Exploring Modal Worlds

H.-H. Dang, R. Glück, B. Möller, P. Roocks, A. Zelend

Institut für Informatik, Universität Augsburg, D-86135 Augsburg, Germany

Abstract

Modal idempotent semirings cover a large set of different applications. The paper presents a small collection of these, ranging from algebraic logics for program correctness over bisimulation refinement, formal concept analysis, database preferences to feature oriented software development. We provide new results and/or views on these domains; the modal semiring setting allows a concise and unified treatment, while being more general than, e.g., standard relation algebra.

Keywords: Bisimulation, formal concept analysis, Pareto front, rectangles, separation logic, software product lines

1. Introduction

Algebraic structures, such as *modal* idempotent semirings or Kleene algebras, offer a large variety of applications, while requiring only a small set of operators and axioms. Such algebras abstractly capture so-called Kripke structures, i.e., access relations over a set of worlds or states. In addition they provide the associated multi-modal operators box and diamond that allow reasoning, e.g., about possible actions of agents in a system or about state transitions in general. Particular instances of modal semirings are provided by the algebra of homogeneous binary relations and by abstract relation algebras.

This setting allows many general considerations and results, ranging from epistemic logics with knowledge and belief [1] to propositional dynamic Hoare logic and resource-based settings such as separation logic [2]. Moreover, many further applications are covered, like abstract reasoning about bisimulations for model refinement [3], formal concept analysis, simple and concise correctness proofs for the optimisation of database preference queries [4] or generally applicable models of module hierarchies in a feature oriented software development process [5].

In this paper, we take the readers on a short tour through several of these modal worlds and hope that they will enjoy the ride, maybe even feel some kind of explorer's excitement. We provide new results and/or views on the mentioned applications; the modal semiring setting allows a concise and unified treatment, while being more general than, e.g., standard relation algebra. Nevertheless the excellent relational papers and books by Gunther Schmidt [6, 7] are gratefully and respectfully acknowledged as a constant source of inspiration (although at times the relational encoding requires some "decryption" to obtain smooth modal formulations). It is our pleasure to dedicate this paper to Gunther at the occasion of his 75th birthday!

The paper is organised as follows. In Section 2 we recapitulate the main definitions of modal idempotent semirings and provide some generally applicable laws. Section 3 extends an existing algebraic framework from autobisimulations to bisimulations between different relations. An algebraic treatment of formal concepts and rectangles is given in Section 4, while in Section 5 we set up a connection between rectangles and Pareto fronts in databases with preference queries. Section 6 considers an abstract partial correctness approach to

Email addresses: h.dang@informatik.uni-augsburg.de (H.-H. Dang), glueck@informatik.uni-augsburg.de (R. Glück), moeller@informatik.uni-augsburg.de (B. Möller), roocks@informatik.uni-augsburg.de (P. Roocks), zelend@informatik.uni-augsburg.de (A. Zelend)

separation logic, and Section 7 provides an algebra of modules for structured documents in software product lines.

2. Basics of Modal Semirings

Idempotent semirings are a well-known concept for modelling choice and sequential composition by the algebraic operations $+$ and $\cdot$.

Definition 2.1. A *semiring* is a structure $(S, +, 0, \cdot, 1)$ with $0 \neq 1$ such that $+$ and $\cdot$ are associative binary operations on S with neutral elements 0 and 1 resp., $+$ is commutative, and $\cdot$ distributes both from left and right over $+$. Moreover, 0 is an annihilator of $\cdot$, i.e., $x \cdot 0 = 0 = 0 \cdot x$ holds for all $x \in S$.

The operations $+$ and $\cdot$ are also called *addition* and *multiplication*, resp. As usual, multiplication binds stronger than addition, so $x + y \cdot z$ stands for $x + (y \cdot z)$. Due to associativity we are free to omit superfluous parentheses.

A semiring is called *idempotent* if $x + x = x$ holds for all $x \in S$. In this case, the relation $\leq \subseteq S \times S$, defined by $x \leq y \Leftrightarrow_{df} x + y = y$, is a partial order on S, called the *natural order*. In particular, the supremum of two elements x and y with respect to the natural order is given by $x + y$, the least element is 0, and both addition and multiplication are isotone. The infimum of two elements x and y need not exist; if it does it is denoted by $x \sqcap y$. The element 0 is irreducible with respect to $+$, i.e., $x + y = 0 \Leftrightarrow x = 0 = y$ holds for all $x, y \in S$.

For an arbitrary set M, the structure $(\mathsf{Rel}(M), \cup, \emptyset, ;, \mathsf{id}(M))$ forms an idempotent semiring where $\mathsf{Rel}(M)$ denotes the set of all relations over M, ; denotes relational composition and $\mathsf{id}(M)$ the identity relation on M.

Definition 2.2. A semiring $(S, +, 0, \cdot, 1)$ is called *Boolean* if it is equipped with a *complement operation* $\overline{\cdot} : S \to S$ with the following properties for all $x, y \in S$:

$$x + \overline{x} = y + \overline{y} \qquad \text{and} \qquad x \sqcap \overline{x} = y \sqcap \overline{y} \;, \tag{1}$$

$$\overline{x + y} = \overline{x} \sqcap \overline{y} \qquad \text{and} \qquad \overline{x \sqcap y} = \overline{x} + \overline{y} \;. \tag{2}$$

In an idempotent Boolean semiring, the element $\top =_{df} \overline{0}$ is the greatest element with respect to the natural order. Moreover, $x + \overline{x} = \top$, $x \sqcap \overline{x} = 0$ and $\overline{\overline{x}} = x$ for all $x \in S$. The structure $(\mathsf{Rel}(M), \cup, \emptyset, ;, \mathsf{id}(M))$ becomes a Boolean semiring if we define the complement operation by $\overline{R} =_{df} (M \times M) \backslash R$ (where $\backslash$ denotes set theoretic difference). The greatest element is the universal relation $M \times M$.

In $\mathsf{Rel}(M)$ a subset $N \subseteq M$ can be characterised by the associated partial identity $\mathsf{id}(N)$. This is abstracted to general idempotent semirings by the notion of tests as axiomatised in [8].

Definition 2.3. An element p of an idempotent semiring is called a *test* if it has a *relative complement* $\neg p$ with the properties $p + \neg p = 1$ and $p \cdot \neg p = 0 = \neg p \cdot p$.

In an idempotent semiring $(S, +, 0, \cdot, 1)$ the set of tests is denoted by $\mathsf{test}(S)$. As a writing convention, elements of $\mathsf{test}(S)$ are denoted by p, q, r and variants thereof. On tests, multiplication coincides with the infimum, i.e., we have $p \sqcap q = p \cdot q$ for all $p, q \in \mathsf{test}(S)$. As a consequence of this fact, multiplication on tests is both idempotent and commutative. Moreover, on tests also addition distributes over multiplication, i.e., $p + q \cdot r = (p + q) \cdot (p + r)$ holds for all tests p, q and r. The structure $(\mathsf{test}(S), +, 0, \cdot, 1)$ is a Boolean semiring with $\neg$ as complement operation and greatest element 1.

Since all tests are ≤ 1, multiplication with a test corresponds to restriction. If a stands for an abstract transition element, such as a relation, $p \cdot a$ restricts a to starting states that lie in the set p and $a \cdot q$ to ending states in q. In $(\mathsf{Rel}(M), \cup, \emptyset, ;, \mathsf{id}(M))$ the tests are exactly the subrelations of $\mathsf{id}(M)$. We will use that in Section 7.

A few further useful properties are collected in the following lemma.

Lemma 2.4.

1. *In a Boolean semiring $(S,+,0,\cdot,1)$ all elements $p \leq 1$ are tests with relative complement $\neg p = 1 \sqcap \overline{p}$.*
2. *In every idempotent semiring we have the following properties for all $a,b \in S$ such that $a \sqcap b$ exists, and all $p,q \in \mathsf{test}(S)$,*
$$p \cdot (a \sqcap b) = p \cdot a \sqcap b = p \cdot a \sqcap p \cdot b \ ,$$
$$(a \sqcap b) \cdot p = a \cdot p \sqcap b = a \cdot p \sqcap b \cdot p \ .$$

The next concept we introduce are the *domain* and *codomain* operations.

Definition 2.5. A *modal* semiring is an idempotent semiring $(S,+,0,\cdot,1)$ with two additional operations $\lceil\cdot : S \to \mathsf{test}(S)$ and $\cdot\rceil : S \to \mathsf{test}(S)$, fulfilling the following properties for all $x,y \in S$ and $p \in \mathsf{test}(S)$:

$x \leq \lceil x \cdot x$	and	$x \leq x \cdot \overline{x}$,	(d1/cd1)
$\lceil(p \cdot x) \leq p$	and	$(x \cdot p)\rceil \leq p$,	(d2/cd2)
$\lceil(x \cdot \lceil y) \leq \lceil(x \cdot y)$	and	$(\overline{x} \cdot y)\rceil \leq (x \cdot y)\rceil$.	(locality)

The operator $\lceil\cdot$ is called the *domain*, $\cdot\rceil$ the *codomain* operator. If only (d1/cd1) and (d2/cd2) hold the operator is called a *predomain/precodomain* operator.

It can be shown that the inequations (d1/cd1) and (locality) strengthen to equations (see [9]). Moreover, both domain and codomain are fully strict, i.e., $\lceil x = 0 \Leftrightarrow x = 0$ and $\overline{x}\rceil = 0 \Leftrightarrow x = 0$ hold for all $x \in S$. Domain and codomain operations on $\mathsf{test}(S)$ are simply the identity operations, i.e., for all $p \in \mathsf{test}(S)$ we have $\lceil p = p = \overline{p}\rceil$. Both operations distribute over addition, i.e., $\lceil(x+y) = \lceil x + \lceil y$ and $(x+y)\rceil = \overline{x}\rceil + \overline{y}\rceil$ hold for all $x,y \in S$. As a consequence thereof, they are isotone with respect to the natural order, i.e., $x \leq y \Rightarrow \lceil x \leq \lceil y \wedge \overline{x}\rceil \leq \overline{y}\rceil$ holds for arbitrary $x,y \in S$. In the case of existence, domain and codomain are uniquely determined.

The domain and codomain operators on $(\mathsf{Rel}(M), \cup, \emptyset, ;, \mathsf{id}(M))$ are given by $\lceil R = \{(m,m) \mid \exists y : (m,y) \in R\}$ and $\overline{R}\rceil = \{(m,m) \mid \exists y : (y,m) \in R\}$.

Based on domain and codomain we define the diamond and box operators for arbitrary $x \in S$ and $p \in \mathsf{test}(S)$ as follows:

Definition 2.6.

$\lvert x\rangle p =_{df} \lceil(x \cdot p)$,	$\langle x\rvert p =_{df} \lceil(p \cdot x)$,	(forward/backward diamond)
$\lvert x] p =_{df} \neg\lvert x\rangle\neg p$,	$[x\rvert p =_{df} \neg\langle x\rvert\neg p$.	(forward/backward box)

For $R \in \mathsf{Rel}(M)$, forward diamond and backward diamond correspond to the preimage and image of a subset of M under R, resp. The forward box $\lvert x]p$ models the set of all elements of M from where every transition under x leads inevitably into the subset corresponding to p. An analogous interpretation can be given for the backward diamond.

As an inheritance of domain and codomain, the diamond operators are isotone in both arguments and distribute in both arguments over addition. The box operators are antitone in the first argument and isotone in the second argument. This follows also from the fact that we have the following Galois connections [10] between the modal operators:

$$p \leq \lvert a]q \Leftrightarrow \langle a\rvert p \leq q \ , \qquad p \leq [a\rvert q \Leftrightarrow \lvert a\rangle p \leq q \ .$$

If locality holds, the operators also distribute over composition:

$$\lvert a \cdot b\rangle p = \lvert a\rangle\lvert b\rangle p \ , \quad \langle a \cdot b\rvert p = \langle b\rvert\langle a\rvert p \ , \quad \lvert a \cdot b]p = \lvert a]\lvert b]p \ , \quad [a \cdot b\rvert p = [b\rvert[a\rvert p \ . \tag{3}$$

Finally, we have for element a and tests p,q with atomic p that

$$p \leq \lvert a\rangle q \Leftrightarrow p \cdot a \cdot q \neq 0 \ . \tag{4}$$

Many further properties of modal operators can be found in [11].

3. Bisimulations

Bisimulations are a frequently used tool, not only in process algebra but also in model checking and control theory. In this section we show how to model them algebraically in modal semirings and prove some of their basic properties in a very simple calculational style.

In relational algebra a left and right total relation $B \subseteq X \times Y$ is called a *bisimulation* between two relations $R_1 \subseteq X \times X$ and $R_2 \subseteq Y \times Y$ if

$$B^\smile ; R_1 \subseteq R_2 ; B^\smile \ \wedge \ B ; R_2 \subseteq R_1 ; B \ .$$

In [3] this characterisation was used to reason about autobisimulations, i.e., bisimulations between a relation R and itself. Here we will derive a framework which allows reasoning about bisimulations between two different relations.

In our setting, left and right totality of a bisimulation b between two elements g_1 and g_2 can easily be modelled by the condition $\lceil b \rceil = \lceil g_1 + g_1 \rceil \wedge \lceil b \rceil = \lceil g_2 + g_2 \rceil$.

Given two functions $f_1, f_2 : \mathsf{test}(S) \to \mathsf{test}(S)$, we write $f_1 = f_2$ iff for all $p \in \mathsf{test}(S)$ the equality $f_1(p) = f_2(p)$ holds. Analogously we define the predicate $f_1 \leq f_2$ by $f_1(p) \leq f_2(p)$ for all $p \in \mathsf{test}(S)$. We say that an element $g_2 \in S$ is a *pseudoconverse* of an element $g_1 \in S$ if $|g_1\rangle = \langle g_2|$. In [3] it is shown that this requirement is equivalent to $\langle g_1| = |g_2\rangle$. These considerations lead to the following definition:

Definition 3.1. Let S be a modal semiring with locality. An element $b \in S$ is called a bisimulation between two elements $g_1 \in S$ and $g_2 \in S$ if the following conditions are fulfilled:

$$\lceil b \rceil = \lceil g_1 + g_1 \rceil \ \wedge \ \lceil b^\smile \rceil = \lceil g_2 + g_2 \rceil \ , \tag{5}$$

$$\langle b||g_1\rangle \leq |g_2\rangle\langle b| \ \wedge \ |b\rangle|g_2\rangle \leq |g_1\rangle|b\rangle \ . \tag{6}$$

In this case, we write $g_1 \sim_b g_2$.

In the sequel we will show how some properties of bisimulations in a relational setting can be stated and proved in an algebraic manner based on Definition 3.1.

It is well known that the identity relation is a bisimulation between a relation and itself. Moreover, bisimulations are closed under taking the converse, relational composition and union. These properties are translated into the language of modal semirings in the following theorem.

Theorem 3.2. *In a modal semiring S, the following properties hold:*

1. *For every g, the test $\lceil g + g \rceil$ is a bisimulation between g and itself.*
2. *Let b be a bisimulation between g_1 and g_2, and let $b^\smile$ be a pseudoconverse of b. Then $b^\smile$ is a bisimulation between g_2 and g_1.*
3. *Let b_{12} be a bisimulation between g_1 and g_2, and let b_{23} be a bisimulation between g_2 and g_3. Then $b_{12} \cdot b_{23}$ is a bisimulation between g_1 and g_3.*
4. *Let b and b' be bisimulations between g_1 and g_2. Then $b + b'$ is a bisimulation between g_1 and g_2.*

Proof.

1. $\lceil g + g \rceil$ obviously fulfills Definition 3.1(5). For (6) we fix an arbitrary test p and reason as follows: by distributivity of $\langle \cdot |$, by $\lceil g, g \rceil \in \mathsf{test}(S)$ and $\langle q|r = q \cdot r$, by $\lceil g, g \rceil \in \mathsf{test}(S)$ again and $|q \cdot a\rangle r = q \cdot |a\rangle r$, by $\lceil g \cdot g = g$, and $|\lceil g \rceil \cdot g\rangle p \leq |g\rangle p$ by $\lceil g \rceil \leq 1$ and isotony of $|\cdot\rangle$:

   $$\langle \lceil g + g \rceil ||g\rangle p \ = \ \langle \lceil g||g\rangle p + \langle \lceil g \rceil ||g\rangle p \ = \ \lceil g \cdot |g\rangle p + \lceil g \rceil \cdot |g\rangle p \ = \ |\lceil g \cdot g\rangle p + |\lceil g \rceil \cdot g\rangle p \ = \ |g\rangle p + |\lceil g \rceil \cdot g\rangle p \ = \ |g\rangle p \ .$$

   Analogously we obtain $|g\rangle\langle \lceil g + g \rceil |p = |g\rangle p$, so we even have the equality $\langle \lceil g + g \rceil ||g\rangle p = |g\rangle\langle \lceil g + g \rceil |p$. The equality $|\lceil g + g \rceil \rangle |g\rangle p = |g\rangle |\lceil g + g \rceil \rangle p$ can be shown symmetrically.

2. By properties of pseudoconverses and Definition 3.1.5, $\lceil(b^\smile) = \overline{b}^\urcorner = \lceil g_2 + g_2\urcorner$ and symmetrically $(b^\smile)^\urcorner = \lceil\overline{b} = \lceil g_1 + g_1\urcorner$. Moreover, by the definition of pseudoconverse, $g_1 \sim_b g_2$, Definition 3.1.6, and property of pseudoconverse:

$$\langle b^\smile || g_2\rangle \;=\; |b\rangle|g_2\rangle \leq |g_1\rangle|b\rangle \;=\; |g_1\rangle\langle b^\smile| \;.$$

In a similar manner we obtain $|b^\smile\rangle|g_1\rangle \leq |g_2\rangle|b^\smile\rangle$, hence $b^\smile$ is a bisimulation between g_2 and g_1.

3. First, by (3), $g_2 \sim_{b_{23}} g_3$, Definition 3.1.5, $g_1 \sim_{b_{12}} g_2$, Definition 3.1.5, (3), and $g_1 \sim_{b_{12}} g_2$, Definition 3.1.6:

$$\lceil(b_{12} \cdot b_{23}) \;=\; \lceil(b_{12} \cdot \lceil b_{23}) \;=\; \lceil(b_{12} \cdot (\lceil g_2 + g_2\urcorner)) \;=\; \lceil(b_{12} \cdot b_{12}\urcorner) \;=\; \lceil b_{12} \;=\; \lceil g_1 + g_1\urcorner \;.$$

The property $(b_{12} \cdot b_{23})^\urcorner = \lceil g_3 + g_3\urcorner$ can be obtained symmetrically. Moreover, we have the following calculation (and a symmetric one for $\langle b_{12} \cdot b_{23}||g_1\rangle \leq |g_3\rangle\langle b_{12} \cdot b_{23}|$): by (3), $g_2 \sim_{b_{23}} g_3$, Definition 3.1.6, $g_1 \sim_{b_{12}} g_2$, Definition 3.1.6, and (3):

$$|b_{12} \cdot b_{23}\rangle|g_3\rangle \;=\; |b_{12}\rangle|b_{23}\rangle|g_3\rangle \leq |b_{12}\rangle|g_2\rangle|b_{23}\rangle \leq |g_1\rangle|b_{12}\rangle|b_{23}\rangle \;=\; |g_1\rangle|b_{12} \cdot b_{23}\rangle \;.$$

4. First, by distributivity of domain over addition, the assumption and idempotence of addition we have $\lceil(b + b') = \lceil b + \lceil b' = \lceil g_1 + g_1\urcorner$. An analogous calculation shows $(b + b')^\urcorner = \lceil g_2 + g_2\urcorner$. Moreover, we can argue as follows: by distributivity of diamond in its first argument, assumption, isotony of $+$, and distributivity of diamond in its first argument, and idempotence:

$$\langle b + b'||g_1\rangle \;=\; \langle b||g_1\rangle + \langle b'||g_1\rangle \leq |g_2\rangle\langle b| + |g_2\rangle\langle b'| \;=\; |g_2\rangle\langle b + b'| \;.$$

The property $|b + b'\rangle|g_2\rangle \leq |g_1\rangle|b + b'\rangle$ follows symmetrically. □

4. Concepts and Rectangles

In this section we deal with formal concept analysis as pioneered by Ganter and Wille [12]. A formal concept defines a maximal association between certain objects and attributes. Applications include data mining, text mining, machine learning, knowledge management, the semantic web, software development and biology. We show that by using a modal semiring formulation many of the basic properties fall out of standard laws quite easily. Also we set up a connection with rectangles and pseudorectangles which are used in the actual computation of concepts.

4.1. The Relational Case

We start with a brief recapitulation of the basic notions.

Definition 4.1. A *context* is a triple (O, A, R) with a set O of *objects*, a set A of *attributes* and a relation $R \subseteq O \times A$ that associates objects with attributes.

Example 4.2. (See e.g.[13]) Let O be the set of natural numbers from 1 to 10 and $A = \{$*composite*, *even*, *odd*, *prime*, *square*$\}$ with the obvious association relation R. □

Definition 4.3. Over a context one defines two functions $F : \mathcal{P}(O) \to \mathcal{P}(A)$ and $G : \mathcal{P}(A) \to \mathcal{P}(O)$ by setting, for $X \subseteq O$ and $Y \subseteq A$,

$$\begin{array}{lll} F(X) & =_{df} & \{a \in A \,|\, \forall\, o \in X : o\, R\, a\} \;, \\ G(Y) & =_{df} & \{o \in O \,|\, \forall\, a \in Y : o\, R\, a\} \;. \end{array}$$

In words: $F(X)$ is the set of attributes shared by all objects in X, while $G(Y)$ is the set of objects that share all attributes in Y.

By the definitions of G, of the Cartesian product twice and of F we have

$$\begin{array}{lll} X \subseteq G(Y) & \Leftrightarrow & (\forall\, o \in X : \forall\, a \in Y : o\, R\, a) \Leftrightarrow X \times Y \subseteq R \\ & \Leftrightarrow & (\forall\, a \in Y : \forall\, o \in X : o\, R\, a) \Leftrightarrow Y \subseteq F(X)\ . \end{array} \tag{GC}$$

This means that F and G form a Galois connection [10] between the posets $(\mathcal{P}(O), \subseteq)$ and $(\mathcal{P}(A), \supseteq)$. The middle property in this calculation gives rise to the following notion.

Definition 4.4. X and Y define a *rectangle* of R iff $X \times Y \subseteq R$.

By the standard theory of Galois connections, one has $X \subseteq G(F(X))$ and $Y \subseteq F(G(Y))$ as well as $F(X) = F(G(F(X))$ and $G(Y) = G(F(G(Y)))$. Hence, given a set X of objects, $G(F(X))$ is the greatest set of objects that have the same attributes as the objects in X, and similarly for $F(G(Y))$. This motivates the following notion.

Definition 4.5. A *concept* within the context (O, A, R) is a pair $C = (X, Y)$ with $X \subseteq O, Y \subseteq A$ and $X = G(Y)$ and $Y = F(X)$. X and Y are called the *extension* and the *intension* of C, resp.

Example 4.6. Let O, A and R be as in Example 4.2. Two examples of concepts are then

$$\begin{array}{ll} (\{3,5,7\}, \{odd, prime\}) & \text{and} \\ (\{1,4,9\}, \{square\})\ . & \end{array}$$

Informally, these concepts would be described as *the odd prime numbers* and *the squares* within the given context. □

By the standard theory of Galois connections, $X \subseteq O$ is the extension of a concept iff $X = G(F(X))$, which then is $(X, F(X))$. Symmetrically, $Y \subseteq A$ is the intension of a concept iff $Y = F(G(Y))$, which then is $(G(Y), Y)$. Moreover, $G \circ F \circ G = G$ and $F \circ G \circ F = F$; hence all images under F are intensions and all images under G are extensions. By (GC) the concepts $(X, F(X))$ and $(G(Y), Y)$ give rise to the rectangles $X \times F(X)$ and $G(Y) \times Y$, resp.

Finally, again by the Galois connection, F and G are universally conjunctive, i.e., preserve all existing infima. This can be exploited for a more efficient way of computing all concepts of a context: for $X \subseteq O, Y \subseteq A$ we have $F(X) = \bigcap_{o \in X} F(\{o\})$ and $G(Y) = \bigcap_{a \in Y} G(\{a\})$ For a concept (X, Y) therefore $Y = \bigcap_{o \in X} F(\{o\})$ and $X = \bigcap_{a \in Y} G(\{a\})$. By standard convention, for empty Y or X these intersections yield A or O, resp. Therefore it suffices to compute first all intensions $\{F(\{o\}) \mid o \in O\}$ or extensions $\{G(\{a\}) \mid a \in A\}$ and to obtain the others as intersections of these.

4.2. The General Case and Modalities

We now abstract from the relational case to a Boolean modal semiring.

First we observe that a rectangle $X \times Y$ can be relationally represented as $I_X ; \mathbb{T} ; I_Y$, where I_X, I_Y are the partial identity relations associated with X, Y and $\mathbb{T}$ is the universal relation $O \times A$. This leads to the following definition.

Definition 4.7. Let $p, q \leq 1$ be elements of a Boolean modal semiring S. Then the *rectangle* defined by p and q is

$$p \times q =_{df} p \cdot \top \cdot q\ ,$$

where $\top =_{df} \overline{0}$ is the greatest element of S. For an arbitrary element a of S we say that $p \times q$ is a *rectangle* of a if $p \times q \leq a$.

Example 4.8. In the path semiring, a rectangle consists of all possible node sequences starting with a node in p and ending with a node in q. □

We list a few simple consequences of the definition.

Lemma 4.9.

1. $p \times q \sqcap a = p \cdot a \cdot q$.
2. $(p+q) \times r \leq a \Leftrightarrow p \times r \leq a \wedge q \times r \leq a$.
3. $p \times (q+r) \leq a \Leftrightarrow p \times q \leq a \wedge p \times r \leq a$.

Proof.

1. Immediate from the definition of rectangles and Lemma 2.4.2.
2. Immediate from distributivity and lattice algebra.
3. Immediate from distributivity and lattice algebra. □

This allows establishing a connection with program semantics. It is well known that the semantics of Hoare triples can be given algebraically as

$$\{p\}\, a\, \{q\} \Leftrightarrow_{df} p \cdot a \cdot \neg q \leq 0 \ .$$

Using Lemma 4.9.1 this transforms as follows:

$$\{p\}\, a\, \{q\} \Leftrightarrow p \times \neg q \sqcap a \leq 0 \Leftrightarrow p \times \neg q \leq \overline{a} \ .$$

Informally, any transitions from p states to $\neg q$ states must lie outside a.

We have the following characterisation of the rectangles of a, which for the relational case is also given in [14], albeit without using the notion of modal operators.

Lemma 4.10. *The following properties are equivalent.*

1. $p \times q$ *is a rectangle of* a.
2. $p \leq |\overline{a}]\neg q$.
3. $q \leq [\overline{a}|\neg p$.
4. $|\overline{a}\rangle q \leq \neg p$.
5. $\langle \overline{a}|p \leq \neg q$.

Proof. We obtain by the definition of $\times$, shunting, Lemma 2.4, greatestness of $\top$ and the standard connection between box and restriction,

$$\begin{array}{lll} p \times q \leq a & \Leftrightarrow & p \cdot \top \cdot q \leq a \Leftrightarrow p \cdot \top \cdot q \sqcap \overline{a} \leq 0 \\ & \Leftrightarrow & p \cdot \overline{a} \cdot q \sqcap \top \leq 0 \Leftrightarrow p \cdot \overline{a} \cdot q \leq 0 \Leftrightarrow p \leq |\overline{a}]\neg q \ . \end{array}$$

The remaining equivalences are standard for modal operators. □

Corollary 4.11. *For arbitrary tests* p, q *the following elements are rectangles of an element* a*:*

$$(|\overline{a}]\neg q) \times q \ , \qquad\qquad p \times ([\overline{a}|\neg p) \ .$$

Proof. Immediate from Lemma 4.10. □

The functions $f_a(p) =_{df} [\overline{a}|\neg p$ and $g_a(q) =_{df} |\overline{a}]\neg q$ are the abstract counterparts of F and G from Sect. 4.1. This motivates the following notion.

Definition 4.12. We call a pair (p, q) of tests a *concept* of a if $p = g_a(q)$ (equivalently, if $q = f_a(p)$).

4.3. Comparing Rectangles

We will now set up a connection between concepts and maximal rectangles of an element. Let us therefore first investigate the order between rectangles.

It turns out that we need a special assumption about the domain/codomain operators defining the modal operators.

Definition 4.13. A modal semiring S satisfies the

1. *Tarski property* if for all $a \in S$ we have $a \neq 0 \Rightarrow \top \cdot a \cdot \top = \top$ (TAR);
2. *weak Tarski property* if for all $p \in \mathsf{test}(S)$ we have $p \neq 0 \Rightarrow \top \cdot p \cdot \top = \top$ (WT);
3. *weak Tarski domain property* if for all $p \in \mathsf{test}(S)$ we have $p \neq 0 \Rightarrow \ulcorner(\top \cdot p) = 1$ (WTD);
4. *weak Tarski codomain property* if for all $p \in \mathsf{test}(S)$ we have $p \neq 0 \Rightarrow (p \cdot \top)^{\urcorner} = 1$ (WTC).

It is clear that (TAR) implies (WT) and that (WT) implies both (WTD) and (WTC). The reverse implications do not hold. The path semiring is an example satisfying both (WTD) and (WTC) but not (WT). However, we have the following result.

Lemma 4.14. *In a modal semiring* (WTD) *and* (WTC) *are equivalent.*

Proof. We show (WTD) $\Rightarrow$ (WTC): by $(p \cdot \top)^{\urcorner}$ being the least right preserver of $p \cdot \top$, Boolean algebra, characterisation of box, $q < 1$ implies $\neg q \neq 0$ and hence by (WTD) we have $|\top]q = \neg\ulcorner(\top \cdot \neg q) = \neg 1 = 0$, and logic:

$$(p \cdot \top)^{\urcorner} = 1 \Leftrightarrow \forall\, q < 1 : p \cdot \top \cdot q < p \cdot \top \Leftrightarrow \forall\, q < 1 : p \cdot \top \cdot \neg q \neq 0 \Leftrightarrow \forall\, q < 1 : p \not\leq |\top]q \Leftrightarrow \forall\, q < 1 : p \not\leq 0 \Leftrightarrow p \not\leq 0\ .$$

The reverse implication is symmetric. □

Because of this lemma we refer to both (WTD) and (WTC) uniformly as (WTM) ("M" standing for "modal").

Now we obtain the following representation of the order relation between non-empty rectangles.

Lemma 4.15. *Assume* (WTM)*. If* $p, q, r, s \neq 0$ *then*

$$p \times q \leq r \times s \Leftrightarrow p \leq r \wedge q \leq s\ .$$

Proof. ($\Leftarrow$) Immediate from isotony of $\cdot$.
($\Rightarrow$) We have, by the definition of rectangles, the import/export property of domain and (WTM),

$$\ulcorner(p \times q) = \ulcorner(p \cdot \top \cdot q) = p \cdot \ulcorner(\top \cdot q) = p \cdot 1 = p\ .$$

Symmetrically, $(p \times q)^{\urcorner} = q$. Now the claim follows by isotony of domain and codomain. □

This enables a characterisation of maximal non-empty rectangles.

Lemma 4.16. *Assume* (WTM) *and* $p, q \neq 0$*. Then* $p \times q$ *is a maximal rectangle of* a *iff* (p, q) *is a concept of* a*.*

Proof. ($\Rightarrow$) By Corollary 4.11 and Lemma 4.10 $b =_{df} (|\overline{a}]\neg q) \times q$ is a rectangle of a with $p \leq |\overline{a}]\neg q$. Since $p \times q$ is assumed to be a rectangle of a, Lemma 4.10 and Lemma 4.15 yield $p \times q \leq b$. Now maximality of $p \times q$ and again Lemma 4.15 show $p = |\overline{a}]\neg q = g_a(q)$.
($\Leftarrow$) First, by the definition of a concept and Lemma 4.10, $p \times q$ is a rectangle of a. Let $r \times s \leq p \times q$ be another rectangle of a. Then by Lemmas 4.10 and 4.15 we have

$$p \leq |\overline{a}]\neg q \wedge r \leq |\overline{a}]\neg s \wedge p \leq r \wedge q \leq s\ .$$

By shunting we obtain $\neg s \leq \neg q$ and isotony of box in its second argument shows $r \leq |\overline{a}]\neg q$, so that $p \leq r \leq |\overline{a}]\neg q$. But since (p,q) is a concept we have $p = |\overline{a}]\neg q$ and hence also $p = r$. Symmetrically one obtains $q = s$. □

We conclude this section by another useful consequence of Lemma 4.10.

Lemma 4.17. *If* (WTM) *holds then*

$$q \neq 0 \wedge p \times q \leq a \Rightarrow p \leq |a\rangle q \ ,$$
$$p \neq 0 \wedge p \times q \leq a \Rightarrow q \leq \langle a| p \ .$$

Proof. By $q \neq 0$, (WTM) and distributivity we have

$$1 = |\top\rangle q = |a + \overline{a}\rangle q = |a\rangle q + |\overline{a}\rangle q \ .$$

By this, distributivity and the shunted form $p \cdot |\overline{a}\rangle q \leq 0$ of Lemma 4.10.4,

$$p = p \cdot 1 = p \cdot |a\rangle q + p \cdot |\overline{a}\rangle q = p \cdot |a\rangle q \ ,$$

which by $p \leq 1$ implies $p \leq |a\rangle q$. The second claim can be shown symmetrically. □

4.4. Pseudo-Rectangles

It is a frequent task to find for a given element a coverage by formal concepts, i.e., by maximal rectangles. Since this is very expensive, in [15] the concept of a pseudo-rectangle is introduced, which can be determined in a much simpler way. Relationally, given a relation R, the pseudo-rectangle $\mathrm{P}(x,y,R)$ associated with a pair $(x,y) \in R$ is the union of all rectangles and hence of all maximal rectangles of R that contain (x,y). This is non-empty, since $\{(x,y)\}$ is a rectangle of R. We can give a general algebraic definition as follows.

Definition 4.18. We now assume that the underlying semiring S is a *quantale*, i.e., a complete lattice in which multiplication distributes over arbitrary suprema. Let $a \in S$ and $r \times s \leq a$ be a non-empty rectangle of a, i.e., assume $r, s \neq 0$. Then the set of a-rectangles covering $r \times s$ is

$$\mathit{rect}(r,s,a) \ =_{df} \ \{p \times q \,|\, r \times s \leq p \times q \leq a\} \ ,$$

and we call $\mathrm{P}(r,s,a) \ =_{df} \ \sum \mathit{rect}(r,s,a)$ the *pseudo-rectangle* induced by $r \times s$.

Lemma 4.19.

1. *If* (WTM) *holds then* $\mathrm{P}(r,s,a) \leq (|a\rangle s) \cdot a \cdot (\langle a|r)$.
2. *In the quantale of relations this strengthens to an equality provided r, s are atomic tests.*

Proof.

1. We show that $b \ =_{df} \ (|a\rangle s) \cdot a \cdot (\langle a|r)$ is an upper bound of $\mathit{rect}(r,s,a)$. Let $p \times q \in \mathit{rect}(r,s,a)$. By $r \times s \leq p \times q$ and Lemma 4.15 we obtain $r \leq p \wedge s \leq q$ so that by isotony also $r \times q \in \mathit{rect}(r,s,a)$ and $p \times s \in \mathit{rect}(r,s,a)$. Now, since $r \times s$ is non-empty, Lemma 4.17 implies $p \leq |a\rangle s \wedge q \leq \langle a|r$. Therefore, by $p \times q \leq a$, Lemma 4.9.1 and isotony,

$$p \times q \ = \ p \times q \ \sqcap \ a \ = \ p \cdot a \cdot q \ \leq \ |a\rangle s \cdot a \cdot \langle a|r \ = \ b \ .$$

2. We show that every pair in relation b lies also in every upper bound of $\mathit{rect}(r,s,a)$, so that b is indeed the least upper bound of $\mathit{rect}(r,s,a)$. Assume r, s to represent the elements u, v and that $(u,v) \in a$. For arbitrary pair (x,y) we obtain, by the definitions of b and relational composition as well as the definition of diamonds

$$\begin{aligned} (x,y) \in b \quad &\Leftrightarrow \quad x \in |a\rangle v \wedge y \in \langle a|u \wedge (x,y) \in a \\ &\Leftrightarrow \quad (x,v) \in a \wedge (u,y) \in a \wedge (x,y) \in a \ . \end{aligned}$$

   Together with $(u,v) \in a$ this implies that $c \ =_{df} \ \{u,x\} \times \{v,y\} \in \mathit{rect}(r,s,a)$ and hence $c \leq d$ for every upper bound d of $\mathit{rect}(r,s,a)$. By $(x,y) \in c$ this implies $(x,y) \in d$ as well. □

	a_1	a_2	a_3	a_4	a_5
o_1	1	1	0	0	0
o_2	1	1	0	0	1
o_3	1	1	1	1	0
o_4	1	0	0	0	1

(a) Relation a

	a_1	a_2	a_3	a_4	a_5
o_1	1	1	0	0	0
o_2	1	1	0	0	1
o_3	1	1	0	0	0
o_4	1	0	0	0	1

(b) Overapproximation $|a\rangle s \cdot a \cdot \langle a|r$

	a_1	a_2	a_3	a_4	a_5
o_1	1	1	0	0	0
o_2	1	1	0	0	0
o_3	1	1	0	0	0
o_4	0	0	0	0	0

(c) Pseudo-rectangle $\mathrm{P}(r, s, a)$

Figure 1: Behaviour of Pseudo-rectangles

It is clear that computing $\mathrm{P}(r, s, a) \leq (|a\rangle s) \cdot a \cdot (\langle a|r)$ is less expensive than determining and summing up all maximal elements of $rect(r, s, a)$. The following example shows that the assumption of atomicity of r, s cannot be dropped from Part 2, not even in the relational case.

Example 4.20. Consider the relation a of Fig.1(a). For $r =_{df} \{o_1, o_2\}$ and $s =_{df} \{a_1, a_2\}$ we obtain $|a\rangle s = \{o_1, o_2, o_3, o_4\}$ and $\langle a|r = \{a_1, a_2, a_5\}$. Hence $|a\rangle s \cdot a \cdot \langle a|r$ is the relation shown in Fig.1(b). This is strictly larger than $\mathrm{P}(r, s, a)$ which is shown in Fig.1(c). □

5. Pareto Fronts and Rectangles

A preference is a strict partial order on a given set, i.e., a special kind of a homogeneous binary relation. Queries in a database with a preference are supposed to return the maximal objects w.r.t. that preference, corresponding to optimal satisfaction in some sense of the user's wishes. By the so-called *Pareto operator* two preference orders a, b can be composed into the preference $a \otimes b$. The maxima set of $a \otimes b$ under a given set are *compromises* w.r.t. to the orders a, b in the following sense: In the maxima set there is no element which is strictly better in at least one of the preferences a, b while being not worse w.r.t. the other preference. Because of their characteristic shape, maxima sets of Pareto preferences are also called *Pareto fronts* or *skylines*. The Pareto principle is used to express that two (competing) objectives are equally important.

In the following we will derive a connection between these Pareto fronts and rectangles. This is based on our work in [4] and [16]. We recapitulate just some formal foundations while further definitions are introduced where needed.

Consider a set of type names. For each type name T let D_T be some domain of data base tuples and let 1_T and $\top_T$ represent the identity and universal relations on D_T. A type assertion $a :: T^2$ is short for $a \leq 1_T \cdot a \cdot 1_T$. We view such elements as representations of preference relations on the domain D_T. An assertion $p :: T$ means that p is a test, representing a set of tuples, with $p \leq 1_T$.

The join operator $\bowtie$ (more precisely defined in the *Join Algebra* in [16]) acts quite similar to a Cartesian product. For elements $a :: T_a$ and $b :: T_b$ we have $a \bowtie b :: T_a \bowtie T_b$ where the latter is the joined type of T_a and T_b. Joins on the same type are equivalent to the intersection, i.e., $a_1 \bowtie a_2 = a_1 \sqcap a_2$ for $a_i :: T$. For the sake of readability we define $1_x =_{df} 1_{T_x}$ and $\top_x =_{df} \top_{T_x}$.

5.1. Idea

The set of elements which are maximal under a Pareto preference form a Pareto front when connected as shown in Fig. 2. This Pareto front subdivides the given domain into two areas: One describes the *dominance region* consisting of the maximal elements of the given dataset and the elements "dominated" by them. Elements in the other area are not dominated w.r.t the preference and the given dataset. We show that if there are N maximal elements the dominance region can be described by N rectangles and its complement by $N+1$ rectangles. Ordering these rectangles by size (or a weighted size with respect to a given probability

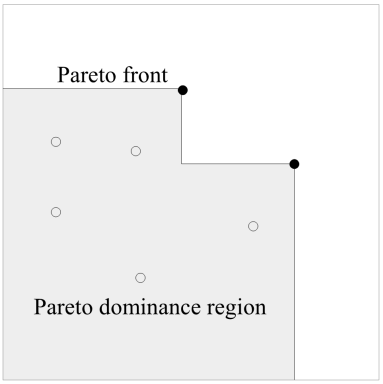

Figure 2: The Pareto dominance region (gray) and the Pareto front (black line). The filled circles are the maxima and the unfilled circles are dominated objects.

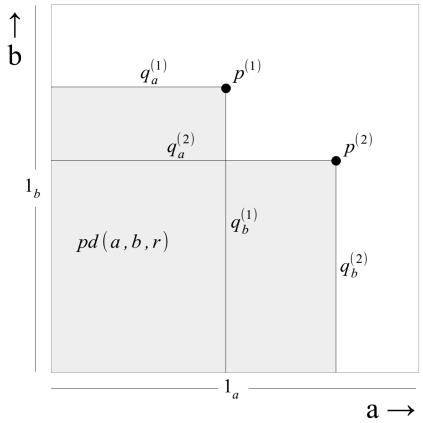

Figure 3: Rectangular representation. The arrows indicate where elements are better w.r.t. the preferences a and b. The $p^{(i)}$ are points while the $q^{(i)}$ are areas.

distribution of elements) paves the way for a fast calculation on which side of the Pareto front a new element would be placed.

5.2. Layered Preferences

In this section we assume a Boolean modal semiring with converse $^\smile$ satisfying the axioms

$$p^\smile = p \ , \quad (a+b)^\smile = a^\smile + b^\smile \ , \quad (a \cdot b)^\smile = b^\smile \cdot a^\smile \ \text{ and } \ \overline{a}^\smile = \overline{a^\smile} \ .$$

A *layered preference (or weak preorder)* is an irreflexive, transitive and negatively transitive element $a :: T_a^2$, i.e., satisfying, with $\overline{a} =_{df} \top_a - a$,

$$1 \sqcap a = 0 \ , \qquad a \cdot a \leq a \ , \qquad \overline{a} \cdot \overline{a} \leq \overline{a} \ .$$

For a layered preference a we define the element $\mathsf{s}_a =_{df} \overline{a} \sqcap \overline{a}^\smile$. It is easy to show that s_a is an equivalence relation, i.e., reflexive, transitive and symmetric. Moreover, a is a linear strict-order, i.e., $a + a^\smile + \mathsf{s}_a = \top_a$.

5.3. Representing the Pareto Front by Rectangles

Let $a :: T_a^2, b :: T_b^2$ be layered preferences with disjoint types T_a, T_b and let $r :: T_a \bowtie T_b$ be a dataset. To ease reading we define the following notation for any test p and element x:

$$p_x = p \sqcap 1_x \ .$$

Thus p_x is the projection to the domain of x. With this we have $p = p_a \bowtie p_b$.

The maximal elements of the Pareto preference $a \otimes b$ in r are given by

$$(a \otimes b) \triangleright r =_{df} r - |a \otimes b\rangle r \ ,$$

where the Pareto preference is defined by $a \otimes b =_{df} a \bowtie (b + \mathsf{s}_b) + (a + \mathsf{s}_a) \bowtie b$. Note that this coincidences with the *substitutable value* semantics of Pareto preferences defined in [16] (where the substitutability relations are $s_a = \mathsf{s}_a$ and $s_b = \mathsf{s}_b$). Note that if a is a total order then $\mathsf{s}_a = 1_a$. In general we have $1_x \leq \mathsf{s}_x$ for $x \in \{a, b\}$.

We define the *Pareto dominance region* by

$$pd(a, b, r) =_{df} |(a + \mathsf{s}_a) \bowtie (b + \mathsf{s}_b)\rangle((a \otimes b) \triangleright r) \ .$$

Assume a maxima set of N atomic elements $p^{(1)}, ..., p^{(N)}$, i.e., that

$$(a \otimes b) \triangleright r = p^{(1)} + ... + p^{(N)} \ .$$

Then the Pareto dominance region is given by

$$pd(a, b, r) = q^{(1)} + ... + q^{(N)} \quad \text{with} \quad q^{(i)} =_{df} |(a + \mathsf{s}_a) \bowtie (b + \mathsf{s}_b)\rangle p^{(i)} \ .$$

In Fig. 3 the $p^{(i)}$ and $q^{(i)}$ are shown.

Definition 5.1. A *joined rectangle* $r :: T_a \bowtie T_b$ is a test which can be written as a join, i.e., there are elements $r_a :: T_a$ and $r_b :: T_b$, such that $r = r_a \bowtie r_b$.

This coincidences with the usual definition of rectangles: Consider the following bijection, defined on atomic tests, where D_{T_a} and D_{T_b} are the domains of the disjoint types T_a and T_b:

$$f : T_a \bowtie T_b \to D_{T_a} \times D_{T_b}, \ \ t_a \bowtie t_b \mapsto t_a \cdot \top \cdot t_b \ .$$

By defining $f(p) =_{df} \sum_{q \in \mathsf{At}(p)} f(q)$, where $\mathsf{At}(p)$ is the set of atomic tests $\leq p$, this mapping can be extended to arbitrary tests.

Corollary 5.2.

1. *Any element r, for which $f(r)$ is a rectangle in the sense of Definition 4.7, is a joined rectangle.*
2. *Atomic tests are joined rectangles.*

Thus, datasets in $T_a \bowtie T_b$ can considered as heterogeneous relations via f. Note that this interpretation coincidences with the concept of a "database relation": A dataset with n columns (attributes) can be considered as a heterogeneous n-ary relation between its attributes. From now on, we abbreviate "joined rectangle" to *rectangle*.

Since the $p^{(i)}$ are atomic and hence rectangles, also the $q^{(i)}$ are rectangles as justified by

$$\begin{aligned} q^{(i)} \ &= \ |(a + \mathsf{s}_a) \bowtie (b + \mathsf{s}_b)\rangle p^{(i)} \ = \ |(a + \mathsf{s}_a) \bowtie (b + \mathsf{s}_b)\rangle (p_a^{(i)} \bowtie p_b^{(i)}) \\ &= \ |a + \mathsf{s}_a\rangle p_a^{(i)} \bowtie |b + \mathsf{s}_b\rangle p_b^{(i)} \ . \end{aligned}$$

Therewith we have a representation of $pd(a, b, r)$ as a sum of N rectangles.

Next, we derive a representation of its complement with $(N + 1)$ rectangles. We define the complement of the Pareto dominance region by

$$\overline{pd}(a, b, r) \ =_{df} \ \neg pd(a, b, r) = \mathsf{s}_a \bowtie \mathsf{s}_b \ - \ pd(a, b, r) \ .$$

Note that "$\bowtie$" binds stronger than "$-$" and "$+$".

Our aim is to find a more compact representation of $\overline{pd}(a, b, r)$. To this end we first give the following lemma.

Lemma 5.3. *Let a, b be layered preferences. For the maxima set $(a \otimes b) \triangleright r = p^{(1)} + ... + p^{(N)}$, the $p^{(i)}$ can always be arranged such that for all $i \in \{1, ..., N - 1\}$:*

1. (a) $p_a^{(i)} \ (\mathsf{s}_a + a) \ p_a^{(i+1)}$, (b) $p_b^{(i+1)} \ (\mathsf{s}_b + b) \ p_b^{(i)}$.
2. (c) $p^{(i)} \leq |\mathsf{s}_a + a\rangle p^{(i+1)}$, (d) $p^{(i+1)} \leq |\mathsf{s}_b + b\rangle p^{(i)}$.

Proof.

1. As a is a layered preference, the arrangement (a) is obviously possible. Next, we show that this implies (b). Since b is layered we have one of the following cases: (i) $p_b^{(i)} \ b \ p_b^{(i+1)}$, (ii) $p_b^{(i+1)} \ b \ p_b^{(i)}$, (iii) $p_b^{(i)} \ \mathsf{s}_b \ p_b^{(i+1)}$. Suppose case (i) $p_b^{(i)} \ b \ p_b^{(i+1)}$. Then $p^{(i)} \ ((\mathsf{s}_a + a) \bowtie b) \ p^{(i+1)}$ and hence $p^{(i)} \ (a \otimes b) \ p^{(i+1)}$, i.e., $p^{(i)}$ is dominated by $p^{(i+1)}$, a contradiction. Hence only the cases (ii) and (iii) are possible, which are compatible with (b).

2. Follows immediately from Part 1 and Equation (4). □

Now we give a compact representation of $\overline{pd}(a,b,r)$:

Lemma 5.4. *The set $\overline{pd}(a,b,r)$ can be expressed as a sum of $(N+1)$ rectangles as follows (where $\neg$ binds stronger than $\bowtie$):*

$$\overline{pd}(a,b,r) = \neg q_a^{(1)} \bowtie \mathsf{s}_b \ + \ \neg q_a^{(2)} \bowtie \neg q_b^{(1)} \ + \ ... \ + \ \mathsf{s}_a \bowtie \neg q_b^{(N)} \ . \tag{7}$$

Proof.

For convenience we set

$$q_a^{(0)} =_{df} \mathsf{s}_a, \ \ q_a^{(N+1)} =_{df} 0_a, \quad q_b^{(0)} =_{df} 0_b, \ \ q_b^{(N+1)} =_{df} \mathsf{s}_b \ .$$

From Lemma 5.3 we conclude for $i \in \{1,...,N-1\}$:

$$q_a^{(i+1)} = |a + \mathsf{s}_a\rangle p_a^{(i+1)} \leq |a + \mathsf{s}_a\rangle p_a^{(i)} = q_a^{(i)} \ .$$

Analogously $q_b^{(i)} \leq q_b^{(i+1)}$ and with the above conventions, $(q_a^{(i)})_{i=0,...,N+1}$ is decreasing while $(q_b^{(i)})_{i=0,...,N+1}$ is increasing in i. Due to this, the claim in Equation (7) is equivalent to

$$\overline{pd}(a,b,r) = \sum_{i=0}^{N} \neg q_a^{(i)} \bowtie \neg q_b^{(i+1)} \ .$$

For this we show $pd(a,b,r) \cdot \overline{pd}(a,b,r) = 0$ and $pd(a,b,r) + \overline{pd}(a,b,r) = \mathsf{s}_a \bowtie \mathsf{s}_b$.

1. Remember that $pd(a,b,r) = \sum_{i=1}^{N} q^{(i)}$. We have to show that all summands in $pd(a,b,r) \cdot \overline{pd}(a,b,r)$ are 0. We conclude for all i,j:

$$q^{(i)} \cdot (\neg q_a^{(j)} \bowtie \neg q_a^{(j+1)}) = \underbrace{(q_a^{(i)} - q_a^{(j)})}_{=:u_a} \bowtie \underbrace{(q_b^{(i)} - q_b^{(j+1)})}_{=:u_b} \ .$$

   Now we have either $j \leq i-1$ which implies that $u_a = 0$, or we have $j \geq i$ which implies that $u_b = 0$. Hence all summands are 0.

2. We use the following decomposition of $\mathsf{s}_a \bowtie \mathsf{s}_b$:

$$\mathsf{s}_a \bowtie \mathsf{s}_b = \sum_{i,j=0}^{N} \underbrace{(q_a^{(i)} - q_a^{(i+1)}) \bowtie (q_b^{(j+1)} - q_b^{(j)})}_{=:u_{i,j}} \ .$$

   Next, we show that any $u_{i,j}$ is contained either in $pd(a,b,r)$ or in $\overline{pd}(a,b,r)$. We distinguish two cases:

   (i) $j \leq i-1$: Because $q_b^{(i)}$ is increasing in i we have

$$u_{i,j} \leq q_a^{(i)} \bowtie q_b^{(j+1)} \leq q_a^{(i)} - q_b^{(i)} \leq pd(a,b,r) \ .$$

   (ii) $j \geq i$: Because $\neg q_b^{(i)}$ is decreasing in i we have

$$u_{i,j} \leq \neg q_a^{(i+1)} \bowtie \neg q_b^{(j)} \leq \neg q_a^{(i+1)} \bowtie \neg q_b^{(i)} \leq \overline{pd}(a,b,r) \ .$$

   Hence the sum of $pd(a,b,r)$ and $\overline{pd}(a,b,r)$ is the entire domain and the claim follows. □

Due to the above lemma we have a representation of $\overline{pd}(a,b,p)$ with $(N+1)$ rectangles.

5.4. Application

Assume a stream of *points*, i.e., atomic tests $t_a \bowtie t_b$. After receiving a point from the stream, the Pareto front has to be updated. We sketch how the above calculation will help to determine if a new point changes the Pareto front or not.

Assume a measure function $\mu : T_a \bowtie T_b \to [0,1]$ representing the probability in which area of $\mathsf{s}_a \bowtie \mathsf{s}_b$ points from the stream occur. An algorithm for deciding quickly if a new point is within pd or within $\overline{pd}$ should check the most probable rectangles w.r.t. μ first, i.e., we calculate $\mu(q^{(i)})$ for $i = 1, ..., N$ and $\mu(\neg q_a^{(i)} \bowtie \neg q_b^{(i+1)})$ for $i = 0, ..., N$. Then for a new point $t = t_a \bowtie t_b$ we check if $t \leq r^{(i)}$, where the sequence $(r^{(i)})$ enumerates the $2N + 1$ rectangles of pd and $\overline{pd}$ in a μ-decreasing order. The algorithm terminates if $t \leq r^{(i)}$ is true which gives evidence whether t is in pd or $\overline{pd}$.

If a new point t is in $\overline{pd}$ and should be added to the dataset then the rectangles representing pd and $\overline{pd}$ have to be recalculated. Note that it suffices to recalculate only those rectangles $q^{(i)}$ where the index i belongs to the set

$$I = \{i \in \{1, ..., N\} \mid p^{(i)} \leq |a \otimes b\rangle t\} \ ,$$

i.e., those rectangles are affected, where the corresponding points $p^{(i)}$ are dominated by t. For example, if $I = \{k\}$, then the three rectangles

$$q^{(k)}, \quad \neg q_a^{(k-1)} \bowtie \neg q_b^{(k)}, \quad \neg q_a^{(k)} \bowtie \neg q_b^{(k+1)}$$

have to be recalculated. Note that by transitivity of a and b and the definition of the Pareto preference, the set I is always an interval in $\mathbb{N}$, i.e., $\{l_1, l_1 + 1, ..., l_2\}$. If we have $|I| = k$, then $2k + 1$ rectangles will be replaced by 3 new rectangles.

Such an algorithm for quickly deciding if a new point is dominated by the existing maxima set is of interest for an application where the current maxima set of a stream (generating constantly new points) should be always up-to-date in real time, i.e., the dominance test for new points is a time-critical task.

For real applications one might not have the probability measure μ available, but this can be roughly estimated by the points from the stream which are already known to a given time.

6. Separation Logic, Partial Correctness and Abortion

We now turn to reasoning about program resources in standard separation logic (SSL). SSL is an extension of Hoare logic and facilitates reasoning about concurrent programs and sharing in data structures. It enables, due to its popular *frame rule*, modular reasoning which allows scalable program proofs. It is well-known that propositional Hoare logic can be treated using modal Kleene algebras. In [2, 17] a relation-algebraic treatment of SSL was presented. Starting from that approach we will show in the sequel that modal Kleene algebras can also be used to abstractly model separation logic both in a partial and a total correctness setting.

We first recapitulate the relational model of [17] to explain definitions of the algebra within that concrete model and to provide a better intuition. The relational model is built on a basic algebraic structure called *separation algebra* [18] that is used to abstractly capture resources of programs.

Definition 6.1. A *separation algebra* is a partial commutative monoid $(\Sigma, \bullet, u)$ where $\bullet$ is a commutative, associative and partial binary operation on Σ with unit u. An equation holds iff both sides are defined and equal, or both are undefined. The induced *combinability* relation $\#$ is given by

$$\sigma_0 \# \sigma_1 \Leftrightarrow_{df} \sigma_0 \bullet \sigma_1 \text{ is defined} \ .$$

Using this structure a *command* is a relation $C \subseteq \Sigma \times \Sigma$.

To model separation of commands we introduce as a next step special relations that enrich the setting with relations between pairs of states. By this all possible splits of a state w.r.t. $\#$ can be considered and thus an independent treatment of parts of a state is feasible.

Definition 6.2. Assume a separation algebra $(\Sigma, \bullet, u)$. The *split* relation $\lhd \subseteq \Sigma \times (\Sigma \times \Sigma)$ is given by

$$\sigma \lhd (\sigma_1, \sigma_2) \Leftrightarrow_{df} \sigma_1 \# \sigma_2 \wedge \sigma = \sigma_1 \bullet \sigma_2 \ .$$

The *join* relation $\rhd$ is the converse of split, i.e., $\rhd = \lhd^\smile$ with $(\sigma_1, \sigma_2) \rhd \sigma \Leftrightarrow_{df} \sigma_1 \# \sigma_2 \wedge \sigma = \sigma_1 \bullet \sigma_2$. The *Cartesian product* $C \times D \subseteq (\Sigma \times \Sigma) \times (\Sigma \times \Sigma)$ of two commands C, D is defined by

$$(\sigma_1, \sigma_2)\,(C \times D)\,(\tau_1, \tau_2) \Leftrightarrow_{df} \sigma_1\, C\, \tau_1 \wedge \sigma_2\, D\, \tau_2 \ .$$

Relation composition on Cartesian products is defined component-wise and hence we have the exchange law

$$(C_1 \times D_1) \,;\, (C_2 \times D_2) \ = \ C_1 \,;\, C_2 \ \times \ D_1 \,;\, D_2 \ .$$

Finally, we can define $*$-composition on arbitrary commands C, D that allows their concurrent execution on combinable or disjoint parts of a state.

Definition 6.3. The *$*$-composition* of commands $C, D \subseteq \Sigma \times \Sigma$ is again a command defined by

$$C * D =_{df} \lhd \,;\, (C \times D) \,;\, \rhd \ .$$

By definition of the underlying separation algebra, also $*$ is associative, commutative and has unit $\{(u, u)\}$. We will abstract, in the following, relations to elements of a modal Kleene algebra S. In particular, we use pairs of elements $c, d \in S$ in $c * d = \lhd(c \times d)\rhd$ and omit $\cdot$ before and after the brackets for better readability. As an abstract counterpart of the above exchange law we state the additional axiom

$$(c_1 \times d_1) \cdot (c_2 \times d_2) \ = \ c_1 \cdot c_2 \ \times \ d_1 \cdot d_2 \ . \tag{8}$$

Moreover, to characterise the interplay of $*$ with domain and codomain we assume validity of the following inequations for arbitrary c, d:

$$\ulcorner(c * d) \leq \ulcorner c * \ulcorner d \qquad \text{and} \qquad (c * d)\urcorner \leq \bar{c}\urcorner * \bar{d}\urcorner \ . \tag{9}$$

These laws are again abstract counterparts of valid relational variants. A proof in the relational model can be found in [17].

6.1. Characterising Hoare Triples

With the given algebraic background we now start characterising Hoare triples of SSL with partial correctness semantics abstractly. In contrast to usual Hoare logic, the triples in SSL come with an extra safety condition which makes them resource-sensitive. The general idea of this is to distinguish program abortion, e.g., due to a lack of required resources, from non-termination.

Definition 6.4. A command starting from a state σ *aborts* iff $\sigma\ C\ \bot$ where $\bot \in \Sigma$ denotes a distinguished state [19]. For command C and assertions p, q the *SSL Hoare triple* $\{p\}\, C\, \{q\}$ for partial correctness holds iff for all states $\sigma \in p$ both

$$\neg(\sigma\ C\ \bot) \quad \text{and} \quad \sigma\ C\ \sigma' \ \Rightarrow \ \sigma' \in q \quad \text{hold.}$$

Conceptually a non-terminating program relationally coincides with $\emptyset$, i.e., considering a starting state σ no final state can be obtained due to non-termination. Program abortion is identified by $(\sigma, \bot) \in C$ which means that an execution starts from σ but eventually gets stuck.

Generally, in modal idempotent semirings the algebraic semantics of *Hoare triples* can be given by

$$\{p\}\, c\, \{q\} \ \Leftrightarrow \ p \leq |c]q \ \Leftrightarrow \ \langle c|p \leq q \ \Leftrightarrow \ p \cdot c \leq c \cdot q \ \Leftrightarrow \ p \cdot c \cdot \neg q \leq 0 \ .$$

where assertions p, q can be realised using tests. This characterisation is too weak for SSL Hoare triples since the condition on program abortion is not considered. As before we start by modelling that condition in our presented relational approach and then turn to the abstract setting.

We introduce an extra command $\mathsf{abort} =_{df} \{(\bot, \bot)\}$ which is a test. To model basic commands in SSL we also define special commands D that respect abort, i.e., aborting executions in another command C will not be ruled out in $C\,;D$. Concrete examples are, e.g., the mutation, dereferencing or allocation commands in SSL [19].

Definition 6.5. A command c respects abort iff $\mathsf{abort} \cdot c = \mathsf{abort}$.

This corresponds to relations C that satisfy $\bot\, C\, \sigma \;\Rightarrow\; \sigma = \bot$. For such commands C the test $\bot$ is a left-annihilator. For treating $*$-compositions we need to extend the separation algebra operation $\bullet$ to also capture $\bot$ by

$$\sigma \bullet \tau = \bot \;\Leftrightarrow\; \sigma = \bot \vee \tau = \bot \; .$$

Hence $(\sigma, \tau) \rhd \bot \;\Leftrightarrow\; \sigma = \bot \vee \tau = \bot$. By this, we can infer useful laws in the relational model which we further abstract to the algebra.

Lemma 6.6. *For arbitrary c, d we have*

$$\begin{array}{rcl} (c * d) \cdot \neg\,\mathsf{abort} & = & (c \cdot \neg\,\mathsf{abort}\,) * (d \cdot \neg\,\mathsf{abort}\,) \; , \\ (c * d) \cdot \mathsf{abort} & = & (c \cdot \mathsf{abort}\,) * d + c * (d \cdot \mathsf{abort}\,) + (c \cdot \mathsf{abort}\,) * (d \cdot \mathsf{abort}\,) \; , \\ c * \mathsf{abort} & = & \mathsf{abort} \; . \end{array}$$

We assume these laws for the algebraic treatment in the following.

Lemma 6.7. abort *-respecting commands are closed under $+$, $\cdot$ and $*$.*

Now we are ready to characterise the safety condition in the Hoare triples of SSL, i.e., a state σ is safe w.r.t. a command C iff $\neg(\sigma\, C\, \bot)$ holds. In a point-free fashion, this can be formalised by $p \cdot \ulcorner(C \cdot \mathsf{abort}\,) \leq 0$. The test p includes at most those states from which C will not abort. Moreover the inequation is equivalent to $p \leq |C]\neg\,\mathsf{abort}$. Hence we can now give a pointfree characterisation of the set of safe states using the modal box operator. For better readability we abbreviate $\widehat{p} =_{df} p \cdot \neg\,\mathsf{abort}$ in the following. By this, we have $\widehat{1} = \neg\,\mathsf{abort}$ and the special case of Lemma 6.6: $\widehat{p * q} = \widehat{p} * \widehat{q}$.

Definition 6.8. By $\mathsf{safe}\,(c) =_{df} |c]\,\widehat{1}$, we characterise all *safe* states of an element c. We call a test p *safe for* c iff $p \leq \mathsf{safe}\,(c)$.

Note that the element $|c\rangle\widehat{1}$ would not be adequate to characterise safe states although it contains all starting states that will not end in $\bot$. To see this, consider as a simple example the relation $c = \{(\sigma, \bot), (\sigma, \tau)\}$. Clearly, we have $(\sigma, \sigma) \in |c\rangle\widehat{1}$, but still σ can lead to program abortion. Thus we need the box operator to state that all execution paths of c do not abort.

For a characterisation of SSL Hoare triples we use that $p \leq [c]\,\widehat{1}$ is equivalent to $|c\rangle p \leq \widehat{1}$. Hence, by a property of diamond we can immediately infer $\langle c|p \leq q \;\wedge\; \langle c|p \leq \widehat{1} \;\Leftrightarrow\; \langle c|p \leq \widehat{q}$. This form is still not fully adequate for our purposes due the asymmetry in excluding abort only in the assertion q. This asymmetry will falsify validity of the Hoare logic while inference rule. Thus, we define

Definition 6.9. A *partial correctness* Hoare triple in SSL is given by

$$\{p\}\, c\, \{q\} \;\Leftrightarrow_{df}\; \langle c|\widehat{p} \leq \widehat{q} \; .$$

Theorem 6.10. *All partial correctness inference rules of propositional Hoare logic remain valid under the partial correctness interpretation of Hoare triples with* abort *.*

A proof can be given analogously to the proof using the standard interpretation of Hoare triples without abortion in a modal Kleene algebra.

Another possibility for Definition 6.9 would be to use $\langle c|p \leq q \wedge q \leq \widehat{1}$ which implies the above condition and therefore is stronger. We will stay with the above definition since it is more compact and simpler to use.

One advantage of the above encoding of the Hoare triples is that they also imply that the test p involved always only characterises safe states.

Lemma 6.11. $\{p\}\, c\, \{q\}$ *implies $\widehat{p}$ is safe for c.*

Proof. We have, by Galois connection, isotony of box in its second argument, definition of $\mathsf{safe}\,(\_)$,

$$\langle c|\widehat{p} \leq \widehat{q} \;\Leftrightarrow\; \widehat{p} \leq |c]\widehat{q} \;\Rightarrow\; \widehat{p} \leq |c]\widehat{1} \;\Leftrightarrow\; \widehat{p} \leq \mathsf{safe}\,(c) \; .$$

□

This yields an abstract definition of the frame property [17] in a partial correctness setting. The frame property is an additional condition on commands in SSL to obtain validity of the prominent frame rule.

6.2. A Simple Proof for the Frame Rule

We start by giving an algebraic variant of the so-called *frame property*. Commands satisfying this property only depend on a certain set of resources, i.e., part of a state for a safe execution.

Definition 6.12. An element c has the *frame-property* iff

$$(\mathsf{safe}\,(c) \times 1)\rhd \cdot c \;\leq\; (c \times 1)\rhd\,.$$

In earlier work [2] this definition was used in combination with so-called *compensator* relations instead of the abstract identity relation 1. Such relations were used to model aliasing effects, e.g., on the set of shared variables. Like in [18], we follow a simplified approach that works with so-called *variable-as-resource* separation algebras. They do not require the usage of the above mentioned relations.

Using the above pointfree variant of the frame property with the modal encoding of Hoare triples, we can give an algebraic validity proof of the frame rule without requiring any further assumptions like safety monotonicity or any preservation property. We start by a localisation property that holds for elements c satisfying the frame property.

Lemma 6.13. *Assume c has the frame property. If $\widehat{p}$ is safe for c then*

$$\langle c|\widehat{p * r} \;\leq\; (\langle c|\widehat{p}) * \widehat{r}\,.$$

Proof. We have, by Lemma 6.6, definition of $*$ and backward diamond, Lemma 6.11 and (8), frame property, (8) and definition of $*$, (9) and since r is a test $\widehat{r}^{\urcorner} = r$, definition of backward diamond,

$$\langle c|\widehat{p * r} \;=\; \langle c|(\widehat{p} * \widehat{r}) \;=\; (\lhd(\widehat{p} \times \widehat{r})\rhd \cdot c)^{\urcorner} \;=\; (\lhd(\widehat{p} \times \widehat{r}) \cdot (\mathsf{safe}\,(c) \times 1)\rhd \cdot c)^{\urcorner} \leq (\lhd(\widehat{p} \times \widehat{r}) \cdot (c \times 1)\rhd)^{\urcorner} \;= \\ ((\widehat{p} \cdot c) * \widehat{r})^{\urcorner} \leq (\widehat{p} \cdot c)^{\urcorner} * \widehat{r} \;=\; (\,\langle c|\widehat{p}\,) * \widehat{r}\,.$$

□

Theorem 6.14. *If c has the frame property then the frame rule holds w.r.t. partial correctness.*

Proof. We have, by Lemma 6.13, assumption $\{p\}\,c\,\{q\}$, Lemma 6.6,

$$\langle c|\widehat{p * r} \leq (\langle c|\widehat{p}) * \widehat{r} \leq \widehat{q} * \widehat{r} \;=\; \widehat{q * r}\,.$$

□

Finally, we turn to the case of total correctness as in [2]. The semantics in this approach is : A state σ only belongs to the domain of a command iff there exists an execution starting from σ that does not abort and terminates in some final state τ. Hence, program abortion and non-termination are identified. As above the side conditions can be built in the definition of the SSL Hoare triples.

Definition 6.15. We define a *total correctness* SSL Hoare triple by

$$\{p\}\,c\,\{q\} \;\Leftrightarrow_{df}\; \langle c|p \leq q \;\wedge\; p \leq \ulcorner c\,.$$

Theorem 6.16. *All total correctness inference rules of standard Hoare logic remain valid under the total correctness interpretation of Hoare triples.*

Proof. We only prove the termination condition of the sequential composition rule $\{p\}\,c\,\{r\} \wedge \{r\}\,d\,\{q\} \Rightarrow \{p\}\,c \cdot d\,\{q\}$: by p being a test, property of domain, $\{p\}\,c\,\{r\}$ and isotony of domain, $\{r\}\,d\,\{q\}$ and isotony of domain, and (locality):

$$p \leq \ulcorner c \;\Rightarrow\; p \leq p \cdot \ulcorner c \;\Leftrightarrow\; p \leq \ulcorner(p \cdot c) \;\Rightarrow\; p \leq \ulcorner(c \cdot r) \;\Rightarrow\; p \leq \ulcorner(c \cdot \ulcorner d) \;\Leftrightarrow\; p \leq \ulcorner(c \cdot d)\,.$$

Validity proofs for the remaining inference rules can easily be obtained. □

Unfortunately, for a proof of the frame rule in the total correctness case one would have to additionally require $\ulcorner c * 1 \leq \ulcorner c$ as a point-free variant of a property called *termination monotonicity* [19] in the literature. Intuitively, if c terminates starting from a state σ it also will terminate starting from any possibly larger state $\sigma \bullet \tau$ assuming $\sigma \# \tau$. This property is in particular needed to calculate the termination condition in the consequence.

7. Overriding in Domain Modules

Feature Oriented Software Development (e.g. [5]) has been established in computer science as a general programming paradigm that provides formalisms, methods, languages, and tools for building maintainable, customisable, and extensible *software product lines (SPLs)* [20]. An SPL is a collection of programs that share a common part, e.g., functionality or code fragments. To encode an SPL, one can use *variation points (VPs)* in the source code. A VP is a location in a program whose content, called a *fragment*, can vary among different members of the SPL. A prominent example of an SPL is the Linux kernel, where `ifdef` directives are used as VPs. In [21] a *Structured Document Algebra (SDA)* is used to algebraically describe modules that include VPs and their composition. An SDA module is a collection of fragments named by VPs, and composition of SDA modules constructs programs.

In the next section we briefly recapitulate SDA.

7.1. Structured Document Algebra

VPs and Fragments. Let V denote a set of VPs at which fragments may be inserted and $F(V)$ be the set of *fragments* which may, among other things, contain VPs from V. Elements of $F(V)$ are denoted by $f_1, f_2, \ldots$. There are two special elements, a default fragment $\square$ and an error $\lightning$. An error signals an attempt to assign two or more non-default fragments to the same VP within one module. The addition, or supremum operator $+$ on fragments obeys the following rules:

$$\begin{array}{ll} \square + x = x \ , & \lightning + x = \lightning \ , \\ x + x = x \ , & f_i + f_j = \lightning \ \ (i \neq j) \ , \end{array}$$

where $x \in \{\square, f_i, \lightning\}$. This structure forms a flat lattice with least element $\square$ and greatest element $\lightning$. By standard lattice theory $+$ is commutative, associative and idempotent and has $\square$ as its neutral element.

Modules. A *module* is a partial function $m : V \rightsquigarrow F(V)$ with finite domain. VP v is *assigned* in m if $v \in dom(m)$, otherwise *unassigned* or *external*. Every assigned VP $v \in dom(m)$ has at least the default value $\square$ assigned to it.

Module Addition. The main goal of feature oriented programming is to construct programs step by step using reusable modules. In the algebra this is done by the module addition $+$. Addition fuses two modules while maintaining uniqueness (and signaling an error upon a conflict). Desirable properties for $+$ are commutativity and associativity. Since modules are partial functions, modules can be combined if they agree on VPs common to their domains.

For module addition, $+$ on fragments is lifted to partial functions:

$$(m+n)(v) =_{df} \begin{cases} m(v) & \text{if } v \in dom(m) - dom(n) \ , \\ n(v) & \text{if } v \in dom(n) - dom(m) \ , \\ m(v) + n(v) & \text{if } v \in dom(m) \cap dom(n) \ , \\ \text{undefined} & \text{if } v \notin dom(m) \cup dom(n) \ . \end{cases}$$

If in the third case $m(v) \neq n(v)$ and $m(v), n(v) \neq \square$ then $(m+n)(v) = \lightning$, thus signaling an error.

The set of modules forms a commutative monoid under $+$.

Deletion and Subtraction. For modules m and n the *subtraction* $m - n$ is defined as:

$$(m-n)(v) =_{df} \begin{cases} m(v) & \text{if } v \in (dom(m) - dom(n)) \ , \\ \text{undefined} & \text{otherwise.} \end{cases}$$

Subtraction satisfies the following laws:

$$\begin{array}{rcl@{\qquad\qquad}rcl}
dom(m-n) & = & dom(m) - dom(n)\ , & m-\emptyset & = & m\ , \\
\emptyset - n & = & \emptyset\ , & m-m & = & \emptyset\ , \\
(m+n)-p & = & (m-p)+(n-p)\ , & m-n & \subseteq & m\ , \\
m-(n+p) & = & (m-n)-p\ , & m \subseteq n & \Rightarrow & m-n=\emptyset\ .
\end{array}$$

Overriding. To allow *overriding* an operation | can be defined in terms of subtraction and addition. Module m overrides n, written $m \,|\, n$:

$$m \,|\, n \;=\; m + (n - m)$$

This replaces all assignments in n for which m also provides a value. | is associative and idempotent with neutral element $\emptyset$.

Modules m and n are called *compatible* if for all $v \in dom(m) \cap dom(n)$ we have $m(v) = n(v)$. The following properties are equivalent:

$$m \text{ and } n \text{ are compatible}\ , \qquad m \,|\, n = m + n\ , \qquad m \,|\, n = n \,|\, m\ .$$

With this, we have the following laws:

$$\begin{array}{ll}
m \,|\, (n+p) = (m \,|\, n) + (m \,|\, p)\ , & \text{(left distributivity)} \\
(m+n) \,|\, p = m \,|\, (n \,|\, p) & \\
\qquad \text{when } m \text{ and } n \text{ are compatible,} & \text{(sequentialisation)} \\
(m+n) \,|\, p = n \,|\, p & \\
\qquad \text{when } m \text{ and } n \text{ are compatible} & \\
\qquad \text{and } m \,|\, n = n, & \text{(absorption)} \\
m \,|\, (n+p) = n + (m \,|\, p) & \\
\qquad \text{when } dom(m) \cap dom(n) = \emptyset. & \text{(localisation)}
\end{array}$$

7.2. Abstracting from SDA

The set M of modules, i.e., partial maps $m : V \rightsquigarrow F(V)$, and $+$ and $-$, defined like in subsection 7.1, form an algebraic structure $SDA =_{df} (M, +, -, 0)$ which satisfies the following laws for all $l, m, n \in M$:

1. $(M, +, 0)$ is an idempotent and commutative monoid.
2. $(l - m) - n = l - (m + n)$.
3. $(l + m) - n = (l - n) + (m - l)$. (right distributivity)
4. $0 - l = 0$. (left annihilator)
5. $l - 0 = l$.

To reason about that structure with domain theory we will use an algebraic *module* [22] (not to be confused with the above SDA modules). This is a triple $(R, M, :)$ where : is an operation $R \times M \to M$, called *scalar product*, R is a ring and M is an Abelian group. Since we do not have a ring and a group, we will use a commutative and idempotent monoid together with a Boolean algebra.

Definition 7.1. A *mono module* is an algebra $(B, M, :)$ where $(M, +, 0)$ is a idempotent and commutative monoid and $(B, +, \cdot, 0, 1, \neg)$ is a Boolean algebra in which 0 and 1 are the least and greatest element and $\cdot$ and $+$ denote meet and join. Note that 0 and $+$ are overloaded, like in classical modules or vector spaces. The scalar product : is a mapping $B \times M \to M$ satisfying for all $p, q \in B$ and $a, b \in M$:

$$(p+q) : a = p : a + q : a\ , \qquad (10) \qquad\qquad (p \cdot q) : a = p : (q : a)\ , \qquad (13)$$

$$p : (a+b) = p : a + p : b\ , \qquad (11) \qquad\qquad 1 : a = a\ , \qquad (14)$$

$$0 : a = 0\ , \qquad (12) \qquad\qquad p : 0 = 0\ . \qquad (15)$$

Lemma 7.2. *Define, as for idempotent semirings,* $l \leq m \Leftrightarrow_{df} l + m = m$.

1. *Restriction* : *is isotone in both arguments.*

2. $p : a \leq a$.

Proof.

1. Follows from distributivity.

2. With $p \leq 1$ and Part 1 and Equation (14) we have: $p \leq 1 \Rightarrow p : a \leq 1 : a = a$

□

Lemma 7.3. *In a mono module Lemma 2.4 holds. Lemma 2.4.2 has the form*

$$p :(a \sqcap b) \;=\; p : a \;\sqcap\; b \;=\; p : a \;\sqcap\; p : b \;. \tag{16}$$

Proof.

1. Since we use a Boolean algebra there is nothing to prove for Part 1.

2. We start with the first equation. We have to show that $p :(a \sqcap b)$ is the greatest lower bound of $p : a$ and b. Since $p \leq 1$ we have $p :(a \sqcap b) \leq b$ and by isotony $p :(a \sqcap b) \leq p : a$. Therefore $p :(a \sqcap b)$ is a lower bound of $p : a$ and b. Now let c also be a lower bound of $p : a$ and b. Then $c \leq a \sqcap b$ since c is a lower bound of a by transitivity and Lemma 7.2.2. Moreover, $\neg p : c \leq \neg p :(p : a) = (\neg p \cdot p :)a = 0 : a = 0$ by Equation (13) and (12). Hence, by Equation (14) and (10) we have $c = (p + \neg p) : c = p : c + \neg p : c = p : c$. Therefore $c = p : c \leq p :(a \sqcap b)$, i.e., $p :(a \sqcap b)$ is indeed the greatest lower bound of $p : a$ and b. For the second equation we use idempotence of B, Equation (13) and the first equation:

$$\begin{aligned} p :(a \sqcap b) &= (p \cdot p) :(a \sqcap b) = p :(p : a \sqcap b) = p :(p : a \sqcap b) = p :(b \sqcap p : a) \\ &= p : b \sqcap p : a = p : a \sqcap p : b \;. \end{aligned}$$

□

Lemma 7.4. *Let M, N be sets. The algebra* RMM $=_{df}$ $(\mathcal{P}(M), \mathcal{P}(M \times N), :)$*, where* : *is restriction, i.e., $p : a = \{(x, y) \mid x \in p \wedge (x, y) \in a\}$, forms a mono module.*

The proof is straightforward.
We now extend mono modules with the predomain operator $^{\ulcorner} : M \to B$.

Definition 7.5. A *predomain mono module* $(B, M, :, ^{\ulcorner})$ is a mono module where $^{\ulcorner} : M \to B$ fulfills the analogues of (d1) and (d2), cf. Definition 2.5:

$$a \leq {}^{\ulcorner}a : a \;, \qquad {}^{\ulcorner}(p : a) \leq p \;.$$

Lemma 7.6. RMM *is a predomain mono module with* ${}^{\ulcorner}a = \{x \mid (x, y) \in a\}$.

Proof.
(d1): Assume $(x, y) \in a$. Then $x \in {}^{\ulcorner}a$ and therefore $(x, y) \in {}^{\ulcorner}a : a$.
(d2): Assume $x \in {}^{\ulcorner}(p : a)$. Then $x \in p$ and $(x, y) \in a$ for some $y \in N$. □

Having done this preliminary work, we can use an RMM over binary functional relations $R \subseteq M \times N$, i.e., $R^{\smile}; R \subseteq \mathsf{id}(M)$, to reason about SDA.

Lemma 7.7. *SDA's subtraction $m - l$ of modules is equivalent to $\neg {}^{\ulcorner}l : m$ in the corresponding* RMM.

Proof. We conclude

$$\begin{aligned}(x,y) \in m - l \quad &\Leftrightarrow \neg\exists z : (x,z) \in l \wedge (x,y) \in m \\ &\Leftrightarrow x \in \neg\lceil l \wedge (x,y) \in m \Leftrightarrow (x,y) \in \neg\lceil l : m\ .\end{aligned}$$

□

Now it is easy to verify that the SDA laws from the beginning of subsection 7.2 also hold in RMM. Note that the sides change, e.g., *right* distributivity becomes *left* distributivity. Further important properties of predomain also hold in a predomain module in analogous form.

Lemma 7.8. *Assume a predomain mono module* $(B, M, :, \lceil\ )$. *Then for all* $p \in B$ *and* $a, b \in M$*:*

1. $a = 0 \Leftrightarrow \lceil a = 0$.
2. $\lceil a \leq p \Leftrightarrow a \leq p : a$.
3. $p \leq \neg\lceil a \Leftrightarrow p : a \leq 0$.
4. $a \leq b \Rightarrow \lceil a \leq \lceil b$.
5. $\lceil(a + b) = \lceil a + \lceil b$.
6. $\lceil p : a \leq p \cdot \lceil a$.

The proofs are similar to the ones for a predomain IL-semiring.

SDA's overriding operator $m \mid n$ can also be defined in a predomain mono module: $b \mid a =_{df} b + \neg\lceil b : a$. In [23] this operator, embedded in a Kleene algebra, is used to update links in pointer structures.

Lemma 7.9. *Assume a predomain mono module* $(B, M, :, \lceil\ )$. *Then for all* $p \in B$ *and* $a, b \in M$*:*

1. $1 \mid a = 1$,
2. $b \leq b \mid a$,
3. $b = \lceil b : (b \mid a)$,
4. $\lceil(b \mid a) = \lceil b + \lceil a$,
5. $c \mid (a + b) = c \mid a + c \mid b$.

8. Conclusion

Our short tour through the various modal worlds ends here. We hope that the reader enjoyed the discovery ride, both through the novel results and/or views on the mentioned fields of application. We hope to have demonstrated that an algebraic treatment with modal structures allows a simple and unified presentation. Moreover this survey is intended to contribute a basis and an incentive for further case studies and applications along these lines.

Acknowledgements: This research was partially funded by the German Research Foundation (DFG) projects *MO 690/9-1* AlgSep — *Algebraic Calculi for Separation Logic* and *MO 690/7-2* FeatureFoundation.

References

[1] B. Möller, Modal Knowledge and Game Semirings, Computer Journal 56 (2013) 53–69.
[2] H.-H. Dang, P. Höfner, B. Möller, Algebraic Separation Logic, Journal of Logic and Algebraic Programming 80 (2011) 221–247.
[3] R. Glück, B. Möller, M. Sintzoff, A semiring approach to equivalences, bisimulations and control, in: R. Berghammer, A. Jaoua, B. Möller (Eds.), Relations and Kleene Algebra in Computer Science, volume 5827 of *Lecture Notes in Computer Science*, Springer, 2009, pp. 134–149.
[4] B. Möller, P. Roocks, M. Endres, An Algebraic Calculus of Database Preferences, in: J. Gibbons, P. Nogueira (Eds.), Mathematics of Program Construction, volume 7342 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2012, pp. 241–262.
[5] D. Batory, S. O'Malley, The design and implementation of hierarchical software systems with reusable components, ACM Transactions Software Engineering and Methodology 1 (1992) 355–398.
[6] G. Schmidt, T. Ströhlein, Relations and Graphs: Discrete Mathematics for Computer Scientists, Springer, 1993.
[7] G. Schmidt, Relational Mathematics, volume 132 of *Encyclopedia of Mathematics and its Applications*, Cambridge University Press, 2011.
[8] E. Manes, D. Benson, The inverse semigroup of a sum-ordered semiring, Semigroup Forum 31 (1985) 129–152.
[9] J. Desharnais, B. Möller, G. Struth, Kleene algebra with domain, ACM Transactions on Computational Logic 7 (2006) 798–833.
[10] M. Erné, J. Koslowski, A. Melton, G. Strecker, A primer on galois connections, in: S. A. et al. (Ed.), Papers on general topology and its applications. 7th Summer Conf. Wisconsin, volume 704 of *Annals New York Acad. Sci.*, pp. 103–125.
[11] B. Möller, G. Struth, Algebras of modal operators and partial correctness, Theor. Comput. Sci. 351 (2006) 221–239.

[12] B. Ganter, R. Wille, Formal concept analysis - mathematical foundations, Springer, 1999.
[13] X. Liu, W. Hong, J. Song, T. Zhang, Using formal concept analysis to visualize relationships of syndromes in traditional chinese medicine, in: D. Zhang, M. Sonka (Eds.), Medical Biometrics, volume 6165 of *Lecture Notes in Computer Science*, Springer, 2010, pp. 315–324.
[14] G. Schmidt, Rectangles, fringes, and inverses, in: R. Berghammer, B. Möller, G. Struth (Eds.), Relations and Kleene Algebra in Computer Science, volume 4988 of *Lecture Notes in Computer Science*, Springer, 2008, pp. 352–366.
[15] S. A. Ismail, A. Jaoua, Incremental pseudo rectangular organization of information relative to a domain, in: W. Kahl, T. G. Griffin (Eds.), Relational and Algebraic Methods in Computer Science, volume 7560 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 264–277.
[16] B. Möller, P. Roocks, An algebra of layered complex preferences, in: W. Kahl, T. Griffin (Eds.), Relational and Algebraic Methods in Computer Science, volume 7560 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2012, pp. 294–309.
[17] H.-H. Dang, B. Möller, Reverse Exchange for Concurrency and Local Reasoning, in: J. Gibbons, P. Nogueira (Eds.), MPC, volume 7342 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 177–197.
[18] C. Calcagno, P.-W. O'Hearn, H. Yang, Local Action and Abstract Separation Logic, in: Proc. of the 22nd Symposium on Logic in Computer Science, IEEE Press, 2007, pp. 366–378.
[19] H. Yang, P. O'Hearn, A semantic basis for local reasoning, in: M. Nielsen, U. Engberg (Eds.), Foundations of Software Science and Computation Structures, Proc. FOSSACS 2002, volume 2303 of *Lecture Notes in Computer Science*, Springer, 2002, pp. 402–416.
[20] R. Lopez-Herrejon, D. Batory, A standard problem for evaluating product-line methodologies, in: J. Bosch (Ed.), GCSE '01: Generative and Component-Based Software Engineering, volume 2186 of *Lecture Notes in Computer Science*, Springer, 2001, pp. 10–24.
[21] D. Batory, P. Höfner, B. Möller, A. Zelend, Features, Modularity, and Variation Points, Technical Report CS-TR-13-14, The University of Texas at Austin, 2013.
[22] N. Jacobson, Basic Algebra, volume I,II, Freeman, New York, 1985.
[23] T. Ehm, The Kleene Algebra of Nested Pointer Structures: Theory and Applications, Ph.D. thesis, Universität Augsburg, 2005.