very frustrating for users.

As a remedy, over the last decade queries with *soft constraints* have been studied. These arise from a formalisation of the *user's preferences* in the form of partial strict-orders. A realisation of this idea is presented by the language Preference SQL [11] and its current implementation [12]. For example, a consumer wants to buy a new car. Her preference relation has two components: she prefers cars with high power and low fuel consumption. Hence, in addition to offering the possibility of specifying simple preferences, one needs flexible and powerful combination operators. With their help hierarchies of user wishes can be handled by *complex preferences*. The bottom of the hierarchy is formed by *base preferences* like "lowest fuel consumption". These are combined using the constructs of the *Pareto* and the *Prioritisation* preferences which model equal and more/less important user preferences. Since the resulting strict-orders are partial, they frequently admit many best or maximal database objects, which helps to avoid empty result sets for queries.

Naturally, also for queries using such preferences efficient optimisation has to be performed, for which an algebraic calculus is indispensable. Although there is by now a well developed set of preference constructors with associated algebraic laws, the underlying theory was based on pointwise definitions with complex first-order formulas, which made proofs of new optimisation rules or the addition of further preference constructors a tedious and error-prone task. The present paper unifies and extends a point-free calculus for preference relations and their laws that has developed over the last two years and is meant to help in resolving the mentioned problems. Not least, it can be easily used with off-the-shelf automated theorem provers, which provides an additional level of trustworthyness.

Before we delve into the technical details, we provide some examples that illustrate the phenomena to be tackled.

*Email addresses:* moeller@informatik.uni-augsburg.de (Bernhard Möller), roocks@informatik.uni-augsburg.de (Patrick Roocks)

**Example 1.1.** We return to the sketched example concerning cars. The goals the user wants to achieve are *conflicting*, because cars with high power tend to have a higher fuel consumption. To get the optimal results according to both of these equally important goals from a database, the concept of *skyline queries* [3] is used: A car belongs to the result set if there is no other car which is better in both criteria, i.e. has a lower fuel consumption and a higher power. In a 2-dimensional diagram for both criteria the result set looks like a "skyline", viewed from the hypothetical optimum. Figure 1 shows such a skyline for the preference mentioned above, where the *mtcars* data set was taken, a standard example in the statistical computing language "R".

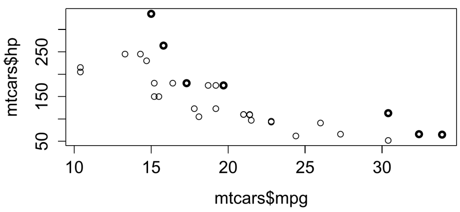


Figure 1: An example data set of cars where the *horsepower* and the value for *miles per gallon* (inverse fuel consumption) is depicted. The skyline for high power and low consumption is highlighted.

Concretely, consider the data set in table 1. The skyline query for minimal fuel consumption and maximal power returns the "BMW 5" and "Mercedes E", because each of these is better than the other by one criterion. The "Audi 6" is not returned, as it is worse by both criteria. □

| **Model** | **Fuel** | **Power** | **Color** |
|---|---|---|---|
| BMW 5 | 11.4 | 230 | silver |
| Mercedes E | 12.1 | 275 | black |
| Audi 6 | 12.7 | 225 | red |

Table 1: Example of a data set of cars

Imagine that a large database, for example a catalogue containing all the cars for the European market, returns a quite large result for the above skyline query. Assume that the consumer has even more wishes, for example prefers cars with a specific colour, but this is *less important* than the preference for low fuel and high power. This is formalised as follows.

**Example 1.2.** In the abstract notation of Preference SQL, the preference for "Lowest fuel consumption and (equally important) highest power, *both more important than* a preference for black cars" can be expressed by

$$P = (\text{LOWEST}(\mathit{fuel}) \otimes \text{HIGHEST}(\mathit{power})) \,\&\, \text{POS}(\mathit{color}, \{\text{black}\}) \ ,$$

where LOWEST and HIGHEST induce the "$<$" and "$>$" orders on their respective numerical domains, while POS creates a preference for values contained in the given set on a discrete domain. Pareto-composition and Prioritisation are denoted by ⊗ and & and are defined precisely later on; they might be pronounced "as well as" and "and then", respectively. □

In Preference SQL many base preferences have the nice property of being *layered* which means that their elements can be grouped into level in each of which the elements are pairwise incomparable. In order theory they are called *strict weak orders*, and prioritisation preserves that property. This allows fast algorithms and a very intuitive way to define *equally good* results: The incomparability relation w.r.t. a layered preference is an equivalence relation. Unfortunately the Pareto preference constructor does not preserve layeredness. This is the technical reason for an counter-intuitive effect which occurs in Preference SQL, shown in the following example.

**Example 1.3.** The best objects according to $P$ from Example 1.2 in the data set of table 1 are again "BMW 5" and "Mercedes E". This is quite counterintuitive, because the preference for black cars should decide for the Mercedes only. □

After these motivating examples we present the pointfree calculus of preferences developed in [20] and [18], extended by some additional material.

As our first contribution, we enrich the standard theory of relational databases by an algebraic framework that allows completely point-free reasoning about (complex) preferences and their best matches. This "black-box view" is amenable to a treatment in first-order logic and hence to fully automated proofs using off-the-shelf verification tools. We exemplify the use of the calculus with some non-trivial laws, notably concerning so-called preference prefilters (introduced in [6]), which perform a preselection to speed up the computation of the best matches proper, in particular, for queries involving expensive join operations. It turns out that the original laws hold under much weaker assumptions; moreover, several new ones are derived.

Additionally we define a transformation of the Pareto preference into a layered one which then avoids the counter-intuitive effect of the previous example. We show the well-definedness and some other interesting properties algebraically using our calculus. The complex preferences are represented in an abstract relation algebra embedded into a *join algebra* which allows reasoning about complex preferences in a point-free fashion.

New results with respect to [20] and [18] are the theorems about additivity of maxima and subset preferences and the discussion of normality and its relation to noetherity in Appendix E. Thereby we algebraically characterise normal elements as noetherian strict orders. Furthermore, we extend the results from [20] to the more general view of [18], thus achieving a much more uniform overall treatment.

## 2. Types and Tuples

In this section we present the formal framework to model database objects as tuples. We introduce typed relations whose types represent attributes, i.e. the columns of a database relation. Conceptually and notationally, we largely base on [9].

### *2.1. Typed Tuples*

**Definition 2.1.** Let $\mathcal{A}$ be a set of *attribute names* and a family $(D_A)_{A\in\mathcal{A}}$ of sets, where for $A \in \mathcal{A}$ the set $D_A$ is called the *domain* of $A$. We define the following notions:

1. A *type* $T$ is a subset $T \subseteq \mathcal{A}$.
2. An attribute $A \in \mathcal{A}$ is also used to denote the singleton type $\{A\}$, omitting the set braces.
3. A $T$*-tuple* is a mapping
$$t : T \to \bigcup_{A\in\mathcal{A}} D_A \;\text{ where }\; \forall\, A \in T : t(A) \in D_A \ .$$
4. For a $T$-tuple $t$ and a sub-type $T' \subseteq T$ we define the projection $\pi_{T'}(t)$ to $T'$ as the restriction of the mapping $t$ to $T'$: $\pi_{T'}(t) : T' \to \bigcup_{A\in\mathcal{A}} D_A$ with $A \mapsto t(A)$.
5. The domain $D_T$ for a type $T$ is the set of all $T$-tuples, i.e., $D_T = \prod\limits_{A\in T} D_A$.
6. The set $\mathcal{U} =_{df} \bigcup\limits_{T\subseteq\mathcal{A}} D_T$ is called the *universe*.
7. For a tuple $t$, and a set of tuples $M$ we introduce the following abbreviations:
$$t :: T \;\Leftrightarrow_{df}\; t \in D_T \ , \qquad M :: T \;\Leftrightarrow_{df}\; M \subseteq D_T \ .$$

**Definition 2.2 (Join).** The *join* of two types $T_1, T_2$ is the union of their attributes:
$$T_1 \bowtie T_2 \;=_{df}\; T_1 \cup T_2 \ .$$
For sets of tuples $M_i :: T_i$ $(i = 1, 2)$, the join is defined as the set of all consistent combinations of $M_i$-tuples:
$$M_1 \bowtie M_2 \;=_{df}\; \{t :: T_1 \bowtie T_2 \mid \pi_{T_i}(t) \in M_i, i = 1, 2\} \ .$$

We illustrate this concept with the following example.

**Example 2.3.** Assume a database of cars with unique IDs and further attributes for model and horsepower. Hence the attribute names, i.e. types, are $\mathsf{ID}, \mathsf{model}$ and $\mathsf{hp}$. The tuples are written as explicit mappings. Assume the following sets:

$$M_1 =_{df} \{\{\mathsf{ID} \mapsto 1,\ \mathsf{model} \mapsto \text{'BMW 7'}\}, \{\mathsf{ID} \mapsto 3,\ \mathsf{model} \mapsto \text{'Mercedes CLS'}\}\}\ ,$$
$$M_2 =_{df} \{\{\mathsf{ID} \mapsto 2,\ \mathsf{hp} \mapsto 230\}, \{\mathsf{ID} \mapsto 3,\ \mathsf{hp} \mapsto 315\}\}\ .$$

The sets have the types $M_1 :: \mathsf{ID} \bowtie \mathsf{model}$ and $M_2 :: \mathsf{ID} \bowtie \mathsf{hp}$. Now we consider the join $M_1 \bowtie M_2 :: \mathsf{ID} \bowtie \mathsf{model} \bowtie \mathsf{hp}$. We have $(\mathsf{ID} \bowtie \text{model}) \cap (\text{ID} \bowtie \mathsf{hp}) = \mathsf{ID}$. The only tuple $t :: \text{ID} \bowtie \mathsf{model} \bowtie \mathsf{hp}$ which fulfills both $\pi_{T_1}(t) \in M_1$ and $\pi_{T_2}(t) \in M_2$ is the one with $t : \mathsf{ID} \mapsto 3$. Hence the join is given by:

$$M_1 \bowtie M_2 = \{\{\mathsf{ID} \mapsto 3,\ \mathsf{model} \mapsto \text{'Mercedes CLS'},\ \mathsf{hp} \mapsto 315\}\}\ .$$

□

**Corollary 2.4.** *The following laws hold:*

1. $\bowtie$ *is associative and commutative and distributes over* $\cup$.
2. $\bowtie$ *preserves the inclusion order, i.e.* $M \bowtie N \subseteq M' \bowtie N$ *for* $M \subseteq M'$.
3. *Assume* $M_i, N_i :: T_i$ $(i = 1, 2)$. *Then the following exchange law holds:*

$$(M_1 \cap N_1) \bowtie (M_2 \cap N_2) = (M_1 \bowtie M_2) \cap (N_1 \bowtie N_2)\ .$$

Proof. 1. and 2. follow directly from the definitions. Using the definition of join and the usual intersection of sets we show the exchange law as follows:

$$\begin{aligned} & x \in (M_1 \cap N_1) \bowtie (M_2 \cap N_2) \\ \Leftrightarrow\ & \pi_{T_1}(x) \in M_1 \cap N_1 \ \wedge\ \pi_{T_2}(x) \in M_2 \cap N_2 \\ \Leftrightarrow\ & \pi_{T_1}(x) \in M_1 \ \wedge\ \pi_{T_1}(x) \in N_1 \ \wedge\ \pi_{T_2}(x) \in M_2 \ \wedge\ \pi_{T_2}(x) \in N_2 \\ \Leftrightarrow\ & x \in M_1 \bowtie M_2 \ \wedge\ x \in N_1 \bowtie N_2 \\ \Leftrightarrow\ & x \in (M_1 \bowtie M_2) \cap (N_1 \bowtie N_2)\ . \end{aligned}$$

□

*2.2. Typed Relations*

**Definition 2.5 (Typed homogeneous binary relations).** For a type $T$ we define the following abbreviations:

$$(t_1, t_2) :: T^2 \ \Leftrightarrow_{df}\ t_1, t_2 \in D_T\ , \qquad R :: T^2 \ \Leftrightarrow_{df}\ R \subseteq D_T \times D_T\ .$$

We say that the *typed relation* $R :: T^2$ has type $T$. There are some special relations for every type $T$: The full relation $\top_T =_{df} D_T \times D_T$, the identity $1_T =_{df} \{(x, x) \mid x \in D_T\}$ and the empty relation $0_T =_{df} \emptyset$.

This concept of typed relations also appears in the relation-based logical, but not primarily algebraic, approach to database notions of [14]. We will generalise it in Section 3.2.

**Definition 2.6 (Join of relations).** Let $R_i :: T_i^2$ $(i = 1, 2)$. Then the *join* $R_1 \bowtie R_2 :: (T_1 \bowtie T_2)^2$ is defined by

$$t\,(R_1 \bowtie R_2)\,u \ \Leftrightarrow_{df}\ \pi_{T_1}(t)\, R_1\, \pi_{T_1}(u) \ \wedge\ \pi_{T_2}(t)\, R_2\, \pi_{T_2}(u)\ .$$

**Corollary 2.7.**

1. *Assume* $M_i, N_i :: T_i$ $(i = 1, 2)$. *Then the following exchange law holds:*

$$(M_1 \bowtie M_2) \times (N_1 \bowtie N_2) = (M_1 \times N_1) \bowtie (M_2 \times N_2)\ .$$

2. *For types* $T_1, T_2$ *and* $X \in \{0, 1, \top\}$ *we have* $X_{T_1 \bowtie T_2} = X_{T_1} \bowtie X_{T_2}$.

Proof.

1. Straightforward from Definition 2.6.
2. Using part 1., $(D_{T_1} \bowtie D_{T_2}) \times (D_{T_1} \bowtie D_{T_2}) = (D_{T_1} \times D_{T_1}) \bowtie (D_{T_2} \times D_{T_2})$. By definition of the join for types we have that $T_1 \bowtie T_2 = T_1 \cup T_2$. From the definition of the join for sets we infer that $D_{T_1 \bowtie T_2} = D_{T_1} \bowtie D_{T_2}$. This shows the claim for $X = \top$. For $X = 1$ we show the equality componentwise using again the argument $D_{T_1 \bowtie T_2} = D_{T_1} \bowtie D_{T_2}$. For $X = \emptyset$ the claim is obvious. □

**Corollary 2.8.**

1. *For* $M, N :: T$ *we have* $M \bowtie N = M \cap N$. *In particular, we have* $N \bowtie N = N$.
2. *For* $R_1, R_2 :: T^2$ *we have* $R_1 \bowtie R_2 = R_1 \cap R_2$.
3. *For* $M_i :: T_i$ $(i = 1, 2)$ *with disjoint* $T_i$, *i.e., with* $T_1 \cap T_2 = \emptyset$, *the join* $M =_{df} M_1 \bowtie M_2$ *is isomorphic to the cartesian product of* $M_1$ *and* $M_2$.

Proof.

1. By the definition of join and the typing assumptions we have
$$t \in M \bowtie N \Leftrightarrow t \in M \wedge t \in N .$$
2. Similarly we conclude for all $x, y :: T$:
$$x\,(R_1 \bowtie R_2)\,y \Leftrightarrow \pi_T(x)\,R_i\,\pi_T(y) \ (i = 1, 2) \Leftrightarrow x\,R_1\,y \wedge x\,R_2\,y .$$
3. For $x \in M$, the two join conditions $\pi_{T_i}(x) \in M_i$ are independent. Hence all elements of $M_1$ can be joined with all elements of $M_2$. Thus, by definition,
$$t \in M \Leftrightarrow \pi_{T_1}(t) \in M_1 \wedge \pi_{T_2}(t) \in M_2 \Leftrightarrow (\pi_{T_1}(t), \pi_{T_2}(t)) \in M_1 \times M_2 .$$ □

### *2.3. Inverse Image and Maximal Elements*

**Definition 2.9 (Inverse image).** For a relation $R :: T^2$ the inverse image of a set $Y :: T$ under $R$ is formally defined as
$$\langle R \rangle\, Y =_{df} \{x :: T \mid \exists y \in Y : x\,R\,y\} .$$

The notation stems from the fact that in modal logic the inverse-image operator is a (forward) diamond.

**Lemma 2.10.** *Assume* $R_i :: T_i^2$ *and* $Y_i :: T_i$ $(i = 1, 2)$ *with disjoint* $T_1, T_2$. *Then the following exchange law for the join and the inverse image holds:*
$$\langle R_1 \bowtie R_2 \rangle\,(Y_1 \bowtie Y_2) = \langle R_1 \rangle\, Y_1 \bowtie \langle R_2 \rangle\, Y_2 .$$

Proof. Using the definition of the inverse image and the join of relations we infer:

$$\begin{aligned}
& x \in \langle R_1 \bowtie R_2 \rangle\,(Y_1 \bowtie Y_2) \\
\Leftrightarrow\ & \exists y \in (Y_1 \bowtie Y_2) : x\,(R_1 \bowtie R_2)\,y \\
\Leftrightarrow\ & \exists y \in (Y_1 \bowtie Y_2) : \pi_{T_1}(x)\,R_1\,\pi_{T_1}(y) \wedge \pi_{T_2}(x)\,R_2\,\pi_{T_2}(y) \\
\Leftrightarrow\ & \exists y_1 \in Y_1 : \exists y_2 \in Y_2 : \pi_{T_1}(x)\,R_1\,y_1 \wedge \pi_{T_2}(x)\,R_2\,y_2 \\
\Leftrightarrow\ & \pi_{T_1}(x) \in \langle R_1 \rangle\, Y_1 \wedge \pi_{T_2}(x) \in \langle R_2 \rangle\, Y_2 \\
\Leftrightarrow\ & x \in (\langle R_1 \rangle\, Y_1 \bowtie \langle R_2 \rangle\, Y_2) .
\end{aligned}$$

Note that splitting $y$ into $y_1$ and $y_2$ in the third step is justified by disjointness of the types: because of $T_1 \cap T_2 = \emptyset$ the two join conditions $\pi_{T_i}(y) \in Y_i$ for $i = 1, 2$ are independent of each other, hence the substitution $y_i := \pi_{T_i}(y)$ is allowed. □

Assume that $R_1, R_2$ are strict orders (irreflexive and transitive), which is the case in our application domain of preferences. Then, together with Corollary 2.8.3, this lemma means that, under the stated disjointness assumption, $R_1 \bowtie R_2$ behaves like the *product order* of $R_1$ and $R_2$ on the Cartesian product $D_{T_1} \times D_{T_2}$.

The inverse image of a set $Y$ under a relation $R$, when viewed the other way around, consists of the objects that have an $R$-successor in $Y$, i.e., are $R$-related to some object in $Y$ or, in the preference context, *dominated* by some object in $Y$. For this reason we can characterise the set of $R$-maximal objects within a set $Y$, as follows.

**Definition 2.11 (Maximal elements).** For a relation $R :: T^2$ and a set $Y :: T$ we define

$$R \triangleright Y =_{df} Y - \langle R \rangle Y \ ,$$

where "$-$" is set difference.

These are the $Y$-objects that do not have an $R$-successor in $Y$, i.e., are not dominated by any object in $Y$. The mnemonic behind the $\triangleright$ symbol is that in an order diagram for a preference relation $R$ the maximal objects within $Y$ are the peaks in $Y$; rotating the diagram clockwise by 90° puts the peaks to the right. Hence $R \triangleright Y$ might also be read as "$R$-peaks in $Y$".

To develop the central properties of our algebra and the maximality operator it turns out useful to abstract from the concrete setting of binary relations over sets of tuples, which will be done in the next section.

## 3. An Algebraic Calculus

Since we have shown how to characterise the maximal elements concisely using a diamond operation, it seems advantageous to reuse the known algebraic theory around that. This also allows us to exhibit clearly which assumptions are really necessary; it turns out that most of the development is completely independent of the properties of irreflexivity and transitivity that were originally assumed for preference relations in [10], and in fact also independent of the use of relations at all.

### *3.1. Semirings*

**Definition 3.1.** An *idempotent semiring* consists of a set $S$ of elements together with binary operations $+$ of *choice* and $\cdot$ of *composition*. Both are required to be associative, choice also to be commutative and idempotent. Moreover, composition has to distribute over choice in both arguments. Finally, there have to be units 0 for choice and 1 for composition, where 0 is an annihilator w.r.t. composition.

Binary homogeneous relations over a set form an idempotent semiring with choice $\cup$ and composition ";", which have $\emptyset$ and the identity relation as their respective units.

**Definition 3.2.** Every idempotent semiring induces a *subsumption order* by $x \leq y \Leftrightarrow x + y = y$. A *test* is an element $x \leq 1$ that has a complement $\neg x$ relative to 1, i.e., that satisfies

$$x + \neg x = 1 \ , \qquad x \cdot \neg x = 0 = \neg x \cdot x \ .$$

It is well known (e.g. [16]) that the complement is unique when it exists and that the set of all tests forms a Boolean algebra with $+$ as join and $\cdot$ as meet. Tests are used to represent subsets or assertions in an algebraic way. In the semiring of binary relations over a set $M$ the tests are subidentities, i.e., subsets of the identity relation, of the form $I_N =_{df} \{(x, x) \mid x \in N\}$ for some subset $N \subseteq M$ and hence in one-to-one correspondence with the subsets of $M$. Because of that we will, by a slight abuse of language, say that $x$ *lies in* $I_N$ when $(x, x) \in I_N$.

We will use small letters $a, b, c, ...$ at the beginning of the alphabet to denote arbitrary semiring elements and $p, q, ...$ to denote tests.

Based on complementation, the difference of two tests $p, q$ can be defined as $p - q =_{df} p \cdot \neg q$. It satisfies, among other laws,

$$(p+q) - r = (p-r) + (q-r) \ , \quad (p-q) - r = p - (q+r) \ ,$$
$$p - (q+r) = (p-q) \cdot (p-r) \ , \qquad p - (q \cdot r) = (p-q) + (p-r) \ .$$

For the interaction between complements and the subsumption ordering we can use the *shunting rule*

$$p \cdot q \leq r \ \Leftrightarrow \ p \leq \neg q + r \ .$$

A special case of applying this rule twice with $p = 1$ is the *contraposition* rule

$$q \leq r \ \Leftrightarrow \ \neg r \leq \neg q \ .$$

Tests can be used to express domain or range restrictions. For instance, when $a$ is a relation and $p, q$ are tests, $p \cdot a$ and $a \cdot q$ are the subrelations of $a$ all of whose initial points lie in $p$ and end points in $q$, respectively. Hence, all initial points of $a$ lie in $p$ if and only if $a \leq p \cdot a$.

With these properties we can give an algebraic characterisation of the test $\langle a \rangle \, q$ that represents the inverse image under $a$ of the set represented by $q$ or, equivalently, the set of initial points of $a \cdot q$.

**Definition 3.3.** Following [5], the *(forward) diamond* is axiomatised by the universal property

$$\langle a \rangle \, q \leq p \ \Leftrightarrow \ a \cdot q \leq p \cdot a \cdot q \ \Leftrightarrow \ a \cdot q \leq p \cdot a \ .$$

Following the terminology of [5], it would be more accurately termed a *pre-diamond*, since we do not require the axiom $\langle a \cdot b \rangle \, q = \langle a \rangle \, \langle b \rangle \, q$, which is not needed for our application. In the relational setting of [1], tests and diamonds are called monotoypes and monotype factors, respectively.

The diamond enjoys the following useful algebraic properties:

$$\langle a \rangle \, 0 = 0 \ , \quad \langle a + b \rangle \, p = \langle a \rangle \, p + \langle b \rangle \, p \ , \quad \langle a \rangle \, (p+q) = \langle a \rangle \, p + \langle a \rangle \, q \ , \quad \langle r \cdot a \rangle \, p = r \cdot \langle a \rangle \, p \ .$$

The middle two imply that diamond is isotone (i.e., monotonically increasing) in both arguments:

$$a \leq b \ \Rightarrow \ \langle a \rangle \, p \leq \langle b \rangle \, p \ , \quad p \leq q \ \Rightarrow \ \langle a \rangle \, p \leq \langle a \rangle \, q \ .$$

A special role is played by the test

$$\ulcorner a =_{df} \langle a \rangle \, 1 \ .$$

It represents the set of all objects that have an $a$-successor at all and therefore is called the *domain* of $a$. From the isotony of diamond we conclude, for test $p$,

$$\langle a \rangle \, p \leq \ulcorner a \ . \tag{1}$$

*3.2. Representing Types*

There are a number of ways to represent types algebraically, among them heterogeneous relation algebras [22], relational allegories [2] or typed Kleene algebra [13]. All these involve some amount of machinery and notation, which we want to avoid here.

More simply, we now interpret the largest test 1 as representing the universe $\mathcal{U}$ and use other tests to stand for subsets of it, e.g., for the domains associated with types. With every type $T \subseteq \mathcal{A}$, we associate a test $1_T$ representing its domain $D_T$. An assertion $p :: T$ means that $p$ is a test, representing a set of tuples, with $p \leq 1_T$. Arbitrary semiring elements $a, b, c, ...$ will stand for preference relations. A type assertion $a :: T^2$ is short for $a \leq 1_T \cdot a \cdot 1_T$. By $1_T \leq 1$ this can be strengthened to an equality. Hence, since tests are idempotent under composition, $a :: T^2$ implies $1_T \cdot a = a = a \cdot 1_T$.

To express that $x$ is an element, which represents a relation, or a test, we introduce the following notation:

$$x :: T^{(2)} \ \Leftrightarrow \ x :: T \ \vee \ x :: T^2 \ .$$

We assume for every type $T$ a greatest element $\top_T$ in $\{x \mid x :: T^{(2)}\}$, i.e., we assume $\forall\, x :: T^{(2)} : x \leq \top_T$ .

Additionally we introduce *sub-types*. Let $r :: T$ be a test. The *$r$-induced sub-type of $T$*, formally $T[r]$, is defined by the new identity $1_{T[r]} =_{df} r$. For the type assertions this implies

$$p :: T[r] \Leftrightarrow p \leq r \ , \quad a :: T[r]^2 \Leftrightarrow a \leq r{\cdot}a{\cdot}r \ . \tag{2}$$

Because of $r \leq 1$ the latter inequation strengthens to an equality. For the greatest element of this type we have $\top_{T[r]} = r{\cdot}\top_T{\cdot}r$ while the smallest element $0_{T[r]} = 0_T$ remains unchanged. As $T = T[1_T]$ the concept of subtypes generalizes the typing concept. The definitions imply that $T[r]$ and $T[r]^2$ are downward closed w.r.t. $\leq$.

For sake of readability we use the following abbreviations for $x :: T^2$:

$$0_x =_{df} 0_T \ , \quad 1_x =_{df} 1_T \ , \quad \top_x =_{df} \top_T \ .$$

Next we investigate how the complement operator behaves in a sub-type.

**Lemma 3.4.** *Let $p, q :: T[r]$ be a disjoint decomposition of $r$, i.e. $p+q = r$, $p \cdot q = 0_T$. Then $p, q$ are relative complements, i.e., we have $p = r - q$ and $q = r - p$.*

The proof can be found in Appendix B. Because of this property, to ease notation we abbreviate $r - p$ by $\neg p$ whenever $p :: T[r]$ is understood; there will be no ambiguity in our usage.

Next, we look at domain and diamond in sub-types. Assume $a :: T[r]^2$. First, by (2), idempotence of tests, a domain property and (2) again,

$$\ulcorner a = \ulcorner(r \cdot a \cdot r) = \ulcorner(r \cdot r \cdot a \cdot r) = r \cdot \ulcorner(r \cdot a \cdot r) = r \cdot \ulcorner a \ ,$$

i.e., $\ulcorner a :: T[r]$. Second, for arbitrary test $p$ we have by definition and isotony of domain

$$\langle a \rangle\, p = \ulcorner(a \cdot p) \leq \ulcorner a :: T[r] \ .$$

Finally, by definition, (2), idempotence of tests and (2) again,

$$\langle a \rangle\, r = \ulcorner(a \cdot r) = \ulcorner(r \cdot a \cdot r \cdot r) = \ulcorner(r \cdot a \cdot r) = \ulcorner a \ .$$

### *3.3. Abstract relation algebra*

The following definitions are the formal foundations for our preference calculus. First, we define abstract relation algebras using the axiomatisation in [15]:

**Definition 3.5 (Abstract relation algebra).** An *abstract relation algebra* is an idempotent semiring with additional operators $(...)^{-1}, \overline{(...)}$ for converse and complement, axiomatised by the Schröder equivalences and Huntington's axiom:

$$x{\cdot}y \leq z \Leftrightarrow x^{-1}{\cdot}\overline{z} \leq \overline{y} \Leftrightarrow \overline{z}{\cdot}y^{-1} \leq \overline{x} \ , \qquad x = \overline{\overline{x} + y} + \overline{\overline{x} + \overline{y}} \ .$$

For our applications, we additionally stipulate the Tarski rule

$$a \neq 0_a \Rightarrow \top_a{\cdot}a{\cdot}\top_a = \top_a \ ,$$

where $\top_a = \overline{0_a}$.

We assume that our underlying semiring is an abstract relation algebra and each type domain is closed under converse and complement, i.e. for $x :: T^{(2)}$ we have also $x^{-1}, \overline{x} :: T^{(2)}$. Note that this also holds for sub-types $T[r]$, where the complement is relative to $r$.

For an easier notation, we introduce the meet operation and the difference between two elements as follows:

$$x \sqcap y =_{df} \overline{\overline{x} + \overline{y}} \ , \qquad x - y =_{df} x \sqcap \overline{y} \ .$$

For relations these correspond to intersection and set difference. For tests $p, q \leq 1$ they coincide with composition and relative complement:

$$p \sqcap q = p{\cdot}q \ , \qquad p - q = p{\cdot}\neg q \ .$$

*3.4. Join Algebras*

We now deal with the central notion of join. For this, we assume the typing mechanism of the previous section.

**Definition 3.6 (Join algebra).** A *join algebra* is an abstract relation algebra with an additional binary operator $\bowtie$ satisfying the following requirements.

1. Join is associative, commutative and idempotent and distributes over choice $+$ in both arguments. Hence $\bowtie$ is isotone in both arguments.
2. Join is zero-strict, i.e., we have $a \bowtie 0_{T'} = 0_{T \bowtie T'}$ for all $a :: T^{(2)}$.
3. If $a_i :: T_i^{(2)}$ $(i = 1, 2)$ then $a_1 \bowtie a_2 :: (T_1 \bowtie T_2)^{(2)}$.
4. For types $T_i$ $(i = 1, 2)$ we have

$$1_{T_1 \bowtie T_2} = 1_{T_1} \bowtie 1_{T_2} \quad \text{and} \quad \top_{T_1 \bowtie T_2} = \top_{T_1} \bowtie \top_{T_2} \ .$$

5. Join and composition satisfy, for $a_i, b_i :: T_i^{(2)}$ $(i = 1, 2)$ with disjoint $T_i$, the exchange law

$$(a_1 \bowtie a_2) \cdot (b_1 \bowtie b_2) = (a_1 \cdot b_1) \bowtie (a_2 \cdot b_2) \ .$$

6. The diamond operator respects joins of elements with disjoint types: for $a :: T_1^2, p :: T_1$ and $b :: T_2^2, q :: T_2$ with $T_1 \cap T_2 = \emptyset$ we have the exchange law

$$\langle a \bowtie b \rangle \, (p \bowtie q) \ = \ \langle a \rangle \, p \bowtie \langle b \rangle \, q \ .$$

7. If $a_1, a_2 :: T^{(2)}$, then $a_1 \bowtie a_2 = a_1 \sqcap a_2$.
8. Join and meet satisfy, for $a_i, b_i :: T_i^{(2)}$ $(i = 1, 2)$ with disjoint $T_i$, the exchange law

$$(a \bowtie b) \sqcap (c \bowtie d) = (a \sqcap c) \bowtie (b \sqcap d) \ .$$

Our typed relations from Section 2.2 form a join algebra.

*3.5. Representation of Preferences*

Preferences introduced in [10] are strict partial orders, i.e. a special kind of binary homogeneous relations. These relations are defined on domains of types, and the objects compared are "database tuples" contained in a "database relation", i.e., a set of tuples.

To avoid confusion between the two uses of the word "relation" we call tuples *database elements* here and the database relation the *basic set* of objects. This means that we consider a "static" snapshot of the database at the time of the respective preference-based query and assume that no data is deleted or inserted into the database while the query is being evaluated.

Abstractly, preferences can now be modelled as typed elements $a :: T^2$ for some type $T$ representing strict partial orders.

**Definition 3.7 ((Layered) preferences).** A relation $a :: T^2$ is a *preference* if and only if it is irreflexive and transitive, i.e.

1. $a \sqcap 1_a = 0_a$,
2. $a \cdot a \leq a$.

It is a *layered preference* if additionally *negative transitivity* $\overline{a} \cdot \overline{a} \leq \overline{a}$ holds.

However, as we will see, for the most part the assumptions of irreflexivity and transitity are inessential for the laws we will derive.

Strict partial orders satisfying negative transitivity are sometimes called *strict weak orders*. In the scope of preferences, i.e. in [11] they are called *weak order preferences*. In this paper we will only use the term *layered preferences*. The reason for this is that such relations induce a "layered structure", i.e. there is always a function $f : D_A \to \mathbb{N}$ s.t. $t_1\, a\, t_2 \;\Leftrightarrow\; f(t_1) < f(t_2)$, which is shown in [7], Thm. 2.2.

Although we use an abstract relation algebra, in the examples one may always think of a concrete representation, where general elements $a, b, c, ...$ are relations and tests $p, q, r, ...$ are sets. To make our examples easy to follow, we sometimes use a point-wise notation: For a relation $a :: T^2$ the expression $t\, a\, t'$ for tuples $t, t' \in D_T$ means that the tuple $t$ is related to $t'$ via $a$. Analogously we use $\neg(t\, a\, t')$ if $t$ is not $a$-related to $t'$.

## 4. Maximal Element Algebra

Now we are ready for the algebraic treatment of our central notion.

### *4.1. Basic Definitions and Results*

**Definition 4.1.** The *best* or *maximal* objects w.r.t. element $a :: T^2$ and a test $p :: T$ are represented by the test

$$a \rhd p =_{df} p - \langle a \rangle\, p \;.$$

This definition is also given, in different notation, in [5]. An analogous formulation, however, with tests encoded as vectors, i.e., right-universal relations, can be found in [23]. To give a first impression of the algebra at work, we show a number of useful basic properties of the $\rhd$ operator.

In the following the test $r$ represents a finite data set. Requiring $a :: T[r]^2$ means that we restrict the element $a$ to this data set. Note that for $p \leq r$ we always have $(r\cdot a\cdot r) \rhd p = a \rhd p$, as the following calculation shows:

$$(r\cdot a\cdot r) \rhd p \;=\; p - \langle r\cdot a\cdot r\rangle\, p \;=\; p - {}^{\ulcorner}(r\cdot a\cdot r\cdot p) \;=\; p - {}^{\ulcorner}(r\cdot a\cdot p) \;=\; p - (r\cdot {}^{\ulcorner}(a\cdot p)) \;=$$
$$p - (r\cdot \langle a\rangle\, p) \;=\; (p - r) + (p - \langle a\rangle\, p) \;=\; 0 + a \rhd p \;=\; a \rhd p \;.$$

The following lemma collects useful properties of the maximality operator; proofs can be found in Appendix B.

**Lemma 4.2.** *Assume a type $T$ and $a, b :: T[r]^2$, $1_{T[r]} = r :: T$ and $p :: T[r]$, i.e., $p \leq r$. Then the following holds:*

1. $a \rhd r = \neg{}^{\ulcorner}a$.
2. ${}^{\ulcorner}b \leq {}^{\ulcorner}a \;\Leftrightarrow\; a \rhd r \leq b \rhd r$.
3. $a \rhd p \leq p$.
4. $a \rhd r \leq p \;\Leftrightarrow\; a \rhd r \leq a \rhd p$.
5. $a \rhd r \leq a \rhd (a \rhd r)$.
6. $a \rhd (a \rhd r) = a \rhd r$.
7. $(a + b) \rhd p = (a \rhd p) \cdot (b \rhd p)$.
8. $b \leq a \;\Rightarrow\; a \rhd p \leq b \rhd p$.
9. $r \leq a \;\Rightarrow\; a \rhd p = 0_T$.

*4.2. Basic Applications*

Now we want to show the maximality operator $\triangleright$ in action.

**Example 4.3.** Let $a :: T^2$ be a preference relation, $r$ a data set and suppose $p_1, p_2 :: T[r]$ are tests that form a disjoint decomposition of $1_T$. Assume that all elements in $p_2$ are better than all elements in $p_1$, i.e.,

$$\langle a \rangle \, p_2 = p_1 \ , \quad \langle a \rangle \, p_1 = 0_T \ .$$

We show that $p_2$ represents the maximal elements, i.e. $p_2 = a \triangleright r$:

$$
\begin{array}{ll}
 & a \triangleright r \\
= & \quad \{\![ \text{ Lemma 4.2.1 } ]\!\} \\
 & \neg \langle a \rangle \, r \\
= & \quad \{\![ \; p_1 + p_2 = r \; ]\!\} \\
 & \neg(\langle a \rangle \, (p_1 + p_2)) \\
= & \quad \{\![ \text{ distributivity of diamond } ]\!\} \\
 & \neg(\langle a \rangle \, p_1 + \langle a \rangle \, p_2) \\
= & \quad \{\![ \text{ assumptions on } a \; ]\!\} \\
 & \neg p_1 \\
= & \quad \{\![ \text{ Lemma 3.4 } ]\!\} \\
 & p_2 \ .
\end{array}
$$

□

By this tiny example one can see how the maximality operator works in general, because one can always decompose $r$ into tests representing the non-maximal ($p_1$) and the maximal ($p_2$) elements, where $p_1$ and $p_2$ are disjoint.

*4.3. Prefilters*

In practical applications, e.g., in databases, the tests, in particular the test $r$ representing all objects in the database, can be quite large. Hence it may be very expensive to compute $a \triangleright r$ for a given $a$. However, it can be less expensive to compute $b \triangleright r$ for another element $b$; ideally, that set is much smaller and the $a$-best objects overall coincide with the $a$-best objects within $b \triangleright r$. This motivates the following definition.

**Definition 4.4.** Assume $a, b :: T[r]^2$ and a data set $r :: T$. We call $b$ a *prefilter* for $a$, written as $b \,\mathsf{pref}\, a$, if and only if

$$a \triangleright r = a \triangleright (b \triangleright r) \ .$$

Note that no connection between $a$ and $b$ is assumed. By Lemma 4.2.6 we have $a \,\mathsf{pref}\, a$ for all $a$. A concrete example of a prefilter will be given in Section 5.1.

We can give another, calculationally useful, characterisation of prefilters. The proof of the following theorem can be found in Appendix B.3.

**Theorem 4.5.** $b \,\mathsf{pref}\, a \ \Leftrightarrow\ \ulcorner b \leq \ulcorner a \wedge \ulcorner a \leq \ulcorner b + \langle a \rangle \, \neg \ulcorner b$.

So far, we have not required any special properties of the elements $a$ that represent, e.g., preference relations. Instead of transitivity or irreflexivity we need an assumption that such elements admit "enough" maximal objects. This is expressed by requiring every non-maximal object to be dominated by some maximal one. In a setting with finitely many objects, such as a database, and a preference relation on them this property is always satisfied and hence is no undue restriction for our purposes. We defer a discussion of this assumption for infinite sets of objects, since there it is related to fundamental issues such as Zorn's Lemma and Hausdorff's maximality principle, hence to the axiom of choice. A brief treatment of the infinite case using the concept of noetherity, i.e., absence of infinite ascending chains, is given in Appendix E.

**Definition 4.6.** We call $a$ *normal* if

$$\forall p : \langle a \rangle\, p \le \langle a \rangle\, (a \triangleright p)\ .$$

By $a \triangleright p \le p$ and isotony of diamond this strengthens to

$$\forall p : \langle a \rangle\, p = \langle a \rangle\, (a \triangleright p)\ . \tag{3}$$

This is a compact algebraic formulation of the above domination requirement. Note that this is a stronger requirement than the normality definition in [20], in which we defined normality as the special case of the above definition with $p = 1$. For the following theorem that weaker notion would be sufficient.

First we show that any relation on a subset of the domain of a normal relation provides a prefilter.

**Theorem 4.7.** *Assume* $a, b :: T[r]^2$*.*

1. *Let* $a$ *be normal. Then* $\ulcorner b \le \ulcorner a \Rightarrow b\, \mathsf{pref}\, a$*.*
2. *Let* $a + b$ *be normal. Then* $a\, \mathsf{pref}\, (a + b)$*.*

Proof.

1. The assumption about $b$ is the first conjunct of the right hand side in Theorem 4.5. For the second conjunct we calculate

$$\begin{array}{ll} & \mathsf{TRUE} \\ \Leftrightarrow & \quad \{\!\![\ \text{normality of } a, \text{ setting } p = 1_T[r]\ ]\!\!\} \\ & \ulcorner a \le \langle a \rangle \neg \ulcorner a \\ \Rightarrow & \quad \{\!\![\ \ulcorner b \le \ulcorner a, \text{ contraposition and isotony of diamond}\ ]\!\!\} \\ & \ulcorner a \le \langle a \rangle \neg \ulcorner b \\ \Rightarrow & \quad \{\!\![\ x \le x + y \text{ and transitivity of } \le\ ]\!\!\} \\ & \ulcorner a \le \ulcorner b + \langle a \rangle \neg \ulcorner b\ . \end{array}$$

2. Since $a \le a + b$, isotony of diamond and hence of domain imply $\ulcorner a \le \ulcorner (a + b)$ and the claim follows from Part 1. □

Next we show that under certain conditions prefilters can be nested.

**Theorem 4.8.** *Assume* $a, b, c :: T[r]^2$*, where* $b \triangleright r \le c \triangleright r$ *and* $b\, \mathsf{pref}\, a$ *with normal* $a$*. Then also* $c\, \mathsf{pref}\, a$*.*

Proof. First, by Theorem 4.5 we have $\ulcorner b \le \ulcorner a \wedge \ulcorner a \le \ulcorner b + \langle a \rangle \neg \ulcorner b$. Second, by Lemma 4.2.1 and contraposition the assumption $b \triangleright r \le c \triangleright r$ is equivalent to $\ulcorner c \le \ulcorner b$. Hence by transitivity of $\le$ we infer $\ulcorner c \le \ulcorner a$. Now normality of $a$ and Theorem 4.7.1 show the claim.

*4.4. Related properties of maxima*

Using the definition of normality we can show further interesting properties of maximal elements.

**Theorem 4.9.** *Let* $a$ *be normal. Then*

$$a \triangleright (p + q) = a \triangleright (a \triangleright p + a \triangleright q)\ .$$

Proof. We first rewrite the left hand side:

$$\begin{array}{ll} & a \triangleright (p + q) \\ = & \quad \{\!\![\ \text{definition of } \triangleright\ ]\!\!\} \\ & (p + q) - \langle a \rangle\, (p + q) \\ = & \quad \{\!\![\ \text{distributivity of diamond}\ ]\!\!\} \end{array}$$

$$
\begin{array}{ll}
& (p+q) - (\langle a \rangle\, p + \langle a \rangle\, q) \\
= & \quad \{\!\![ \text{ De Morgan and distributivity of } \cdot \ ]\!\!\} \\
& (p - \langle a \rangle\, p - \langle a \rangle\, q) + (q - \langle a \rangle\, p - \langle a \rangle\, q) \\
= & \quad \{\!\![ \text{ right-commutativity of } - \text{ and definition of } \triangleright \ ]\!\!\} \\
& (a \triangleright p - \langle a \rangle\, q) + (a \triangleright q - \langle a \rangle\, p)\ .
\end{array}
$$

For the right hand side we obtain therefore

$$
\begin{array}{ll}
& a \triangleright (a \triangleright p + a \triangleright q) \\
= & \quad \{\!\![ \text{ above calculation } ]\!\!\} \\
& (a \triangleright (a \triangleright p) - \langle a \rangle\, (a \triangleright q)) + (a \triangleright (a \triangleright q) - \langle a \rangle\, (a \triangleright p)) \\
= & \quad \{\!\![ \text{ idempotence of } \triangleright \ ]\!\!\} \\
& (a \triangleright p - \langle a \rangle\, (a \triangleright q)) + (a \triangleright q - \langle a \rangle\, (a \triangleright p))\ .
\end{array}
$$

Since normality (3) immediately entails $a \triangleright p - \langle a \rangle\, q = a \triangleright p - \langle a \rangle\, (a \triangleright q)$ and $a \triangleright q - \langle a \rangle\, p = a \triangleright q - \langle a \rangle\, (a \triangleright p)$, we are done. □

This theorem paves the way for a distributed computation of maxima, as the calculations $a \triangleright p$ and $a \triangleright q$ are independent.

Next we deal with nested maxima.

**Theorem 4.10.** *Let $a, b$ preferences with $b \triangleright q \leq a \triangleright q$ for all tests $q$ and $b$ normal. Then for all tests $p$,*

1. $a \triangleright (b \triangleright p) = b \triangleright p$,
2. $b \triangleright (a \triangleright p) = b \triangleright p$.

PROOF.

1. "$\leq$": We have $a \triangleright (b \triangleright p) \leq b \triangleright p$ by Lemma 4.2.3.
   "$\geq$": We use the assumption for $q := b \triangleright p$. This yields $b \triangleright (b \triangleright p) \leq a \triangleright (b \triangleright p)$ and Lemma 4.2.6 shows the claim.
2. First we show "$\leq$":

$$
\begin{array}{ll}
& b \triangleright (a \triangleright p) \\
= & \quad \{\!\![ \text{ definitions } ]\!\!\} \\
& (p - \langle a \rangle\, p) - \langle b \rangle\, (a \triangleright p) \\
\leq & \quad \{\!\![ \text{ definition of } - \ ]\!\!\} \\
& p - \langle b \rangle\, (a \triangleright p) \\
\leq & \quad \{\!\![ \text{ assumption for } q := p \text{ and isotony of diamond } ]\!\!\} \\
& p - \langle b \rangle\, (b \triangleright p) \\
= & \quad \{\!\![\ b \text{ normal } ]\!\!\} \\
& p - \langle b \rangle\, p \\
= & b \triangleright p\ .
\end{array}
$$

   Then we show "$\geq$":

$$
\begin{array}{ll}
& b \triangleright (a \triangleright p) \\
= & \quad \{\!\![ \text{ definitions } ]\!\!\} \\
& (p - \langle a \rangle\, p) - \langle b \rangle\, (p - \langle a \rangle\, p) \\
\geq & \quad \{\!\![ \text{ isotony of diamond } ]\!\!\} \\
& (p - \langle a \rangle\, p) - \langle b \rangle\, p
\end{array}
$$

$$
\begin{aligned}
\geq\quad & \{\!\![\ \text{assumption for } q := p \text{ and isotony of diamond}\ ]\!\!\} \\
& (p - \langle b\rangle\, p) - \langle b\rangle\, p \\
=\quad & \{\!\![\ \text{tests are idempotent, definitions}\ ]\!\!\} \\
& b \rhd p\ .
\end{aligned}
$$

□

Note that $a \leq b$ is also sufficient for the above theorem, as Lemma 4.2.8 then implies $b \rhd q \leq a \rhd q$ for all $q$. Hence $a \leq b$ implies $a \,\mathsf{pref}\, b$ by Theorem 4.10.2, i.e., $a$ is a prefilter for $b$ in this case.

## 5. Complex Preferences

We have seen how some laws of single preference relations can be proved in point-free style in our algebra. Now we want to *compose* preferences into *complex preferences*. To this end we will introduce some special operators. The standard semiring operations like multiplication, addition and meet also lead to some kind of complex preferences, but they are rarely used in the typical application domain of preference algebra [10, 12]. Instead, the so-called *Prioritisation* and *Pareto composition* are the most important constructors for complex preferences.

### *5.1. Complex Preferences as Typed Relations*

To motivate our algebraic treatment we first repeat the definitions of these preference combinators in the concrete setting of typed relations [10].

For basic sets $M, N$ and preference relations $R \subseteq M^2, S \subseteq N^2$ the prioritisation $R \,\&\, S$ is defined as:

$$(x_1, x_2)\,(R \,\&\, S)\,(y_1, y_2) \;\Leftrightarrow_{df}\; x_1\, R\, y_1 \;\vee\; (x_1 = y_1 \;\wedge\; x_2\, S\, y_2)$$

where $x_i \in M, y_i \in N$. The Pareto preference is defined as:

$$
\begin{aligned}
(x_1, x_2)\,(R \otimes S)\,(y_1, y_2) \;\Leftrightarrow_{df}\; & x_1\, R\, y_1 \wedge (x_2\, S\, y_2 \vee x_2 = y_2) \;\vee \\
& x_2\, S\, y_2 \wedge (x_1\, R\, y_1 \vee x_1 = y_1)\ .
\end{aligned}
$$

In order theory the prioritisation is well-known as *lexicographic order*.

We now want to get rid of the point-wise notation in favour of operators on relations. The technique is mostly standard; we exemplify it for the prioritisation. We calculate, assuming first $M :: A, N :: B$ with distinct attribute names $A, B$,

$$
\begin{aligned}
& (x_1, x_2)\,(R \,\&\, S)\,(y_1, y_2) \\
\Leftrightarrow\quad & \{\!\![\ \text{definition}\ ]\!\!\} \\
& x_1\, R\, y_1 \;\vee\; (x_1 = y_1 \;\wedge\; x_2\, S\, y_2) \\
\Leftrightarrow\quad & \{\!\![\ \text{logic}\ ]\!\!\} \\
& (x_1\, R\, y_1 \;\wedge\; \mathsf{true}) \;\vee\; (x_1 = y_1 \;\wedge\; x_2\, S\, y_2) \\
\Leftrightarrow\quad & \{\!\![\ \text{definitions of } \top_B \text{ and } 1_A\ ]\!\!\} \\
& (x_1\, R\, y_1 \;\wedge\; x_2\, \top_B\, y_2) \;\vee\; (x_1\, 1_A\, y_1 \;\wedge\; x_2\, S\, y_2) \\
\Leftrightarrow\quad & \{\!\![\ \text{definition of cartesian product of relations}\ ]\!\!\} \\
& (x_1, x_2)\,(R \times \top_B)\,(y_1, y_2) \;\vee\; (x_1, x_2)\,(1_A \times S)\,(y_1, y_2) \\
\Leftrightarrow\quad & \{\!\![\ \text{definition of relational union}\ ]\!\!\} \\
& (x_1, x_2)\,((R \times \top_B) \cup (1_A \times S))\,(y_1, y_2)\ .
\end{aligned}
$$

A similar calculation can be done for the Pareto composition. Now we can write the point-free equations

$$
\begin{aligned}
R \,\&\, S &= (R \times \top_B) \cup (1_A \times S)\ , \\
R \otimes S &= (R \times (S \cup 1_B)) \cup ((R \cup 1_A) \times S)\ .
\end{aligned}
$$

This is close to an abstract algebraic formulation. However, since we want to cover also the case of non-disjoint, overlapping tuples, we will replace the Cartesian product $\times$ by the join $\bowtie$. To simplify notation we use the convention that a preference $x$ has type $T_x$, i.e. $a :: T_a^2, b :: T_b^2, \ldots$. We will keep these types "general" as long as possible. Only when a complement relative to a given data set $r :: T$ is needed, e.g., for the maxima $a \rhd r$, we will use explicit sub-types , e.g., $T_a[r]$.

In a join algebra, assume preferences $a :: T_a^2, b :: T_b^2$. We can rewrite the above point-free equation as

$$a \,\&\, b =_{df} a \bowtie \top_b + 1_a \bowtie b \ .$$

So "equal w.r.t. $a$" is represented by the identity $1_a$, while $\top_b$ expresses that one does not care about the $T_b$-part when the $T_a$-part already decides the overall relation.

However, it is questionable whether this always meets the user's expectation. If $a$ is a layered preference, then the incomparability relation $\overline{(a + a^{-1})}$ is an equivalence relation. Hence, if two tuples $t_1, t_2$ are incomparable w.r.t. a layered preference $a$ but $t_2$ is better than $t_1$ w.r.t. $b$, it is quite intuitive to say that $t_2$ is better than $t_1$ (and not incomparable with $t_1$) in $a \,\&\, b$.

Formally this is reflected by the *SV-semantics* of [11]. "SV" stands for "substitutable values" and means that a comparison between two tuples $t_1, t_2$ with respect to $a$ remains unchanged if $t_1$ is substituted for an SV-related $t_1'$. In relational notation, with $s_a$ being the SV relation and "$\equiv$" meaning logical equivalence of formulas, we have

$$\forall t_1, t_1', t_2 : \quad t_1 \, s_a \, t_1' \implies t_1 \, a \, t_2 \equiv t_1' \, a \, t_2 \ \wedge \ t_2 \, a \, t_1 \equiv t_2 \, a \, t_1' \ . \tag{4}$$

We give an algebraic characterisation of SV relations.

**Definition 5.1 (SV relation).** For a preference $a :: T_a^2$ we call $s_a :: T_a^2$ an *SV relation for a*, if $s_a$ fulfils the following properties:

1. The relation $s_a$ is an equivalence relation, i.e. $s_a$ is reflexive ($1_a \leq s_a$), symmetric ($s_a^{-1} = s_a$) and transitive ($s_a \cdot s_a \leq s_a$).
2. $s_a$ is compatible with $a$:
   (a) $s_a \cdot a \leq a$,
   (b) $a \cdot s_a \leq a$.

If the SV relation is not stated explicitly, then it is assumed to be the identity, i.e. we set $s_a = 1_a$.

**Corollary 5.2.** *Compatibility of $a :: T_a^2$ and $s_a :: T_a^2$ implies $s_a \sqcap a = 0_a$.*

Proof. We use the Dedekind rule

$$c \cdot d \sqcap e \leq (c \sqcap e \cdot d^{-1}) \cdot d \ ,$$

which follows from the Schröder rule and Huntington's Axiom, cf. [23].

$$\begin{array}{ll}
& s_a \sqcap a \\
= & \quad \{\![ \text{ neutrality of } 1 \ ]\!\} \\
& 1_a \cdot s_a \sqcap a \\
\leq & \quad \{\![ \text{ Dedekind rule } ]\!\} \\
& (1_a \sqcap a \cdot s_a^{-1}) \cdot s_a \\
= & \quad \{\![ \text{ symmetry of } s_a \ ]\!\} \\
& (1_a \sqcap a \cdot s_a) \cdot s_a \\
\leq & \quad \{\![ \text{ compatibility of } a \text{ with } s_a \text{, condition 2b) } ]\!\} \\
& (1_a \sqcap a) \cdot s_a \\
= & \quad \{\![ \text{ irreflexivity of } a \ ]\!\} \\
& 0_a \cdot s_a \\
= & \quad \{\![ \ 0 \text{ an annihilator } ]\!\} \\
& 0_a \ .
\end{array}$$

□

Using the concept of SV relations we will below adapt our definition of prioritisation. Next to the prioritisation preference there is another important complex preference constructor, called *Pareto composition*, which combines two preferences $a :: T_a^2, b :: T_b^2$. It is denoted as $a \otimes b :: (T_a \bowtie T_b)^2$ and has the intuitive meaning: "Better w.r.t. to $a$ in the $T_a$-part or $b$ in the $T_b$-part and not worse (i.e. equal or better) in the other part".

Using SV relations where the "equal w.r.t to $a$" is formalised by "$s_a$-related" we define the following complex preferences:

**Definition 5.3 (Prioritisation and Pareto composition with SV).** *Let $a :: T_a^2$ and $b :: T_b^2$ be preferences with associated SV relations $s_a :: T_a^2$ and $s_b :: T_b^2$. The* prioritisation with SV *is given by:*

$$\begin{aligned} a \,\&\, b &:: (T_a \bowtie T_b)^2 \ , \\ a \,\&\, b &=_{df} \ a \bowtie \top_b \,+\, s_a \bowtie b \ , \end{aligned}$$

*whereas the* Pareto compositions with SV *are defined as*

$$\begin{aligned} a \mathbin{<\!\!\!\otimes} b, a \mathbin{\otimes\!\!\!>} b, a \otimes b &:: (T_a \bowtie T_b)^2 \ , \\ a \mathbin{<\!\!\!\otimes} b &=_{df} \ a \bowtie (b + s_b) \ , \\ a \mathbin{\otimes\!\!\!>} b &=_{df} \ (a + s_a) \bowtie b \ , \\ a \otimes b &=_{df} \ a \mathbin{<\!\!\!\otimes} b + a \mathbin{\otimes\!\!\!>} b \ . \end{aligned}$$

*We say that $a {*} b$, for $* \in \{\&, \mathbin{<\!\!\!\otimes}, \mathbin{\otimes\!\!\!>}, \otimes\}$, is* SV-preserving *if $s_{a*b} = s_a \bowtie s_b$. Note that one may always define SV relations other than* 1 *for complex preferences, as long as they fulfil the conditions of Definition 5.1.*

**Corollary 5.4.** *The above notions are well-defined, i.e. $s_a \bowtie s_b$ is indeed a valid SV relation for $a * b$ with $* \in \{\&, \mathbin{<\!\!\!\otimes}, \mathbin{\otimes\!\!\!>}, \otimes\}$.*

Proof. Straightforward from distributivity and isotony of join and the exchange laws. □

Now we consider SV relations larger than $1_a$. For weak orders a typical SV relation is the incomparability relation, which we state in the following lemma:

**Lemma 5.5.** *If $a :: T^2$ is a layered preference then $s_a = \overline{a + a^{-1}}$ is an SV relation.*

Proof. The equivalence property for this relation is well known and $s_a \sqcap a = 0_a$ is clear. We show $s_a \cdot a \leq a$. By definition of $s_a$, the exchange law for complement and converse, and finally the infimum property we infer:

$$s_a \cdot a = \left(\overline{a} \sqcap \overline{a^{-1}}\right) \cdot a = \left(\overline{a} \sqcap (\overline{a})^{-1}\right) \cdot a \leq (\overline{a})^{-1} \cdot a \ .$$

We still have to show that $(\overline{a})^{-1} \cdot a \leq a$. By the Schröder equivalences, this is equivalent to $\overline{a} \cdot \overline{a} \leq \overline{a}$, which is just negative transitivity of $a$. For $a \cdot s_a \leq a$ an analogous argument holds; hence $s_a$ is compatible with $a$. □

Note, that this does not hold if $a$ is not a layered preference, which we show with the following example:

**Example 5.6.** *Assume preferences $a :: A^2, b :: B^2$ on attributes $A, B$ with $D_A = D_B = \{0, 1, 2\}$. The preferences $a$ and $b$ both are the $<$-relation on the natural numbers, i.e. we have $(0\,x\,1), (1\,x\,2)$ and $(0\,x\,2)$ for $x \in \{a, b\}$. Assume tuples $t_1 = (1, 2)$ and $t_2 = (2, 1)$. Then, by definition, $t_1$ and $t_2$ are not related w.r.t. to $(a \otimes b)$ or $1_A \bowtie 1_B$, i.e. they are incomparable. Now we consider the incomparability relation $s_{\mathrm{inc}} =_{df} \overline{(a \otimes b) + (a \otimes b)^{-1}}$. We have, by definition of $a \otimes b$,*

$$(2,0)\,(a \otimes b)\,(2,1), \ \ \neg((2,0)\,(a \otimes b)^k\,(1,2)), \ \ \neg((2,1)\,(a \otimes b)^k\,(1,2)) \ ,$$

*where $k \in \{-1, 1\}$. By definition of $s_{\mathrm{inc}}$ this implies*

$$\neg((2,0)\,s_{\mathrm{inc}}\,(2,1)), \ \ (2,0)\,s_{\mathrm{inc}}\,(1,2), \ \ (2,1)\,s_{\mathrm{inc}}\,(1,2) \ ,$$

*which means that $s_{\mathrm{inc}}$ is not transitive, hence no equivalence relation and therefore no SV relation for $a$.* □

**Remark 5.7.** *Under certain circumstances the term* $\langle \top_T \rangle\, q$ *occurring, for example, in* $\langle a \,\&\, b\rangle\, (p \bowtie q)$ *can be simplified. Call an idempotent semiring weakly Tarskian if for all types* $T$ *and tests* $q :: T$ *we have*

$$\langle \top_T \rangle\, q = \begin{cases} 1_T & \text{if } q \neq 0_T, \\ 0_T & \text{if } q = 0_T. \end{cases}$$

*For instance, the semiring of binary relations is weakly Tarskian. This implies that in a term like* $\langle a \bowtie \top_b \rangle\, (q_1 \bowtie q_2)$ *with* $a :: T_a^2$ *the test* $q_2$ *is irrelevant as long as* $q_2 \neq 0_b$*. This is exactly what we want, because* $q_1 \bowtie 0_b\ (= 0_{a \bowtie b})$ *is a zero element and must not have successors in any relation.*

*A semiring with* $\top$ *is called Tarskian when* $a \neq 0 \Rightarrow \top \cdot a \cdot \top = \top$*. This property was first stated for the semiring of binary relations (see, e.g., [23]). By the standard theory of diamond and domain [5], a Tarskian semiring is also weakly Tarskian, but generally not vice versa.*

In our car example from the introduction, the user would typically express her preference as the Pareto composition of minimal fuel consumption and highest power.

The definition of the Pareto compositions immediately yields important optimisation tools.

**Corollary 5.8.** *The preferences* $a \lessdot\!\!\otimes b$ *and* $a \otimes\!\!\gtrdot b$ *are prefilters for* $a \otimes b$*. Likewise,* $a \bowtie \top_B$ *is a prefilter for* $a \,\&\, b$*.*

Proof. By definition, $a \lessdot\!\!\otimes b, a \otimes\!\!\gtrdot b \leq a \otimes b$ and $a \bowtie \top_B \leq a \,\&\, b$; hence Theorem 4.7.1 applies. □

Hence, in our car example from the introduction, we may prefilter by the preference for fuel consumption or by the one for power to speed up the overall filtering. Further applications of this principle are discussed in detail in [6].

### *5.2. Maximality for Complex Preferences*

We first state the behaviour of the maximality operator for joins of preference elements.

**Lemma 5.9.** *For* $a :: T_a^2, p :: T_a$ *and* $b :: T_b^2, q :: T_b$ *with* $T_a \cap T_b = \emptyset$ *we have*

$$(a \bowtie b) \rhd (p \bowtie q) \;=\; (a \rhd p) \bowtie q + p \bowtie (b \rhd q)\ .$$

Proof. We observe that, under the disjointness assumption, by Corollary 2.8.3 and the axioms of the join algebra, for $r :: T_a, s :: T_b$, we have the join complement of tests (similar to Cartesian products):

$$\neg(p \bowtie q) = \neg p \bowtie q + p \bowtie \neg q + \neg p \bowtie \neg q\ .$$

This yields

$$(p \bowtie q) - (r \bowtie s) \;=\; (p - r) \bowtie q \;+\; p \bowtie (q - s)\ .$$

Hence, by the definitions and the distributivity of the diamond over $\bowtie$,

$$\begin{array}{rl} & (a \bowtie b) \rhd (p \bowtie q) \\ = & (p \bowtie q) - \langle a \bowtie b\rangle\, (p \bowtie q) \\ = & (p \bowtie q) - (\langle a\rangle\, p \bowtie \langle b\rangle\, q) \\ = & (p - \langle a\rangle\, p) \bowtie q + p \bowtie (q - \langle b\rangle\, q) \\ = & (a \rhd p) \bowtie q + p \bowtie (b \rhd q)\ . \end{array}$$

□

Since both prioritisation and Pareto composition are defined as sums of joins, we can now use this together with Lemma 4.2.7, 4.2.1 and the exchange axiom of Definition 3.6.5 to calculate their maximal elements.

**Lemma 5.10.** *For $a :: T_a^2, p :: T_a$ and $b :: T_b^2, q :: T_b$ with $T_a \cap T_b = \emptyset$ we have*

$$\begin{aligned}(a \lessdot\!\!\otimes b) \rhd (p \bowtie q) &= (a \rhd p) \bowtie q \ ,\\ (a \otimes\!\!\gtrdot b) \rhd (p \bowtie q) &= p \bowtie (b \rhd q) \ ,\\ (a \otimes b) \rhd (p \bowtie q) &= (a \rhd p) \bowtie (b \rhd q) \ ,\\ (a \,\&\, b) \rhd (p \bowtie q) &= (a \rhd p) \bowtie (b \rhd q) \ .\end{aligned}$$

The proofs are straightforward and hence omitted.

**Remark 5.11.** *It follows directly from the above lemma that*

$$(a \,\&\, b) \rhd (p \bowtie q) \;=\; (b \,\&\, a) \rhd (q \bowtie p) \;=\; (a \otimes b) \rhd (p \bowtie q)\,,$$

*i.e., Pareto composition and Prioritisation induce identical maxima sets on tests of the form $p \bowtie q$.*

*Note that this does not hold for general tests. Consider, for instance, the basic set $\{0,1\}^2$ and its subset $N =_{df} \{(0,1),(1,0)\}$, both represented by tests. Assume a preference order $R_i$ in the i-th component which fulfills $0\,R_i\,1$, for $i = 1,2$. Then $(R_1 \,\&\, R_2) \rhd N = \{(1,0)\}$, whereas $(R_1 \otimes R_2) \rhd N = N$. This does not contradict our above result, since $N$ cannot be represented in the form $L \times M$ with $L, M \subseteq \{0,1\}$.* □

*5.3. Equivalence of Preference Terms*

In the following we will show equalities of more complex preference terms, where both prioritization and Pareto composition are involved. In the following we assume that all complex preferences (Definition 5.3) are SV-preserving.

**Corollary 5.12.** *Let $a :: T_a^2$ and $b, b' :: T_b^2$. Then we have:*

$$a \,\&\, (b + b') = a \,\&\, b + a \,\&\, b' \ .$$

Proof. Follows from definition of &, idempotence of + and distributivity of $\bowtie$ over +. □

**Corollary 5.13.** *For $a :: T_a^2$ we have $a \lessdot\!\!\otimes a \;=\; a \otimes\!\!\gtrdot a \;=\; a \otimes a \;=\; a$.*

Proof. $a \lessdot\!\!\otimes a =_{df} (a + s_a) \bowtie a = (a + s_a) \sqcap a = a$ by Definitions 3.6.7 and 5.1.2. For Right Semi-Pareto and Pareto an analogous argument shows the claim.

**Theorem 5.14.** *For $a :: T_a$ we have that $(a\,\&)$, where & is SV-preserving, distributes over $\lessdot\!\!\otimes, \otimes\!\!\gtrdot$ and $\otimes$.*

Proof. Let $b :: T_b^2, c :: T_c^2$. We use the auxiliary equation (see Appendix C for a proof)

$$a \,\&\, b + s_{a\,\&\,b} = a \,\&\, (b + s_b) \ . \tag{5}$$

Now we calculate:

$$\begin{aligned}
& (a \,\&\, b) \otimes\!\!\gtrdot (a \,\&\, c)\\
=\;& \quad \{\!\![\text{ definition of } \otimes\!\!\gtrdot \,]\!\!\}\\
& (a \,\&\, b + s_{a\&b}) \bowtie (a \,\&\, c)\\
=\;& \quad \{\!\![\text{ equation (5) }]\!\!\}\\
& (a \,\&\, (b + s_b)) \bowtie (a \,\&\, c)\\
=\;& \quad \{\!\![\text{ definition of \& }]\!\!\}\\
& (a \bowtie \top_b + s_a \bowtie (b + s_b)) \bowtie (a \bowtie \top_c + s_a \bowtie c)\\
=\;& \quad \{\!\![\text{ distributivity of } \bowtie \,]\!\!\}\\
& a \bowtie \top_b \bowtie a \bowtie \top_c \quad + a \bowtie \top_b \bowtie s_a \bowtie c \;+\\
& s_a \bowtie (b + s_b) \bowtie a \bowtie \top_c \quad + s_a \bowtie (b + s_b) \bowtie s_a \bowtie c
\end{aligned}$$

$$
\begin{array}{ll}
= & \{\!\![\ a \bowtie a = a \text{ and } a \bowtie s_a = a \sqcap s_a, \text{ Definition 3.6.7 } ]\!\!\} \\
& a \bowtie \top_b \bowtie \top_c \qquad\qquad + (a \sqcap s_a) \bowtie \top_b \bowtie c\, + \\
& (a \sqcap s_a) \bowtie (b + s_b) \bowtie \top_c \quad + s_a \bowtie (b + s_b) \bowtie c \\
= & \{\!\![\ a \sqcap s_a \leq a,\ c \leq \top_c, \text{ subsumption order } ]\!\!\} \\
& a \bowtie \top_b \bowtie \top_c + s_a \bowtie (b + s_b) \bowtie c \\
= & \{\!\![\ \top_{b \bowtie c} = \top_b \bowtie \top_c, \text{ definition of } \&\ ]\!\!\} \\
& a \,\&\, ((b + s_b) \bowtie c) \\
= & \{\!\![\ \text{definition } \otimes\!\!> \ ]\!\!\} \\
& a \,\&\, (b \otimes\!\!> c)\ .
\end{array}
$$

A symmetric argument holds for $<\!\!\otimes$, so that $(a\,\&)$ distributes over $<\!\!\otimes$ and $\otimes\!\!>$. Using this we infer the distributivity over $\otimes$, see Appendix C for details. □

The proof of this theorem shows that the framework of typed relations is rich enough to prove non-trivial preference term equivalences.

We have proved this theorem using Prover9. The input for the auxiliary equation (5) can be found in Appendix A and the input for the entire theorem is given in [19]. The most important ingredients for these theorems are the axioms of our join algebra. The sub-theory of abstract relation algebra is quite simple and mainly consists of choice and subsumption order. Automated proofs with Prover9 for more complex relation algebras, but not for join algebras, are considered in [8].

Such equivalences are useful for an optimized evaluation of preferences, because the evaluation of an equivalent term may be faster.

## 6. Layered Preferences

Layered Preferences as defined in Definition 3.7 are not closed under the Pareto operator. Therefore the question arises whether there is a similar operator under which layered preferences are closed. The answer is yes and our strategy to construct such a preference is as follows:

- Let $r$ be a test representing the *basic set*, i.e. the data set. We take the maxima of $r$ w.r.t. $(a \otimes b)$, i.e. $(a \otimes b) \triangleright r$, and call them layer-0 elements.
- We remove them from the basic set, i.e. we define $r_1 = r - (a \otimes b) \triangleright r$, take their maxima and call these layer-1 elements.
- We iterate this process to obtain the layer-$n$ elements for $n = 2, 3, ....$
- We define a new preference induced by ordering the elements according to their layers, placing layer 0 at the top and the layer with the largest number at the bottom. This yields a layered preference by construction.

We will call the new preference the *Pareto-regular* preference. This name stems from the implementation of preferences in Preference SQL [12]: The keyword `regular` after a layered preference means that the SV relation from Lemma 5.5 is applied. As the Pareto preference need not be a layered preference, we call "Pareto-regular" the result of transforming the Pareto preference into a layered preference and then applying Lemma 5.5.

The process of successive removal of maximal elements corresponds to the repeated removal of sinks in a classical algorithm for cycle detection in directed graphs [17, 23]. That algorithm terminates when the set of maxima/sinks to be removed becomes empty. The original graph contained a cycle iff the remaining set is nonempty. Since our preference relations, as strict partial orders, correspond to acyclic graphs, the iteration necessarily will reach the empty set, which, however, is not counted as a separate layer. We will make this more precise in the next section.

*6.1. Computing Layer-i Elements*

The concept of layer-$i$ elements was originally introduced under the term "Iterated preferences" in [4], where the maximum operator is called "winnow operator". Here we give an algebraic definition, and prove some properties. In particular, we show that the induced order is indeed a layered preference.

**Definition 6.1 (Layer-$i$ Elements).** *Let $a :: T_a^2$ be a preference, and $r :: T_a$ a basic set. For $i = 0, 1, 2, \ldots$ we define the tests $q_i$ and $r_i$ characterising the layer-i elements and the remainders, respectively:*

$$q_i =_{df} a \rhd r_i \text{ where } r_i =_{df} r - \sum_{j=0}^{i-1} q_j \ .$$

*By convention, the empty sum is $0_a$, hence we have $r_0 = r$.*

A mnemonic for the $q_i$ is that the letter "b" for "best", rotated by 180° becomes a "q". This matches our convention that $a, b, c, \ldots$ are used for general elements and $p, q, r$ for tests.

Using our algebra we deduce a closed formula for the $r_i$. In this, we write $ra$ short for $r \cdot a$. The powers $x^k$ are defined by $x^k = x^{k-1} \cdot x$ and $x^0 = 1$. The proof of the following lemma can be found in Appendix D.1.

**Lemma 6.2 (Closed formula for layer-$i$ elements).** *For $i \in \mathbb{N}$ we have the following properties:*

1. $(ra)^{i+1} \leq (ra)^i$ *provided* $i > 0$,
2. $\langle (ra)^{i+1} \rangle r \leq \langle (ra)^i \rangle r$,
3. $r_i = \langle (ra)^i \rangle r$.

With this lemma we have a compact representation of layer-$n$ elements. This helps us to show some interesting properties of $r_i$ and $q_i$ which we will need for the construction of the Pareto-regular preference later on. These properties are stated in the following lemma, which is proved in Appendix F.

**Lemma 6.3.** *Assume $q_i, r_j$ as in Definition 6.1. We have:*

1. *The $r_i$ are decreasing in $i$, i.e., $r_0 \geq r_1 \geq r_2 \geq \ldots$.*
2. *The $q_i$ are pairwise disjoint, i.e., for $i \neq j$ we have $q_i \cdot q_j = 0_a$.*
3. *Let $r$ be finite, i.e., assume that there do not exist infinitely many disjoint $p_i \neq 0$ with $\sum_i p_i = r$. Then the calculation of the $r_i$ becomes stationary, i.e. there exists an $N \in \mathbb{N}$ with $N = \max\{k \in \mathbb{N} \mid r_k \neq 0_a\}$.*
4. *The $q_i$ cover $r$, i.e., $\sum_{i=0}^{N} q_i = r$.*
5. *For $i \leq j$ we have $q_i \cdot a \cdot q_j = 0_a$.*

*6.2. The Induced Layered Preference*

Now we will construct the preference induced by the layer-$i$ elements and the corresponding induced SV relation.

**Definition 6.4 (Induced layered preference).** *Let $a :: T_a^2$ be a preference and $r :: T_a$ a basic set. Consider the corresponding layer-i elements $q_i = a \rhd \langle (ra)^i \rangle r$ with $i \in [1, N]$ and $N = \max\{k \in \mathbb{N} \mid r_k \neq 0_a\}$ (see Lemma 6.3.3). We define relations $b_{ij}$ ($i, j \in [1, N]$) by $b_{ij} = q_i \cdot \top_a \cdot q_j$. In the concrete model these represent universal relations between the sets $q_i$ and $q_j$. With their help, the* induced layered preference *$m(a, r) :: T_a[r]^2$ is defined as*

$$m(a, r) =_{df} \sum_{i > j} b_{ij} \ ,$$

*where $T_a[r]$ is the sub-type of $T_a$ with identity $r$ and greatest element $r \cdot \top_a \cdot r$.*

*By the summation over $i > j$ the less preferred elements w.r.t. to $a$ (with higher layer numbers) are $m(a, r)$-related to the more preferred elements (with lower layer numbers).*

*A corresponding SV relation $s_{m(a,r)} :: T_a[r]^2$ is defined as*

$$s_{m(a,r)} =_{df} \sum_i b_{ii} \ .$$

We note an important property of the relations $b_{ij}$: by disjointness of the $q_i$ and the Tarski rule we have

$$b_{ij} \cdot b_{kl} = \begin{cases} b_{il} & \text{if } j = k, \\ 0_a & \text{otherwise.} \end{cases} \tag{6}$$

Our goal is to construct the Pareto-regular preference, and we are close to this by constructing $m(a, r)$ for a Pareto preference $a = a_1 \otimes a_2$. But is the resulting relation fulfilling the (layered) preference properties, i.e. would such a preference be well-defined? We show this in the next lemma.

**Lemma 6.5.**

1. *The relation $m(a, r)$ from the previous definition is a layered preference.*
2. *$s_{m(a,r)}$ is an SV relation for $m(a, r)$.*

PROOF.

1. Transitivity follows from the definition of $m(a, r)$ and Eq. (6). Again by definition of $m(a, r)$ and disjointness of $q_i$ (Lemma 6.3.2) we have irreflexivity. It remains to show that negative transitivity holds. Note that due to the type $T_a[r]^2$ of $m(a, r)$ and $s_{m(a,r)}$ the complement $\overline{(...)}$ is relative to $r$. With this we infer $(\overline{m(a, r)})^2 \leq \overline{m(a, r)}$:

$$\left(\overline{m(a, r)}\right)^2 = \left(\sum_{i \leq j} b_{ij}\right) \cdot \left(\sum_{k \leq l} b_{kl}\right) = \sum_{i \leq j \leq l} b_{ij} \cdot b_{jl} \leq \sum_{i \leq l} b_{il} = \overline{m(a, r)} \ .$$

2. We infer that

$$\overline{m(a, r) + m(a, r)^{-1}} = \overline{\sum_{i > j} b_{ij} + \sum_{i < j} b_{ij}} = \overline{\sum_{i \neq j} b_{ij}} = \sum_{i} b_{ii} = s_{m(a,r)} \ .$$

Together with Lemma 5.5 this shows the claim. □

We give two further useful properties of the induced layered preference:

- The original preference is still contained in the induced layered preference.
- The induced SV relation is part of the incomparability relation of the original preference.

Formally this is stated in the following lemma. There again we restrict the "original preference" and the "incomparability relation" to $r$ on both sides, because the induced layered preference and the induced SV relation is only defined on the basic set $r$.

**Lemma 6.6.** *Let $a :: T_a^2$ be a preference and $r :: T_a$ a basic set. We have:*

1. *$r \cdot a \cdot r \leq m(a, r)$.*
2. *$s_{m(a,r)} \leq r \cdot \overline{(a + a^{-1})} \cdot r$.*

PROOF.

1. By Lemma 6.3.5 we get $q_i \cdot a \cdot q_j = 0$ for $i \leq j$. This implies:

$$\sum_{i \leq j} q_i \cdot a \cdot q_j = 0_a \ . \tag{7}$$

We use this in the following deduction:

TRUE

$\Leftrightarrow$ $\{\!\!\{$ definition $\top_a$ $\}\!\!\}$

$a \leq \top_a$

$\Rightarrow$ $\{\!\!\{$ $q_j \cdot (...)$, $(...) \cdot q_i$, summation over $i > j$ $\}\!\!\}$

$\sum_{i>j} q_i \cdot a \cdot q_j \leq \sum_{i>j} q_i \cdot \top_a \cdot q_j$

$\Leftrightarrow$ $\{\!\!\{$ Eq. (7) (additional term is 0), def. of $b_{ij}$ $\}\!\!\}$

$\sum_{i>j} q_i \cdot a \cdot q_j + \sum_{i \leq j} q_i \cdot a \cdot q_j \leq \sum_{i>j} b_{ij}$

$\Leftrightarrow$ $\{\!\!\{$ re-indexing of sum, def. of $m(a, r)$ $\}\!\!\}$

$\sum_{i,j} q_i \cdot a \cdot q_j \leq m(a, r)$

$\Leftrightarrow$ $\{\!\!\{$ distributivity and $\sum_i q_i = r$ (Lemma 6.3.4) $\}\!\!\}$

$r \cdot a \cdot r \leq m(a, r)$ .

2. The claim is equivalent to

$$s_{m(a,r)} \sqcap (r \cdot a^k \cdot r) = 0_a \text{ for } k \in \{-1, 1\} .$$

From Part 1 we obtain $r \cdot a^k \cdot r \leq m(a^k, r)$ for $k \in \{-1, 1\}$, because $a^{-1}$ is again a preference, hence the same argument holds for it. Thus it is sufficient to prove

$$s_{m(a,r)} \sqcap m(a^k, r) = 0_a \text{ for } k \in \{-1, 1\} .$$

This follows from the definitions of $s_{m(a,r)}$ and $m(a^k, r)$ and the disjointness of the $q_i$ (Lemma 6.3.2). □

**Remark 6.7.** *The inequations in the previous lemma are equations if $a$ is already a layered preference.*

Now we have everything ready to define the Pareto-regular preference. All we have to do is to apply $m(...)$ to the classic Pareto preference, which yields a well defined result by the previous lemma.

**Definition 6.8 (Pareto-regular preference).** *For preferences $a :: T_a^2$, $b :: T_b^2$ and a basic set $r :: T_a \bowtie T_b$, the Pareto-regular preference and its SV relation are defined as:*

$$a \otimes_{\text{reg}} b :: (T_a \bowtie T_b)^2 \ ,$$
$$a \otimes_{\text{reg}} b = m(a \otimes b, r) \ ,$$
$$s_{a \otimes_{\text{reg}} b} = s_{m(a \otimes b, r)} \ .$$

Note that the Pareto-regular preference depends on the concrete basic set, i.e. the data stored in the database table. This is a fundamental difference to the classical preferences, which are independent of the data. But Lemma 6.6 tells us that this is kind of "harmless".

*6.3. Application: Pareto(-regular) and Prioritisation*

With the Pareto-regular preference we have a layered preference which is quite similar to the classic Pareto preference (which is not layered in general). But this is primarily a technical feature, which at first sight changes nothing, because the maxima set of a Pareto preference is the same as that for the associated Pareto-regular preference. For preferences $a :: T_a^2, b :: T_b^2$ and a basic set $r :: T_a \bowtie T_b$ we get immediately from the definitions that $(a \otimes b) \triangleright r = (a \otimes_{\text{reg}} b) \triangleright r$.

So where is the *practical* difference between $\otimes$ and $\otimes_{\text{reg}}$? The effect of the latter becomes evident in combination with other complex preferences, especially if the Pareto-regular preference is the first part of a prioritisation. We consider the following example:

**Example 6.9.** *Let* $a :: A^2, b :: B^2, c :: C^2$ *be preferences on attributes* $A, B, C$*. Let their type domains be* $D_A = D_B = D_C = \{1, 2\}$*. Let* $a, b, c$ *each be the* $<$*-order, i.e. we have* $(1\,x\,2)$ *for* $x \in \{a, b, c\}$*. Consider the basic set* $r :: A \bowtie B \bowtie C$ *given by* $t_1 =_{df} (1, 2, 1),\ \ t_2 =_{df} (2, 1, 2),\ r =_{df} t_1 + t_2$*, and the preferences*

$$d_1 =_{df} (a \otimes_{\text{reg}} b) \,\&\, c, \quad d_2 =_{df} (a \otimes b) \,\&\, c \ .$$

*For* $(a \otimes b)$ *we have only one maxima set* $q_0 = (a \otimes b) \triangleright r = r$*. Hence we get:*

$$t_1\, d_1\, t_2 \ , \quad \neg(t_1\, d_2\, t_2) \ .$$

*This means that the preference* $c$ *decides about the maxima of* $d_i$ *only if the previous Pareto preference* $a \otimes b$ *has no incomparable elements. This incomparability is avoided by construction in the Pareto-regular preference. Probably the preference* $d_1$ *is what the user expected, or at least expected more than* $d_2$*.* □

This is the abstract formulation of the introductory example on a car database (Examples 1.1–1.3). To reproduce the introductory example in terms of our algebraic calculus we define $a =_{df}$ LOWEST(*fuel*), $b =_{df}$ HIGHEST(*power*), $c =_{df}$ POS(*color*, {black}) and set the tuples $t_1$ and $t_2$ to the fuel/power/color values of "BMW 5" ($t_1$) and "Mercedes E" ($t_2$) as denoted in table 1 on page 2. Then the result is that "Mercedes E" is the best object according to $d_1 = (a \otimes_{\text{reg}} b) \,\&\, c$, while both cars are returned for $d_2 =_{df} (a \otimes b) \,\&\, c$.

Unfortunately the $\otimes_{\text{reg}}$ operator is not associative, i.e., in general we have $(a \otimes_{\text{reg}} b) \otimes_{\text{reg}} c \neq a \otimes_{\text{reg}} (b \otimes_{\text{reg}} c)$. Assume that a user wants to compose three equally important preferences, followed by a prioritization, i.e., $(a \otimes b \otimes c) \,\&\, d$. We think that the most intuitive way to construct a layered preference from this is to define a modified prioritisation operator that incorporates the $m$ function. We call this the *regularised prioritisation.*

**Definition 6.10 (Regularised prioritisation).** *For preferences* $a :: T_a^2$*,* $b :: T_b^2$ *and a basic set* $r :: T_a \bowtie T_b$ *the regularised prioritisation is defined as:*

$$\begin{aligned} a \,\&_{\text{reg}}\, b &:: (T_a \bowtie T_b)^2 \ , \\ a \,\&_{\text{reg}}\, b &= m(a, r) \,\&\, b \ . \end{aligned}$$

*An explicit SV relation is not specified for* $\&_{\text{reg}}$*. According to Definition 5.3 we call it SV-preserving if* $s_{a \&_{\text{reg}} b} = s_a \bowtie s_b$*. As this does not respect the layered preference* $m(a, r)$*, we call it* layered-SV-preserving *if* $s_{a \&_{\text{reg}} b} = s_{m(a)} \bowtie s_b$*.*

**Corollary 6.11.** *Pareto-regular is a special case of the regularized prioritisation; we have for preferences* $a, b, c$*:*

$$(a \otimes_{\text{reg}} b) \,\&\, b = (a \otimes b) \,\&_{\text{reg}}\, b \ .$$

## 7. Conclusion and Outlook

The present work intends to advance the state of the art in formalising preference algebra. Besides the point-wise "semi-formal" proofs by hand that had been used originally we wanted to use automatic theorem provers like PROVER9 to get the theorems of preference algebra machine-checked. But we realized that there was no straightforward way to put theorems like the prefilter properties or the distributive law for Prioritisation/Pareto into a theorem prover. Especially for the latter problem, the main reason is that originally the equivalence of preference terms was defined, e.g. in [10], in a very implicit manner: two preference terms are equivalent if and only if the corresponding relations are identical on the basic set. This definition is not very useful if one tries to find (automatically) general equivalence proofs.

The presented concept of a typed join algebra makes it possible to define such equivalences explicitly: two preference terms are identical, if and only if their algebraic representations are equal in the algebra.

The search for another kind of Pareto preference was originally motivated by a past project where Preference SQL was used for context-aware suggestions in a hiking-tour recommender. In [21] the context model is described and some sample queries are given; these are very complex preference constructs, where base preferences or Pareto compositions are put into long prioritisation chains. Within the project we noticed that the less-prioritized preferences, like "$c$" in Example 6.9, are not decisive for the maxima set, because a Pareto preference at the beginning of the term (like "$a \otimes b$" in the example) generates incomparable elements, hence the set of maxima cannot be reduced by $c$. With the Pareto-regular preference these prioritisation chains become chains of "filters" where the set of maxima can be reduced by adding less-prioritized preferences at the end of a chain. By calculating not only the maximal elements for the Pareto preference but the layer-$i$ elements, we are also able to answer *TOP-k* queries, i.e. a query like "What are the 10 best elements according to preference $a$?"

For this work we have extended our algebraic calculus with layered preferences and SV relations. Interesting connections and properties can be stated and proved algebraically, i.e., in a point-free fashion. These methods are important for a better understanding of preferences and constructing algorithms for preference evaluation.

For future research, there is a large number of theorems about preferences which have only been proved in a point-wise fashion, which makes the proofs hardly readable. These theorems are important for optimizing the algorithms for evaluating preference queries. In addition, we are also working on methods for parallel computation of the maxima according to preferences; hopefully algebraic methods will support us, e.g., the methods of Concurrent Kleene Algebra.

## References


[1] Backhouse, R., van der Woude, J., 11 1993. Demonic operators and monotype factors. Mathematical Structures in Computer Science 3, 417–433.

[2] Bird, R., de Moor, O., Sep. 1996. Algebra of Programming, The. Prentice Hall PTR.

[3] Borzsony, S., Kossmann, D., Stocker, K., 2001. The skyline operator. In: Data Engineering, 2001. Proceedings. 17th International Conference on. pp. 421–430.

[4] Chomicki, J., 2003. Preference Formulas in Relational Queries. In: TODS '03: ACM Transactions on Database Systems. Vol. 28. ACM Press, New York, NY, USA, pp. 427–466.

[5] Desharnais, J., Möller, B., Struth, G., Oct. 2006. Kleene algebra with domain. ACM Trans. Comput. Logic 7 (4), 798–833.

[6] Endres, M., 2011. Semi-Skylines und Skyline Snippets - Theory and Applications. Fakultät für Angewandte Informatik, Universität Augsburg, Dissertation. Books on Demand GmbH, Norderstedt.

[7] Fishburn, P. C., 1970. Utility theory for decision making. Tech. rep., New York, NY, USA.

[8] Höfner, P., Struth, G., 2008. On automating the calculus of relations. In: Automated Reasoning. Springer, pp. 50–66.

[9] Kanellakis, P. C., 1989. Elements of relational database theory. Tech. rep., Providence, RI, USA.

[10] Kießling, W., 2002. Foundations of Preferences in Database Systems. In: VLDB '02: Proceedings of the 28th International Conference on Very Large Data Bases. VLDB, Hong Kong, China, pp. 311–322.

[11] Kießling, W., 2005. Preference Queries with SV-Semantics. In: Haritsa, J. R., Vijayaraman, T. M. (Eds.), COMAD '05: Advances in Data Management 2005, Proceedings of the 11th International Conference on Management of Data. Computer Society of India, Goa, India, pp. 15–26.

[12] Kießling, W., Endres, M., Wenzel, F., 2011. The Preference SQL System - An Overview. Bulletin of the Technical Commitee on Data Engineering, IEEE Computer Society 34 (2), 11–18.

[13] Kozen, D., 1998. Typed Kleene algebra. Tech. Rep. TR98-1669, Computer Science Department, Cornell University.

[14] MacCaull, W., Orłowska, E., 2004. A calculus of typed relations. In: Berghammer, R., Möller, B., Struth, G. (Eds.), Relational and Kleene-Algebraic Methods in Computer Science. Vol. 3051 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 191–201.

[15] Maddux, R. D., 1997. Relation Algebras. Advances in Computing. Springer-Verlag, Wien, New York, Ch. 2, pp. 22–38.

[16] Manes, E., Benson, D., 1985. The inverse semigroup of a sum-ordered semiring. Semigroup Forum 31 (1), 129–152.

[17] Marimont, R. B., Apr. 1959. A new method of checking the consistency of precedence matrices. J. ACM 6 (2), 164–171.

[18] Möller, B., Roocks, P., 2012. An algebra of layered complex preferences. In: Kahl, W., Griffin, T. (Eds.), Relational and Algebraic Methods in Computer Science. Vol. 7560 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 294–309.

[19] Möller, B., Roocks, P., 2012. Proof of the Distributive Law for Prioritisation and Pareto Composition. `http://tinyurl.com/k27jpgh`.

[20] Möller, B., Roocks, P., Endres, M., 2012. An Algebraic Calculus of Database Preferences. In: Gibbons, J., Nogueira, P. (Eds.), Mathematics of Program Construction. Vol. 7342 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 241–262.

[21] Roocks, P., Endres, M., Mandl, S., Kießling, W., 2012. Composition and Efficient Evaluation of Context-Aware Preference Queries. In: DASFAA '12: Proceedings of the 17th international conference on Database systems for advanced applications.

[22] Schmidt, G., Hattensperger, C., Winter, M., 1997. Heterogeneous relation algebra. In: Brink, C., Kahl, W., Schmidt, G. (Eds.), Relational Methods in Computer Science. Advances in Computing Sciences. Springer Vienna, pp. 39–53.

[23] Schmidt, G., Ströhlein, T., 1993. Relations and Graphs: Discrete Mathematics for Computer Scientists. EATCS Monographs on Theoretical Computer Science.

## Appendix A. Sample Prover Input

For $a :: T_a^2$ and $b :: T_b^2$ we show the auxiliary equation (5) from Theorem 5.14:

$$a \,\&\, b + s_{a \bowtie b} = a \,\&\, (b + s_b) \ .$$

We use the following operators:

| Prover-Input | mathematically |
|---|---|
| `a typed T_a` | $a :: T_a^2$ |
| `a join b` | $a \bowtie b$ |
| `T_a tjoin T_b` | $T_a \bowtie T_b$ |
| `a prior b` | $a \,\&\, b$ |
| `a + b` | $a + b$ |

The assumptions are given as follows:

```
% all elements are typed
exists T (x typed T).

% addition is associative
(x + y) + z = x + (y + z).

% addition preserves type
x typed z & y typed z -> (x+y) typed z.

% SV relations preserve type
x typed z -> sv(x) typed z.

% abbreviated typing
x typed z -> top(x) = top(z).

% distributivity of the join over addition
x join (y1 + y2) = x join y1 + x join y2.

% typing of join
x typed z1 & y typed z2 -> (x join y) typed (z1 tjoin z2).

% prioritisation (without resulting type)
x prior y = x join top(y) + sv(x) join y.

% prioritisation is sv-preserving
sv(x prior y) = sv(x) join sv(y).
```

Finally our goal is:

```
% auxiliary equation for distributive law
u prior v + sv(u prior v) = u prior (v + sv(v)).
```

The entire input for the proof of theorem 5.14 can be found in [18].

## Appendix B. Proofs for Section 4

For the sake of readability we define for the proofs in this section: $1 =_{df} 1_{T[r]} = r$.

*Appendix B.1. Proof of Lemma 4.2*

1. Immediate from the definitions and $1 - p = \neg p$.
2. Immediate from Part 1 and contraposition.
3. $$\begin{aligned} & a \triangleright p \\ = \quad & \{\!\![ \text{ definitions of } \triangleright \text{ and } - \ ]\!\!\} \\ & p \cdot \neg \langle a \rangle \, p \\ \leq \quad & \{\!\![ \text{ property of intersection } ]\!\!\} \\ & p \ . \end{aligned}$$
4. $$\begin{aligned} & a \triangleright 1 \leq a \triangleright p \\ \Leftrightarrow \quad & \{\!\![ \text{ definition of } \triangleright \text{ and Part 1 } ]\!\!\} \\ & \neg \ulcorner a \leq p - \langle a \rangle \, p \\ \Leftrightarrow \quad & \{\!\![ \text{ definition of } - \text{ and universal property of intersection } ]\!\!\} \\ & \neg \ulcorner a \leq p \quad \wedge \quad \neg \ulcorner a \leq \neg \langle a \rangle \, p \\ \Leftrightarrow \quad & \{\!\![ \text{ contraposition in second conjunct } ]\!\!\} \\ & \neg \ulcorner a \leq p \quad \wedge \quad \langle a \rangle \, p \leq \ulcorner a \\ \Leftrightarrow \quad & \{\!\![ \text{ second conjunct true by (1) on page 7 } ]\!\!\} \\ & \neg \ulcorner a \leq p \\ \Leftrightarrow \quad & \{\!\![ \text{ Part 1 } ]\!\!\} \\ & a \triangleright 1 \leq p \ . \end{aligned}$$
5. Immediate from the previous property by setting $p = a \triangleright 1$.
6. $$\begin{aligned} & a \triangleright (a \triangleright p) \\ = \quad & \{\!\![ \text{ definition of } \triangleright \ ]\!\!\} \\ & (p - \langle a \rangle \, p) - \langle a \rangle \, (p - \langle a \rangle) \\ = \quad & \{\!\![ \text{ property of difference } ]\!\!\} \\ & p - (\langle a \rangle \, p + \langle a \rangle \, (p - \langle a \rangle)) \\ = \quad & \{\!\![ \text{ distributivity of } \langle\rangle \ ]\!\!\} \\ & p - \langle a \rangle \, (p + (p - \langle a \rangle)) \\ = \quad & \{\!\![ \text{ since } p - \langle a \rangle \leq p \ ]\!\!\} \\ & p - \langle a \rangle \, p \\ = \quad & \{\!\![ \text{ definition of } \triangleright \ ]\!\!\} \\ & a \triangleright p \ . \end{aligned}$$
7. $$\begin{aligned} & (a + b) \triangleright p \\ = \quad & \{\!\![ \text{ definition of } \triangleright \ ]\!\!\} \\ & p - \langle a + b \rangle \, p \\ = \quad & \{\!\![ \text{ distributivity of } \langle\rangle \ ]\!\!\} \\ & p - (\langle a \rangle \, p + \langle b \rangle \, p) \\ = \quad & \{\!\![ \text{ property of difference } ]\!\!\} \\ & (p - \langle a \rangle \, p) \cdot (p - \langle b \rangle \, p) \\ = \quad & \{\!\![ \text{ definition of } \triangleright \ ]\!\!\} \\ & (a \triangleright p) \cdot (b \triangleright p) \ . \end{aligned}$$
8. Assume $b \leq a$, i.e., $b + a = a$.

$$\begin{aligned} & a \triangleright p \\ = \quad & \{\!\![ \text{ assumption } ]\!\!\} \end{aligned}$$

$$
\begin{aligned}
& (b+a) \triangleright p \\
= \quad & \{\!\![ \text{ previous property } ]\!\!\} \\
& (b \triangleright p) \cdot (a \triangleright p) \\
\le \quad & \{\!\![ \text{ property of intersection } ]\!\!\} \\
& b \triangleright p \ .
\end{aligned}
$$

9. By isotony of the diamond we have $p = \langle 1 \rangle\, p \le \langle a \rangle\, p$ and hence $a \triangleright p = p - \langle a \rangle\, p = 0$. □

*Appendix B.2. Proof of Lemma 3.4*

We only show the first equation; the proof of the second one is symmetric.
First, from $r = p + q$ we infer $p \le r$. Moreover, by shunting, $p \cdot q \le 0 \Leftrightarrow p \le \neg q$. Together, we obtain $p \le r \cdot \neg q = r - q$.
Second, $r = p + q$ implies $r \le q + p$, and shunting shows $r - q \le p$. □

*Appendix B.3. Proof of Theorem 4.5*

We split the left-hand side of the claim equivalently into

$$b \,\mathsf{pref}\, a \ \Leftrightarrow\ a \triangleright 1 \le a \triangleright (b \triangleright 1) \ \wedge\ a \triangleright (b \triangleright 1) \le a \triangleright 1 \ .$$

By Parts 4 and 2 of Lemma 4.2 the first conjunct is equivalent to $\ulcorner b \le \ulcorner a$. For the second conjunct we calculate

$$
\begin{aligned}
& a \triangleright (b \triangleright 1) \le a \triangleright 1 \\
\Leftrightarrow \quad & \{\!\![ \text{ definition of } \triangleright \text{ and Lemma 4.2.1 } ]\!\!\} \\
& \neg \ulcorner b - \langle a \rangle\, \neg \ulcorner b \le \neg \ulcorner a \\
\Leftrightarrow \quad & \{\!\![ \text{ contraposition and De Morgan } ]\!\!\} \\
& \ulcorner a \le \ulcorner b + \langle a \rangle\, \neg \ulcorner b \ .
\end{aligned}
$$

□

## Appendix C. Proof of Theorem 5.14

Auxiliary equation (5):

$$
\begin{aligned}
& a \,\&\, b + s_{a \& b} \\
= \quad & \{\!\![ \text{ definition of } \& \text{ (SV-preserving) } ]\!\!\} \\
& a \bowtie \top_b + s_a \bowtie b + s_a \bowtie s_b \\
= \quad & \{\!\![ \text{ distributivity of } \bowtie ]\!\!\} \\
& a \bowtie \top_b + s_a \bowtie (b + s_b) \\
= \quad & \{\!\![ \text{ definition of } \& ]\!\!\} \\
& a \,\&\, (b + s_b) \ .
\end{aligned}
$$

Distributivity of $(a\,\&)$ over $\otimes$:

$$
\begin{aligned}
& a \,\&\, (b \otimes c) \\
= \quad & \{\!\![ \text{ definition of } \otimes ]\!\!\} \\
& a \,\&\, (b \mathbin{<\!\!\otimes} c + b \mathbin{\otimes\!\!>} c) \\
= \quad & \{\!\![ \text{ distributivity of } \& \text{ over } +, \text{ Cor. 5.12 } ]\!\!\} \\
& a \,\&\, (b \mathbin{<\!\!\otimes} c) + a \,\&\, (b \mathbin{\otimes\!\!>} c) \\
= \quad & \{\!\![ \text{ distributivity of } (a\,\&) \text{ over } \mathbin{<\!\!\otimes} \text{ and } \mathbin{\otimes\!\!>} ]\!\!\} \\
& (a \,\&\, b) \mathbin{<\!\!\otimes} (a \,\&\, c) + (a \,\&\, b) \mathbin{\otimes\!\!>} (a \,\&\, c) \\
= \quad & \{\!\![ \text{ definition of } \otimes ]\!\!\} \\
& (a \,\&\, b) \otimes (a \,\&\, c) \ .
\end{aligned}
$$

□

## Appendix D. Proofs for Section 6

*Appendix D.1. Proof of Lemma 6.2*

1. By transitivity of $a$ we have

$$(ra)^2 = r{\cdot}a{\cdot}r{\cdot}a \leq r{\cdot}a{\cdot}a \leq r{\cdot}a = (ra)^1 \ ,$$

   which implies transitivity of $(ra)$. Iterated application of transitivity shows the claim.

2. For $i = 0$ we obtain by a diamond property

$$\left\langle (ra)^1 \right\rangle r = \langle ra \rangle\, r = r{\cdot}\, \langle a \rangle\, r \leq r = \left\langle (ra)^0 \right\rangle r \ .$$

   For $i > 0$ the claim is immediate from Part 1 and isotony of diamond.

3. We perform again an induction on $i$.
   - $i = 0$: $\left\langle (ra)^0 \right\rangle r = \langle 1 \rangle\, r = r$.
   - $i \to i+1$: Assume $r_i = \left\langle (ra)^i \right\rangle r$.

$$
\begin{array}{ll}
 & r_{i+1} \\
= & \quad \{\!\![ \text{ definitions } ]\!\!\} \\
 & r - \sum\limits_{j=0}^{i} q_j \\
= & \quad \{\!\![ \text{ splitting the sum and definitions } ]\!\!\} \\
 & r_i - q_i \\
= & \quad \{\!\![ \text{ definition } q_i \ ]\!\!\} \\
 & r_i - a \rhd r_i \\
= & \quad \{\!\![ \text{ definition } \rhd \ ]\!\!\} \\
 & r_i - (r_i - \langle a \rangle\, r_i) \\
= & \quad \{\!\![ \text{ definition of } -, \text{ De Morgan } ]\!\!\} \\
 & r_i{\cdot}\, (\neg r_i + \langle a \rangle\, r_i) \\
= & \quad \{\!\![ \text{ distributivity, } p{\cdot}\neg p = 0 \ ]\!\!\} \\
 & r_i{\cdot}\, \langle a \rangle\, r_i \\
= & \quad \{\!\![ \ r_i \leq r \text{ by definition } ]\!\!\} \\
 & r_i{\cdot}r{\cdot}\, \langle a \rangle\, r_i \\
= & \quad \{\!\![ \text{ diamond property } ]\!\!\} \\
 & r_i{\cdot}\, \langle ra \rangle\, r_i \\
= & \quad \{\!\![ \text{ induction hypothesis } ]\!\!\} \\
 & \left(\left\langle (ra)^i \right\rangle r\right) \cdot \left(\langle ra \rangle \left\langle (ra)^i \right\rangle r\right) \\
= & \quad \{\!\![ \text{ diamond property, definition of powers } ]\!\!\} \\
 & \left(\left\langle (ra)^i \right\rangle r\right) \cdot \left(\left\langle (ra)^{i+1} \right\rangle r\right) \\
= & \quad \{\!\![ \text{ Part 2 } ]\!\!\} \\
 & \left\langle (ra)^{i+1} \right\rangle r \ .
\end{array}
$$

□

## Appendix E. About Normality

We now state the announced condition for normality that covers also infinite sets. It turns out that essentially absence of infinitely ascending sequences w.r.t. an element $a$ is sufficient. This property is also known as noetherity. Mathematically, it is equivalent to the property that for every non-empty set $p$ also the set $a \rhd p$ of maxima is non-empty. We use the logical contraposition of this property for the formal definition.

**Definition 1 (Noetherian Elements and d-Transitivity).** An element $a$ of a pre-domain semiring $(S, T, {}^\ulcorner)$ is *noetherian* if, for all $p \in T$,

$$a \triangleright p \leq 0 \Rightarrow p \leq 0 \ .$$

An element $a$ is *d-transitive* if for all $p$ we have $\langle a \rangle \langle a \rangle p \leq \langle a \rangle p$.

By the definition of $\triangleright$ and shunting noetherity is equivalent to

$$p \leq \langle a \rangle p \ \Rightarrow \ p \leq 0 \ .$$

The condition of d-transitivity (the "d" referring to "diamond") is more liberal than the notion of transitivity given in Def. 3.7 and suffices for the proofs in this section.

**Theorem 1 (Normality and Noetherity).**

1. *Every normal element $a$ is noetherian and d-transitive.*
2. *Every d-transitive and noetherian element $a$ is normal.*

Proof.

1. Assume $a \triangleright p \leq 0$, which is equivalent to $p \leq \langle a \rangle p$, cf. Definition 1. Now we obtain by the assumptions, normality of $a$ and isotony/strictness of diamond

$$p \leq \langle a \rangle p \leq \langle a \rangle (a \triangleright p) \leq \langle a \rangle 0 = 0 \ .$$

For d-transitivity, we will show $\langle a \rangle p + \langle a \rangle \langle a \rangle p \leq \langle a \rangle p$, which implies $\langle a \rangle \langle a \rangle p \leq \langle a \rangle p$. First, by distributivity of diamond and normality we obtain

$$\langle a \rangle p + \langle a \rangle \langle a \rangle p = \langle a \rangle (p + \langle a \rangle p) \leq \langle a \rangle (a \triangleright (p + \langle a \rangle p)) \ .$$

Now we continue with the maximum expression.

$$\begin{array}{ll} & a \triangleright (p + \langle a \rangle p) \\ = & \quad \{\!\![ \text{ definition of } \triangleright \ ]\!\!\} \\ & (p + \langle a \rangle p) - \langle a \rangle (p + \langle a \rangle p) \\ = & \quad \{\!\![ \text{ distributivities } ]\!\!\} \\ & (p - \langle a \rangle p - \langle a \rangle \langle a \rangle p) + (\langle a \rangle p - \langle a \rangle p - \langle a \rangle \langle a \rangle p) \\ = & \quad \{\!\![ \text{ Boolean algebra } ]\!\!\} \\ & (p - \langle a \rangle p - \langle a \rangle \langle a \rangle p) + 0 \\ \leq & \quad \{\!\![ \text{ definition of } - \text{ and neutrality of } 0 \ ]\!\!\} \\ & p \ . \end{array}$$

Hence, by isotony of diamond we are done.

2. If we have a noetherian element $a$ we can use it to show $t \leq u$ for tests $t, u$ by showing $t - u \leq \langle a \rangle (t - u)$, since this by noetherity of $a$ implies $t - u \leq 0$. The formula $t - u \leq \langle a \rangle (t - u)$ is, by shunting, equivalent to $t \leq u + \langle a \rangle (t - u)$. Specialising $t$ and $u$ to $\langle a \rangle v$ and $\langle a \rangle w$ for tests $v$ and $w$ we obtain

$$\langle a \rangle v \leq \langle a \rangle w \ \Leftarrow \ \langle a \rangle v \leq \langle a \rangle w + \langle a \rangle (\langle a \rangle v - \langle a \rangle w) \ .$$

We transform the right hand side of the premise under the assumption that $a$ is d-transitive.

$$\begin{array}{ll} & \langle a \rangle w + \langle a \rangle (\langle a \rangle v - \langle a \rangle w) \\ = & \quad \{\!\![ \text{ d-transitivity of } a \ ]\!\!\} \\ & \langle a \rangle w + \langle a \rangle \langle a \rangle w + \langle a \rangle (\langle a \rangle v - \langle a \rangle w) \\ = & \quad \{\!\![ \text{ distributivity } ]\!\!\} \end{array}$$

$$
\begin{aligned}
& \langle a\rangle\, w + \langle a\rangle\,(\langle a\rangle\, w + (\langle a\rangle\, v - \langle a\rangle\, w)) \\
= \quad & \{\!\![\ \text{Boolean algebra}\ ]\!\!\} \\
& \langle a\rangle\, w + \langle a\rangle\,(\langle a\rangle\, w + \langle a\rangle\, v) \\
= \quad & \{\!\![\ \text{distributivity and d-transitivity of } a\ ]\!\!\} \\
& \langle a\rangle\,(w + \langle a\rangle\, v)\ .
\end{aligned}
$$

Hence, for noetherian and d-transitive $a$,

$$\langle a\rangle\, v \le \langle a\rangle\, w \ \Leftarrow\ \langle a\rangle\, v \le \langle a\rangle\,(w + \langle a\rangle\, v)\ .$$

To show our main claim we now specialise $v$ to $q$ and $w$ to $a \triangleright q$. Then the right hand side of the premise simplifies, by distributivity, the definition of the maxima operator and Boolean algebra as follows:

$$\langle a\rangle\,(a \triangleright q + \langle a\rangle\, q) = \langle a\rangle\,((q - \langle a\rangle\, q) + \langle a\rangle\, q) = \langle a\rangle\,(q + \langle a\rangle\, q)\ .$$

Therefore, by isotony of diamond the premise of the above implication is satisfied and we are done. □

## Appendix F. Proof of Lemma 6.3

1. Immediate from Lemma 6.2.
2. Let w.l.o.g. $j \ge i+1$. It follows:

$$
\begin{aligned}
& q_i \cdot q_j \\
= \quad & \{\!\![\ \text{definition of } r_i \text{ and } \triangleright\ ]\!\!\} \\
& (r_i - \langle a\rangle\, r_i)\cdot(r_j - \langle a\rangle\, r_j) \\
= \quad & \{\!\![\ \text{Boolean algebra}\ ]\!\!\} \\
& r_i \cdot r_j - (\langle a\rangle\, r_i + \langle a\rangle\, r_j) \\
= \quad & \{\!\![\ r_j \le r_i \text{ by (1) and } j \ge i+1, \text{ isotony of diamond}\ ]\!\!\} \\
& r_j - \langle a\rangle\, r_i \\
= \quad & \{\!\![\ \text{Lemma 6.2 for } r_i, r_j\ ]\!\!\} \\
& \left\langle (ra)^j \right\rangle r - \langle a\rangle \left\langle (ra)^i \right\rangle r \\
\le \quad & \{\!\![\ r \le 1_A\ ]\!\!\} \\
& \left\langle (ra)^j \right\rangle r - r \cdot \langle a\rangle \left\langle (ra)^i \right\rangle r \\
= \quad & \{\!\![\ \text{diamond properties}\ ]\!\!\} \\
& \left\langle (ra)^j \right\rangle r - \left\langle (ra)^{i+1} \right\rangle r \\
= \quad & \{\!\![\ (ra)^j \le (ra)^{i+1} \text{ by Lemma 6.2.1 and } j \ge i+1\ ]\!\!\} \\
& 0_a\ .
\end{aligned}
$$

3. By transitivity and irreflexivity of $a$ there are always maximal elements in non-empty sets, i.e. we have $r \ne 0_a \ \Rightarrow\ a \triangleright r \ne 0_a$. Hence $r_i \ne 0_a$ implies $q_i \ne 0_a$. Additionally the $q_i$ are pairwise disjoint by Part 2, hence $r_{i+1}$ is strictly less (i.e. $r_{i+1} \le r_i \wedge r_{i+1} \ne r_i$) than $r_i$. Induction shows that the sequence $r_i$ is strictly decreasing for $i = 0, ..., (N+1)$ and equals $0_a$ for $i = (N+1), ..., \infty$.
4. Immediate from Part 3 and the definition of the $q_i$, since the definition of $N$ implies $r_{N+1} = 0_a$.
5. First, we have

$$
\begin{aligned}
& q_i \cdot a \cdot q_j = 0_a \\
\Leftrightarrow \quad & \{\!\![\ \text{domain is strict w.r.t. } 0_a\ ]\!\!\} \\
& \ulcorner(q_i \cdot a \cdot q_j) = 0_a \\
\Leftrightarrow \quad & \{\!\![\ \text{definition of diamond}\ ]\!\!\}
\end{aligned}
$$

$$
\begin{array}{ll}
 & \langle q_i \cdot a \rangle\, q_j = 0_a \\
\Leftrightarrow & \quad \{\![ \text{ property of diamond } ]\!\} \\
 & q_i \cdot \langle a \rangle\, q_j = 0_a \ .
\end{array}
$$

Now,

$$
\begin{array}{ll}
 & q_i \cdot \langle a \rangle\, q_j \\
= & \quad \{\![ \text{ definition of } q_i, q_j \ ]\!\} \\
 & (r_i - \langle a \rangle\, r_i) \cdot \langle a \rangle\, (r_j - \langle a \rangle\, r_j) \\
\leq & \quad \{\![ \text{ isotony of diamond } ]\!\} \\
 & (r_i - \langle a \rangle\, r_i) \cdot \langle a \rangle\, r_j \\
\leq & \quad \{\![ \ i \leq j, \text{ hence } r_j \leq r_i \text{ by Part 1 } ]\!\} \\
 & (r_i - \langle a \rangle\, r_i) \cdot \langle a \rangle\, r_i \\
= & \quad \{\![ \text{ Boolean algebra } ]\!\} \\
 & 0_a \ .
\end{array}
$$

□