Journal on Relational Methods in Computer Science, Vol. 1, 2004, pp. 93 - 131

Modal Kleene Algebra and Applications — A Survey —

Jules Desharnais[1], Bernhard Möller[2], and Georg Struth[2]

[1] Département d'informatique et de génie logiciel, Université Laval,
Québec QC G1K 7P4 Canada
Jules.Desharnais@ift.ulaval.ca

[2] Institut für Informatik, Universität Augsburg,
Universitätsstr. 14, D-86135 Augsburg, Germany
{moeller,struth}@informatik.uni-augsburg.de

Abstract Modal Kleene algebras are Kleene algebras with forward and backward modal operators, defined via domain and codomain operations. They provide a concise and convenient algebraic framework that subsumes various popular calculi and allows treating quite a number of areas. We survey the basic theory and some prominent applications. These include, on the system semantics side, wlp and wp calculus, PDL (Propositional Dynamic Logic), predicate transformer semantics, temporal logics and termination analysis of rewrite systems and state transition systems. On the derivation side we apply the framework to game analysis and greedy-like algorithms.

1 Introduction

Kleene algebras are fundamental structures in computer science, with applications ranging from program development and analysis to rewriting theory and concurrency control. Initially conceived as algebras of regular events [30], they have since been extended in several directions. The first direction includes omega algebra, which is a Kleene algebra with an additional operator for infinite iteration [8], demonic refinement algebra [56] and lazy Kleene algebra [35]. The second direction adds tests to Kleene algebra [31]. This allows, among others, reasoning about regular programs. Most of these extensions offer a nice balance between expressive and algorithmic power. The equational theory of Kleene algebra, for instance, can be decided by automata. The third direction is modal in spirit. Here Kleene algebra is combined with Boolean algebra in a module-based

Received by the editors March 14, 2004, and, in revised form, October 12, 2004.
Published on December 10, 2004.

approach [19], the scalar product modelling the application of a modal operator to a state. This yields a calculus similar to certain algebraic approaches to propositional dynamic logics.

The fourth direction we treat here reconciles the modal and the relational approaches to reasoning about programs and state transition systems in the form of Kleene algebra with domain [11]. The three simple equational domain axioms open a new door: they allow the definition of modal operators semantically via abstract image and preimage operations. But still in many cases expressions that mention modalities can be reduced to pure Kleene algebra with tests. This preserves the algorithmic complexity of the latter but also provides a very symmetric approach to reasoning about actions and propositions or transitions and states. Compared with relation algebra, modal Kleene algebra does not need the full power of a complete atomic Boolean algebra as the carrier set, of full additivity of sequential composition, of a converse operation and of residuation.

We survey modal Kleene algebra both from the theoretical and the practical point of view. On the theoretical side, we review the main concepts and the most important facets of a calculus. Modal Kleene algebras are mathematically quite simple: for actions, they provide only the regular operations of addition, multiplication and reflexive transitive closure; propositions are modelled by a Boolean algebra. Their combination via modalities makes the approach expressive enough for a wide variety of applications. We also try to point out that the algebraic approach to modal reasoning provides some advantages over a logical one. Algebra in general is particularly suited for structuring and abstracting. Here, structure is imposed via symmetries and dualities, for instance in terms of Galois connections. Abstraction is provided, for instance, by lifting modal expressions to the algebras of modal operators, which are again algebraically well-behaved. This often allows a very brief and concise point-free style of reasoning. We will also see that by exploiting modal correspondences, switching between relational and modal reasoning can be very simple in modal Kleene algebra. Often there is a one-to-one translation between modal and relational proofs. This is interesting in particular when relational reasoning is visualized by diagrams [17].

On the practical side, we show that modal Kleene algebra may serve both as an abstract program semantics and as a unifying tool that subsumes many popular program calculi and hence admits cross-theory reasoning. Here, we show that both the weakest liberal precondition semantics and the weakest precondition semantics and hence both partial and total program correctness can be modelled. We also show how predicate transformer algebras with convenient properties are induced. In the field of calculi, we present subsumption and completeness results for (propositional) Hoare logic, propositional dynamic logic and temporal logics.

In the field of system development we show applications to reasoning about greedy algorithms, to modelling termination conditions, to game analysis and in reconstructing a considerable part of the theory of abstract rewriting in a simple and convenient way. Since we do not consider equational rewriting but its non-symmetric extension [51], our results are immediately relevant to concurrent systems that interact via commutation or semi-commutation properties.

Many of the results presented here have appeared elsewhere, and we just quote the original papers which should be consulted for full details. Although the aim of this survey is to form an overall picture of the usefulness of modal Kleene algebra, we do not claim completeness. For instance, an approach to pointer analysis based on Kleene algebra [18], though highly relevant, is not treated.

Modal Kleene algebra is a quite recent development. Although the core of the theory seems now well understood and the examples outlined below point out its universality and practical relevance, still many questions are open. In particular, as far as applications are concerned, we feel that we have so far only scratched the surface.

The remainder of this text is organised as follows. Section 2 introduces modal semirings. Section 3 shows game analysis as a first application of modal semirings. Section 4 extends modal semirings to Kleene algebra with domain and to modal Kleene algebra. Section 5 relates the approach to propositional dynamic logic and its relatives. Section 6 shows how partial and total correctness of regular programs can be modelled. Section 7 lifts modal Kleene algebra to predicate transformer algebra. Section 8 relates the approach with temporal logics. Section 9 reconstructs various results from the area of termination analysis, including properties of abstract rewrite systems. Section 10 discusses connections to modal correspondence theory. Section 11 develops a generic greedy-like algorithm. Section 12 summarises the applications and points out further directions for the approach.

2 Domain Semirings and Modalities

2.1 Test Semirings and Domain

A *semiring* is a structure $(K, +, \cdot, 0, 1)$ such that $(K, +, 0)$ is a commutative monoid, $(K, \cdot, 1)$ is a monoid, multiplication distributes over addition from the left and right, and zero is a left and right annihilator, i.e., $a0 = 0 = 0a$ for all $a \in K$ (the operation symbol $\cdot$ is omitted here and in the sequel). The semiring is *idempotent* if it satisfies $a + a = a$ for all $a \in K$. Every idempotent semiring K has a *natural ordering* $\leq$ defined for all $a, b \in K$ by $a \leq b$ iff $a + b = b$. It induces a semilattice with $+$ as join and 0 as the least element; addition and multiplication are isotone with respect to the natural ordering.

In many contexts the semiring operations can be interpreted as follows:

$$\begin{array}{ccl} + & \leftrightarrow & \text{choice,} \\ \cdot & \leftrightarrow & \text{sequential composition,} \\ 0 & \leftrightarrow & \text{abortion,} \\ 1 & \leftrightarrow & \text{identity,} \\ \leq & \leftrightarrow & \text{increase in information or in choices.} \end{array}$$

Programs and state transition systems can be described in a bipartite world in which propositions describe sets of states and actions or events model state transitions. Propositions live in a Boolean algebra and actions in an idempotent semiring with the operations interpreted as above. Test operators embed the proposition space into the action space. To model regular programs, an additional operation of iteration or reflexive transitive closure is required; the corresponding extension of idempotent semirings to Kleene algebras is described in Section 4.

Finally, propositions and actions are connected by modal operators that map actions and propositions to propositions. To prepare this modal view, let semiring element a describe an action or abstract program and p describe a proposition or assertion, also called a *test*. Then pa describes a restricted program that acts like a when the initial state satisfies p and aborts otherwise. Symmetrically, ap describes a restriction of a in its possible final states. We introduce an abstract domain operator that assigns to a the test that describes precisely its enabling states. In combination with restriction, the domain operation yields an abstract preimage operation. This provides the semantic basis of modalities.

Let us now axiomatise the corresponding notions. A *Boolean algebra* is a complemented distributive lattice. By overloading, we usually write $+$ and $\cdot$ also for the Boolean join and meet operation and use 0 and 1 for the least and the greatest elements. The symbol $\neg$ denotes the operation of complementation. We will consistently use the letters $a, b, c, \dots$ for semiring elements and $p, q, r, \dots$ for Boolean elements. We will freely use the concepts and laws associated with Boolean algebra, including relative complement $p - q = p \sqcap \neg q$ and implication $p \to q = \neg p + q$.

A *test semiring* is a two-sorted structure $(K, \mathsf{test}(K))$, where K is an idempotent semiring and $\mathsf{test}(K) \subseteq K$ is a Boolean algebra embedded into K such that the operations of $\mathsf{test}(K)$ coincide with the restricted operations of K. In particular, $p \leq 1$ for all $p \in \mathsf{test}(K)$. In general, $\mathsf{test}(K)$ is only a subalgebra of the subalgebra of all elements below 1 in K.

A *semiring with domain* [11] (a $\ulcorner$-semiring) is a structure $(K, \ulcorner)$, where K is an idempotent semiring such that the *domain operation* $\ulcorner : K \to \mathsf{test}(K)$ satisfies for all $a, b \in K$ and $p \in \mathsf{test}(K)$

$$a \leq \ulcorner a\, a, \tag{d1}$$

$$\ulcorner(pa) \leq p. \tag{d2}$$

Let us explain these axioms. As in the algebra of relations, multiplication with a test from the left or right means domain or range restriction, respectively. Now first, since $\ulcorner a \leq 1$ by $\ulcorner a \in \mathsf{test}(K)$, isotonicity of multiplication shows that the first axiom can be strengthened to an equality expressing that restriction to the full domain is no restriction at all. The second axiom means that after restriction the remaining domain must satisfy the restricting test.

To further explain (d1) and (d2) we note that their conjunction is equivalent to each of

$$\ulcorner a \leq p \Leftrightarrow a \leq pa, \tag{llp}$$

$$\ulcorner a \leq p \Leftrightarrow \neg pa \leq 0, \tag{gla}$$

which constitute elimination laws for domain. (llp) says that $\ulcorner a$ is the least left preserver of a. (gla) says that $\neg\ulcorner a$ is the greatest left annihilator of a. Both properties obviously characterize domain in set-theoretic relations.

Because of (llp), domain is uniquely characterised by the two domain axioms. Moreover, if $\mathsf{test}(K)$ is complete then a domain operation always exists. If $\mathsf{test}(K)$ is not complete, this need not be the case.

A prominent example of a domain semiring is the algebra REL of binary relations over some set. There, the domain operation is given by $\ulcorner R = R\,;R^{\smile} \cap I$, where I is the identity relation, $R^{\smile}$ is the converse of R and ; is relational composition.

Further important domain semirings are the algebra PAT of path sets in a directed graph (see e.g. [34]) and Kleene's original algebra of formal languages, the latter ones being not very interesting, because its test algebra is *discrete*, i.e., consists of 0 and 1 only.

Many natural properties follow from the axioms. Domain preserves arbitrary existing suprema [37]; in particular, it is strict ($\ulcorner a = 0 \Leftrightarrow a = 0$), additive ($\ulcorner(a + b) = \ulcorner a + \ulcorner b$) and isotone ($a \leq b \Rightarrow \ulcorner a \leq \ulcorner b$). Moreover, it is stable on tests ($\ulcorner p = p$) and satisfies the import/export law ($\ulcorner(pa) = p\,\ulcorner a$). See [11] for further information.

2.2 Modal Semirings

A domain semiring is called *modal* if additionally it satisfies

$$\ulcorner(a\,\ulcorner b) \leq \ulcorner(ab). \tag{d3}$$

This axiom serves to make composition of multimodal operators below well-behaved. In a modal semiring, domain is *local*:

$$\ulcorner(ab) = \ulcorner(a\,\ulcorner b).$$

Without (d3), only the inequality $\ulcorner(ab) \leq \ulcorner(a\ulcorner b)$ holds. The additional axiom (d3) guarantees that the domain of ab is independent from the inner structure of b or its codomain; information about the domain of b in interaction with a suffices.

A codomain operation $\urcorner$ can easily be defined as a domain operation in the opposite semiring, where, as usual in algebra, opposition just swaps the order of multiplication. We call a semiring K with local domain and codomain a *modal semiring*.

In a modal semiring K, we can introduce forward and backward diamonds by modelling their standard semantics as abstract preimage and image operations:

$$|a\rangle p = \ulcorner(ap), \qquad \langle a|p = (pa)\urcorner, \tag{1}$$

for all $a \in K$ and $p \in \mathsf{test}(K)$.

The definition implies that the diamonds are strict additive mappings on the algebra of tests. Hence they are operators à la Jónsson and Tarski [27], and structures with such operators are called *modal algebras* in [22].

Duality with respect to opposition transforms forward diamonds into backward diamonds and vice versa. It follows that they satisfy an *exchange law*, a weak analogue of the relational Schröder law. For all $a \in K$ and $p, q \in \mathsf{test}(K)$,

$$|a\rangle p \leq \neg q \Leftrightarrow \langle a|q \leq \neg p. \tag{2}$$

De Morgan duality turns diamonds into boxes and vice versa:

$$|a]p \stackrel{\text{def}}{=} \neg|a\rangle\neg p, \qquad [a|p \stackrel{\text{def}}{=} \neg\langle a|\neg p.$$

It follows that diamonds and boxes are lower and upper adjoints of Galois connections:

$$|a\rangle p \leq q \Leftrightarrow p \leq [a|q, \qquad \langle a|p \leq q \Leftrightarrow p \leq |a]q, \tag{3}$$

for all $a \in K$ and $p, q \in \mathsf{test}(K)$. The Galois connections are useful as theorem generators and the dualities as theorem transformers. A Galois-based treatment of modal operators has also been given in [55].

The above-mentioned import/export law entails

$$p\,(|a\rangle q) = |pa\rangle q, \qquad p\,(\langle a|q) = \langle ap|q. \tag{4}$$

The modal axiom (d3) implies

$$\begin{array}{ll} |ab\rangle p = |a\rangle|b\rangle p, & \langle ab|p = \langle b|\langle a|p, \\ |ab]p = |a]|b]p, & [ab|p = [b|[a|p. \end{array} \tag{5}$$

Thus multiplication acts covariantly on forward modalities and contravariantly on backward ones. In the sequel, when the direction of diamonds and boxes does not matter, we will use the notation $\langle a\rangle$ and $[a]$. For a test p we have

$$\langle p\rangle q = pq, \qquad [p]q = p \to q. \tag{6}$$

Hence, $\langle 1\rangle = [1]$ is the identity function on tests. Moreover, $\langle 0\rangle p = 0$ and $[0]p = 1$.

Diamonds (boxes) commute with all existing suprema (infima) of the test algebra. These and further properties are implied by the Galois connections. They include cancellation laws and isotonicity and antitonicity properties for modalities. Of particular interest are the following demodalisation laws that follow from the domain elimination law (gla) and its dual for codomain.

$$|a\rangle p \le q \Leftrightarrow \neg q a p \le 0, \qquad \langle a|p \le q \Leftrightarrow p a \neg q \le 0. \tag{7}$$

Finally, diamond is disjunctive and box is antidisjunctive:

$$\langle a+b\rangle p \;=\; \langle a\rangle p + \langle b\rangle p, \qquad [a+b]p \;=\; ([a]p)([b]p). \tag{8}$$

To set up the connection to relational algebra, we define a *modal semiring with converse* to be a modal semiring K with an additional operation $\_^{\smile} : K \to K$ that is an involution, distributes over addition, is the identity on tests and is contravariant with respect to multiplication. One can show (see again [11]) that over a modal semiring with converse the axioms (d1) and (d2) imply the Galois connection

$$|a^{\smile}\rangle p \le q \Leftrightarrow p \le |a]q. \tag{9}$$

Therefore in a modal semiring with converse ${}^{\smile}$ we have

$$|a^{\smile}\rangle = \langle a|, \qquad |a^{\smile}] = [a|. \tag{10}$$

3 Two-Player Game Analysis

3.1 Introduction

To illustrate what we can already achieve with modal semirings, we take up part of the analysis of two-player games in [3,47]. Such a game is given by a set of positions with a binary relation describing the admissible moves. A position is *terminal* if it does not have a successor under the move relation. The two players take turns. A player whose turn it is but who is in a terminal position has lost the game. There are no special assumptions about positions and moves; in particular, the move relation need not be Noetherian.

The aim is to characterize positions that mean guaranteed win (under optimal play) or guaranteed loss (even under optimal play) and to compute a winning strategy if possible. We do not focus particularly on computing a winning strategy, which will nevertheless come as a byproduct from an algorithm for iteratively computing the winning and losing positions. The following conditions are obvious:

- Every terminal position is a losing position.
- A position is a losing position iff all moves from it lead to winning positions (for the opponent).
- A position is a winning position iff at least one move from it leads to a losing position (for the opponent).

We abstract from the relational case and represent the move relation as an element a of a modal semiring. Moreover, we want to represent terminal, winning and losing positions by semiring tests t, w and l. We obtain t from a as $t = \neg\ulcorner a$, whereas w and l are yet to be determined. To this end we rewrite the above informal conditions into modal notation:

$$t \leq l,$$

$$l = |a]\, w, \tag{11}$$

$$w = |a\rangle\, l. \tag{12}$$

Conditions (11) and (12) are mutually recursive. Separating them by substitution yields

$$l \;=\; |a]\,|a\rangle\, l, \qquad w \;=\; |a\rangle\,|a]\, w.$$

What kind of solutions do these recursive equations have?

3.2 Existence of Solutions: Fixpoints of Dual Functions

We define the functions

$$f(p) \stackrel{\text{def}}{=} |a]\,|a\rangle\, p, \qquad g(p) \stackrel{\text{def}}{=} |a\rangle\,|a]\, p.$$

By the properties of diamonds and boxes, both functions are isotone. We now assume that in the underlying modal semiring K the sublattice $\mathsf{test}(K)$ is complete. Then, by the Knaster/Tarski fixpoint theorem, f and g each have both a least and a greatest fixpoint. To investigate their relation we recall that two functions $h, k : M \rightarrow M$ on a Boolean lattice $(M, \leq)$ are *(de Morgan) duals* if for all $x \in M$

$$h(x) = \neg k(\neg x).$$

If the least fixpoints μ_h, μ_k and the greatest fixpoints ν_h, ν_k exist then $\mu_h = \neg\nu_k$ and $\mu_k = \neg\nu_h$ (see e.g. [42]). From this it is immediate that μ_h, μ_k and $z \stackrel{\text{def}}{=} \neg(\mu_h \sqcup \mu_k) = \nu_h \sqcap \nu_k$ form a partition of the lattice, i.e.,

$$\mu_h \sqcap \mu_k = \mu_h \sqcap z = \mu_k \sqcap z = \bot,$$
$$\mu_h \sqcup \mu_k \sqcup z = \top,$$

where $\bot$ and $\top$ are the least and greatest elements. Likewise, ν_h, ν_k and $\neg(\nu_h \sqcup \nu_k) = \mu_h \sqcap \mu_k$ form a partition of the lattice.

By definition, the functions $|a\rangle$ and $|a]$ are duals, and a quick calculation shows that the above functions f and g are duals as well. The set of positions is to be partitioned into winning, losing and tie positions. By the above observation there are two possible choices: either $l = \mu_f$ as the set of losing positions and $w = \mu_g$ as the set of winning positions, or $l = \nu_f$ and $w = \nu_g$.

In [3] it is shown that the first of these choices is the adequate one. The remainder $\nu_f \sqcap \nu_g = \nu_f \nu_g$ represents the set of tie positions, i.e., the set of positions from which under optimal play of both opponents none will reach a winning or losing position. Note that a tie position has to start at least one infinite path in the game graph; if the set of positions is finite, this path necessarily has to be cyclic.

One has to ensure that the (separately found) solutions $l = \mu_f$ and $w = \mu_g$ also satisfy the original mutual recursion

$$l = |a]\, w \qquad w = |a\rangle\, l$$

(which need not be the case for arbitrary fixpoints of f and g). This can be done by the rolling rule (see again [42]) of fixpoint calculus.

3.3 Iterative Computation of Win/Lose

We now want to obtain an algorithm for actually computing the winning and losing positions. For this we remember Kleene's fixpoint theorem, the proof of which shows that for an isotone function $h : M \rightarrow M$ on a complete lattice $(M, \leq)$ one has

$$\mathsf{sup}\,\{h^i(\bot) : i \in \mathbb{N}\} \leq \mu_h.$$

So let us consider the first steps of the fixpoint iteration for μ_f and μ_g. In the semiring setting we have $\bot = 0$; moreover, let $t = \neg\ulcorner a$ again be the set of terminal positions.

$$\begin{aligned} f^1(0) &= |a]\,|a\rangle\,0 = |a]\,0 \\ &= \neg\ulcorner a = t \\ f^2(0) &= f(f^1(0)) = f(t) \\ &= |a]\,|a\rangle\,t = |a]\,(g^1(0)) \\ &\ \ \vdots \\ f^{i+1}(0) &= |a]\,(g^i(0)) \end{aligned} \qquad \begin{aligned} g^1(0) &= |a\rangle\,|a]\,0 = |a\rangle\,t \\ &= |a\rangle\,(f^1(0)) \\ g^2(0) &= g(g^1(0)) = g(|a\rangle\,t) \\ &= |a\rangle\,|a]\,|a\rangle\,t = |a\rangle\,(f^2(0)) \\ &\ \ \vdots \\ g^{i+1}(0) &= |a\rangle\,(f^{i+1}(0)) \end{aligned}$$

This can be explained informally as follows. The set $f^1(0)$ of losing positions of "order 1" is the set t of terminal positions. The set $g^1(0)$ of winning positions of "order 1" consists of all immediate predecessors of t. The set $f^{i+1}(0)$ of losing positions of "order $i+1$" consists of the positions whose successors are all winning positions of "order i", and the set $g^{i+1}(0)$ of winning positions of "order $i+1$" consists of the positions that have at least one losing position of "order $i+1$" as a successor.

Hence the fixpoint iteration describes the following algorithm.

1. Start with the terminal positions marked as losing positions.
2. Traverse the game graph backwards and adapt the markings according to the above equations.

But what about termination of the algorithm? And under which circumstances does it really reach the least fixpoints $l = \mu_f$ and $w = \mu_g$? Obviously, for an infinite set of positions there will always be games for which the algorithm doesn't terminate. So we now restrict our attention to games with finitely many positions. This can abstractly be reflected by considering only modal semirings K in which all chains in $\mathsf{test}(K)$ are finite. Then all isotone functions are also continuous, and the fixpoint iteration yields the desired result when it gets stationary at a fixpoint. Recording in every iteration step which moves lead into winning or losing positions yields all possible winning strategies.

The basic fixpoint iteration algorithm reads as follows:

$r := 0$;
$\{\, \mathsf{inv}\ r \le f(r) \wedge r \le \mu_f\}$
$\mathsf{while}\ (f(r) \ne r)$
  $\mathsf{do}\ \ r := f(r)$;
  $\mathsf{od}\ \ \{r = \mu_f\}$

The least fixpoint $\mu_g = w$ of the second function g then results as $w = |a\rangle\, l$.

3.4 Efficiency Improvement

Let us now use this example to show that the algebra of modal semirings is also very useful in formal transformation of a basic algorithm into more efficient (but much less understandable) versions.

The main technique employed here is that of *formal differentiation* or *strength reduction* (see e.g. [43]), where expensive recomputation of a quantity in every step of an iteration is replaced by computation of the increments between the values of that quantity. By their many distributive laws, modal semirings are an ideal setting for this technique.

In the algorithm above, we first introduce an auxiliary variable s that always has the value $f(r)$ and is incremented correspondingly:

$r := 0\,;\, s := f(0)\,;$
$\{\,\mathsf{inv}\ s = f(r)\ \wedge\ r \leq s\ \wedge\ r \leq \mu_f\}$
$\mathsf{while}\ (s \neq r)$
$\quad\mathsf{do}\ \ (r,\, s) := (s,\ f(s))\ ;$
$\quad\mathsf{od}\ \ \{r = \mu_f\}$

Because of $r \leq s$ we have $s = r + (s - r)$ and $s \neq r\ \Leftrightarrow\ s - r \neq 0$. (This only needs isotonicity of f.) To simplify the assignment $s := f(s)$ we have to consider the special form of f. We obtain

$$\begin{aligned} f(s) &= f(r + (s - r)) = |a]\,|a\rangle\,(r + (s - r)) \\ &= |a]\,(|a\rangle\,r + |a\rangle\,(s - r)). \end{aligned} \tag{$*$}$$

Now we set $u = |a\rangle\,r$ and examine $|a]\,(u + x)$ for arbitrary x:

$$|a]\,(u + x) = \neg\ulcorner(a\neg(u + x)) = \neg\ulcorner(a\neg u\neg x) = |a\neg u]\,x.$$

If we now carry the part $a\neg u$ in a variable m, the assignment $s := f(s)$ becomes $s := |m]\,x$ with $x = |a\rangle\,(s - r)$. Our new invariant reads

$$\{\,\mathsf{inv}\ s = f(r)\ \wedge\ r \leq s\ \wedge\ r \leq \mu_f\ \ \wedge\ u = |a\rangle\,r\ \wedge\ m = a\neg u\}$$

This is established by the initialisation

$u := 0\,;\, m := a\ ;$

How to maintain it?

The calculation ($*$) shows that after the assignment $r := s$ variable u has to have the new value $u + x$, so m needs the new value

$$a\neg(u + x) = a\neg u\neg x = m\neg x$$

This yields

$r := 0\,;\, s := f(0)\,;$
$u := 0\,;\, m := a\,;$
$\{\,\mathsf{inv}\ s = f(r) \wedge r \leq s \wedge r \leq \mu_f \ \wedge u = |a\rangle\, r \wedge m = a\neg u\}$
$\mathsf{while}\ (s - r \neq 0)$
$\quad \mathsf{do}\ \ \mathsf{let}\ x = |a\rangle\,(s - r)$
$\qquad\quad \mathsf{in}\ \ (r,\, s,\, u,\, m) \ := \ (s,\, |m]\, x,\, u + x,\, m\,\neg x)\,;$
$\quad \mathsf{od}\ \ \{r = \mu_f \ \wedge u = \mu_g\}$

The simultaneous assignment can be sequentialised from left to right.

Our final improvement results from examining the expression involving m, namely $|m]\,x = \neg\ulcorner(m\,\neg x)$. Since we need $n \stackrel{\text{def}}{=} m\,\neg x$ anyway, it makes sense to compute n and $\neg\ulcorner n$ simultaneously.

For this we maintain a new variable d that always contains $\neg\ulcorner m$. It is initialised to $\neg\ulcorner a$ and is incrementally adjusted using a vector of out-degrees. Then, for each position $p \in x$ and every predecessor q of p under m,

1. decrease q's outdegree by 1 and remove the edge from q to p;
2. if the outdegree of q becomes 0 that way, add q to d.

Again, the corresponding program can be calculated algebraically.

4 Modal Kleene Algebras

While modal semirings suffice for some applications, others require an explicit notion of iteration. This is provided by extending idempotent semirings to Kleene algebras.

A *Kleene algebra* [30] is a structure $(K, {}^*)$ such that K is an idempotent semiring and the *star* * satisfies, for $a, b, c \in K$, the *unfold* and *induction laws*

$$1 + aa^* \leq a^*, \tag{$*$-1}$$
$$1 + a^*a \leq a^*, \tag{$*$-2}$$
$$b + ac \leq c \Rightarrow a^*b \leq c, \tag{$*$-3}$$
$$b + ca \leq c \Rightarrow ba^* \leq c. \tag{$*$-4}$$

Therefore, a^* is the least pre-fixpoint and the least fixpoint of the mappings $\lambda x.ax + b$ and $\lambda x.xa + b$. The star is isotone with respect to the natural ordering.

Two important consequences of these axioms are the laws

$$ba \leq ac \ \Rightarrow\ b^*a \leq ac^*, \qquad ab \leq ca \ \Rightarrow\ ab^* \leq c^*a. \tag{13}$$

A *Kleene algebra with tests (*KAT*)* is a test semiring $(K, \mathsf{test}(K))$ such that K is a Kleene algebra [31]. For all $p \in \mathsf{test}(K)$ we have that $p^* = 1$.

In a KAT one can model the (angelic) abstract semantics of regular programs as follows:

$$
\begin{aligned}
\mathsf{abort} &\stackrel{\text{def}}{=} 0\\
\mathsf{skip} &\stackrel{\text{def}}{=} 1\\
a \,[\,]\, b &\stackrel{\text{def}}{=} a + b\\
a \,;\, b &\stackrel{\text{def}}{=} ab\\
\mathsf{if}\ p\,\mathsf{then}\,a\,\mathsf{else}\,b &\stackrel{\text{def}}{=} pa + \neg pb\\
\mathsf{assert}\ p &\stackrel{\text{def}}{=} \mathsf{if}\ p\,\mathsf{then\,skip\,else\,abort} \quad = \quad p\\
\mathsf{while}\ p\,\mathsf{do}\,a &\stackrel{\text{def}}{=} (pa)^*\neg p
\end{aligned}
$$

The definition of $\mathsf{assert}\ p$ via $\mathsf{if\ then\,else}$ is the usual one from assertion macro packages in programming languages like *C* or *Java*; algebraically it simplifies to p alone.

A *Kleene algebra with domain (codomain)*, briefly $\ulcorner$-($\urcorner$-)Kleene algebra is a KAT in which the underlying test semiring is a domain (codomain) semiring. Finally, a *modal Kleene algebra (*MKA*)* is a KAT in which the underlying test semiring is modal.

Examples of MKAs are again REL and PAT.

Using the star induction axioms, one can show the following induction principle for the diamond operator (cf. [11]):

$$|a\rangle p + q \leq p \;\Rightarrow\; |a^*\rangle q \leq p. \tag{14}$$

Having now defined our setting, we will tie it in with various other calculi and present a number of applications.

5 Kleene Modules and PDL

Most previous algebraic approaches to modelling programs or state transition systems show an asymmetric treatment of propositions and actions. On the one hand, propositional dynamic logic (PDL) [23] and its algebraic relatives dynamic algebras [29,40,46] and test algebras [40,46,54] are proposition-based. Dynamic algebra has only modalities, test algebra also has propositions. Most axiomatisations do not even contain explicit axioms for actions: their algebra is only implicitly induced via the definitions of the modalities. On the other hand, KAT has both actions and propositions, but, complementarily to dynamic algebra, it lacks modalities, i.e., the possibility to combine actions and propositions into new propositions. Therefore, reasoning about actions in dynamic algebra and test algebra and about propositions in KAT is indirect and restricted.

These rather artificial asymmetries and limitations have already been questioned by Pratt [46], but persisted for several decades. They are overcome in MKA in a very smooth and simple way Therefore MKA provides an algebraic alternative to PDL that supports both proposition- and action-based reasoning and admits both tests and modalities. In a more abstract sense, MKA reconciles relational and modal reasoning about programs. However, the defining axioms of MKA are quite different from and more economic than those of dynamic algebra and test algebra. We will now briefly describe the precise relation between MKA and PDL and its algebraic relatives. This can best be done by introducing an additional intermediate structure which we call a *Kleene module*. Kleene modules are on the one hand straightforward adaptations of the standard modules of algebra that allow us to introduce modal operators via scalar products. On the other hand, the coupling between actions and propositions in Kleene modules is not as tight as in modal Kleene algebra.

5.1 Definition of Kleene Modules

Kleene modules are natural variants of the usual modules from algebra [26], where the ring is replaced by a Kleene algebra and the Abelian group by a Boolean algebra. Certain variants of Kleene modules have already been studied in [5,32].

A *Kleene left-module* $(K, B, :)$ consists of a Kleene algebra K, a Boolean algebra B and the *left scalar product* $:$, a mapping of type $K \times B \to B$, such that for all $a, b \in K$ and $p, q \in B$,

$$a : (p + q) = a : p + a : q, \tag{km1}$$
$$(a + b) : p = a : p + b : p, \tag{km2}$$
$$(ab) : p = a : (b : p), \tag{km3}$$
$$1 : p = p, \tag{km4}$$
$$0 : p = 0, \tag{km5}$$
$$q + a : p \leq p \Rightarrow a^* : q \leq p. \tag{km6}$$

As usual, we do not distinguish between the Boolean and Kleenean zeros and ones. In accordance with the relation-algebraic tradition, we also call the scalar products of Kleene modules *Peirce products*.

Axioms of the form (km1)–(km4) also occur in algebra. For rings, an analogue of (km5) is redundant, while for semirings — in absence of inverses — it is independent. Axiom (km6) is beyond ring theory. It is the star induction rule $(*\text{-}3)$ with the semiring product replaced by the Peirce product and the sorts of elements adjusted, i.e., b and c replaced by Boolean elements; it also corresponds to (14).

Analogously to the situation for domain and codomain we define *Kleene right-modules* as Kleene left-modules on the opposite semiring. A *Kleene bimodule* is a Kleene left-module that is also a Kleene right-module. We will henceforth consider only Kleene left-modules.

5.2 Calculus of Kleene Modules

The relation between Kleene left-modules and modal Kleene algebra is straightforward.

Proposition 5.1 *Let K be a modal Kleene algebra. Setting $a : p = |a\rangle p$, the structure $(K, \mathsf{test}(K), :)$ is a Kleene left-module.*

The left-module axioms and also the right-module axioms are easily seen to be theorems of modal Kleene algebra. Hence, these axioms yield further properties of MKA in a well-structured way.

We first present some further properties that do not mention the star. The scalar product is right-strict, i.e., $a : 0 = 0$, and left- and right-isotone. Hence, it is subconjunctive, $a : (pq) \leq (a : p)(a : q)$, and satisfies

$$a : p - a : q \leq a : (p - q).$$

The following Peirced variants of the star unfold laws ($*$-1) and ($*$-2) hold.

$$p + a : (a^* : p) = a^* : p, \qquad p + a^* : (a : p) = a^* : p. \tag{15}$$

Therefore, of course, these do not have to be explicitly added to the module axioms. Finally, the module axiom (km6), which is a quasi-identity, is equivalent to the identity

$$a^* : p - p \leq a^* : (a : p - p). \tag{16}$$

This identity appears in PDL (cf. [23]), but also in axiomatisations of temporal logics as an induction law. In [19], we present various additional properties that all translate easily to theorems of PDL.

5.3 Relatives of Kleene Modules

We now position the Kleene modules within the context of Kleene algebra with domain and algebraic variants of propositional dynamic logic.

First, the class of *dynamic algebras* [46] can be obtained as a variant of Kleene modules by requiring, instead of a Kleene algebra, an absolutely free algebra of

Kleenean signature (without 0 and 1), by removing (km4) and (km5), by adding right-strictness and the star unfold law of (15) and by replacing (km6) by (16). Consequently, the algebra of actions is implicitly axiomatised in dynamic algebra. We call a dynamic algebra or Kleene module *extensional* if

$$\forall\, p\,.\,(a:p \leq b:p) \Rightarrow a \leq b. \tag{17}$$

This property is independent of the module axioms. The relation induced by the left-hand side of this quasi-identity is a precongruence on Kleene modules. It can also be interpreted as a notion of *observational equivalence*. Intuitively, it is a point-wise measurement of the behaviour of actions. In the extensional case, the action is completely determined by its observations.

The following Theorem shows that Kleene modules subsume dynamic algebras and yield an exact representation of equational reasoning about Kripke frames.

Theorem 5.2

1. *Every Kleene module is a dynamic algebra.*
2. *The equational theories of extensional Kleene modules and extensional dynamic algebras coincide.*

Second, there are two extensions of dynamic algebras that also include tests. In Pratt's variant, the test axiom $p?:q = pq$ is added to the axioms of dynamic algebra, where ? models an embedding of tests into actions. Again, therefore, the Kleene algebra remains implicit.

Hollenberg [25] has given a variant of test algebra that explicitly uses the Kleene algebra axioms and also the embedding operator ?. This test algebra subsumes Pratt's variant. The connection between these approaches is made precise in the following theorem.

Theorem 5.3

1. *Every modal Kleene algebra is a Pratt test algebra.*
2. *The classes of modal Kleene algebras and Hollenberg test algebras coincide.*

We see two decisive advantages of modal Kleene algebra over test algebra. First, it inherits from KAT the notational economy of leaving the embedding ? is omitted. Second, it is it axiomatically more economy. It is defined via three axioms, whereas Hollenberg's test algebra has eight. On the other hand, it follows from results for Hollenberg's test algebra that the equational theory of extensional modal Kleene algebra is EXPTIME-complete.

For further technical details as well as further discussion of related work we refer to [19].

6 Modelling Program Correctness

6.1 Partial Correctness and wlp

We now return to the Kleene semantics of simple while programs introduced in Section 4. As is well known, *partial program correctness* can be modelled using the weakest liberal precondition $\mathsf{wlp}(a, q) = |a]\, q$. Then a *Hoare triple* $\{p\}\ a\ \{q\}$ is *valid* if $p \le |a]\, q$.

Kozen has shown that already in KAT one can formulate validity of $\{p\}\ a\ \{q\}$ as $pa\neg q = 0$, which by (gla) is equivalent to $\langle a|p \le q$ and hence to the above definition of validity by the Galois connection (3). Although this allows proving soundness of the rules of propositional Hoare logic, i.e., Hoare logic without the assignment rule, the MKA formulation leads to still simpler and readable encodings of Hoare triples and rules and also to more concise soundness proofs. Moreover, in contrast to KAT, the MKA formulation also admits a simple, fully algebraic proof of relative completeness of propositional Hoare logic [37].

Example 6.1 As an example consider the while-rule:

$$\frac{\{p \wedge q\}\ a\ \{q\}}{\{q\}\ \mathsf{while}\ p\ \mathsf{do}\ a\ \{\neg p \wedge q\}}$$

Its translation into MKA reads

$$\langle a|\,(pq) \le q \;\Rightarrow\; \langle (pa)^*\neg p|\, q \le \neg p\, q. \tag{18}$$

Now the soundness proof of this rule proceeds as follows:

$$\begin{aligned} \langle a|\,(pq) \le q &\Leftrightarrow \langle pa|\, q \le q \\ &\Rightarrow \langle (pa)^*|\, q \le q \\ &\Rightarrow \neg p\,(\langle (pa)^*|\, q) \le \neg p\, q \\ &\Leftrightarrow \langle (pa)^*\neg p|\, q \le \neg p\, q \end{aligned}$$

The first step uses the definition of diamond twice, the second one induction (14), the third one isotonicity, the fourth one import/export (4). An even shorter proof is possible in predicate transformer algebra (see Section 7). □

The result of encoding Hoare rules and showing that they are theorems of modal Kleene algebra can be expressed as follows.

Theorem 6.2 *Propositional Hoare logic is sound with respect to the modal Kleene algebra semantics.*

Of course, this is not surprising, since MKA subsumes KAT. However, the proof is more succinct. In fact, the specialised syntax of Hoare logic could easily be abandoned in favour of the simple and more universal algebraic calculus of modal Kleene algebra.

The demodalisation rules of modal Kleene algebra that arise as generalisations of (llp) and (gla) also yield a simple translation of the modal encoding of Hoare rules into KAT. The resulting formulas have a special shape and their validity can be decided by automata in PSPACE [9]. Thus the gain of expressiveness and flexibility introduced by MKA does not compromise the algorithmic complexity of KAT.

Using the MKA encoding of the weakest liberal precondition semantics for Hoare logic, one can carry out an entirely algebraic and fully formal relative completeness proof of propositional Hoare logic. This proof (see again [37]) is by far shorter than the standard textbook proofs that are based on set theory and usually leave many assumptions implicit.

Theorem 6.3 *Propositional Hoare logic is relatively complete for the partial correctness semantics of regular programs in modal Kleene algebra.*

6.2 Total Correctness and wp

For modelling *total correctness*, an MKA element a now receives the following interpretation: it abstractly represents a set of terminating computation paths, while its domain $\ulcorner a$ represents the set of starting states of these computations [13,14]. Under this interpretation, the weakest precondition is given by

$$\mathsf{wp}(a,q) \stackrel{\text{def}}{=} \ulcorner a\ \mathsf{wlp}(a,q),$$

the refinement relation by

$$c \sqsubseteq a \quad \Leftrightarrow \quad \ulcorner a \leq \ulcorner c \wedge \ulcorner a\, c \leq a.$$

This entails the following properties of the non-iterative angelic programming constructs:

$$\begin{aligned}
\mathsf{wp}(a,0) &= 0,\\
\mathsf{wp}(a,1) &= \ulcorner a,\\
\mathsf{wp}(\mathsf{abort},q) &= 0,\\
\mathsf{wp}(\mathsf{skip},q) &= q,\\
\mathsf{wp}(\mathsf{if}\ r\ \mathsf{then}\ a\ \mathsf{else}\ b,q) &= r\,\mathsf{wp}(a,q)+\neg r\,\mathsf{wp}(b,q),\\
\mathsf{wp}(a+b,q) &= \mathsf{wp}(a,q)\mathsf{wlp}(b,q)+\mathsf{wlp}(a,q)\mathsf{wp}(b,q).
\end{aligned}$$

The corresponding demonic programming constructs can be defined as follows:

– Demonic join (choice): $a \sqcup b \stackrel{\mathrm{def}}{=} \ulcorner a \ulcorner b\,(a+b)$.
– Demonic composition: $a \mathbin{\scriptstyle\Box} b \stackrel{\mathrm{def}}{=} ([a]\ulcorner b)\,ab$.

A demonic redefinition of loop is also possible, see [13,14] for details. These definitions imply the following properties that can all be shown by concise algebraic calculation. First, demonic refinement is the natural order associated with demonic choice, i.e., $a \sqsubseteq b \Leftrightarrow a \sqcup b = b$. Hence we have an upper semilattice (which is even complete if the underlying MKA is). Second, $\mathbin{\scriptstyle\Box}$ distributes through $\sqcup$ in both arguments and hence is $\sqsubseteq$-isotone in both arguments. Third, demonic composition is associative.

The above semantics is "fully demonic" in that one cannot model programs in which for certain states both termination and nontermination are possible. We show how approaches that solve this problem (e.g. [4,6,15,39,44]) can be represented in modal Kleene algebra.

The basic idea is to model a program as a pair consisting of a transition relation between states and a set of states from which no divergence is possible.

We again abstract to a modal Kleene algebra K and let the elements of K represent transition behaviour of programs, regardless of termination. Programs are then modelled by pairs (a,p) with $a \in K$ describing the state transition behaviour and $p \in \mathsf{test}(K)$ characterizing the states with guaranteed termination.

The essential program constructors are the following:

– Demonic composition: $(a,p) \mathbin{\scriptstyle\Box} (b,q) \stackrel{\mathrm{def}}{=} (ab,\, p\,(|a]q))$.
– Demonic choice: $(a,p) \bigsqcap (b,q) \stackrel{\mathrm{def}}{=} (a+b,\, pq)$.
– Angelic choice: $(a,p) \bigsqcup (b,q) \stackrel{\mathrm{def}}{=} (a+b,\, p+q)$.

Then $\mathbin{\scriptstyle\Box}$ is associative, has left annihilator $(0,0)$, neutral element $(1,1)$ and distributes through $\bigsqcap$. Both choices are idempotent and associative and distribute over each other. The refinement order is

$$(a,p) \sqsupseteq (b,q) \;\stackrel{\mathrm{def}}{\Leftrightarrow}\; (a,p) \bigsqcap (b,q) = (b,q) \;\Leftrightarrow\; a \le b \wedge p \ge q.$$

Both choice operators are isotone w.r.t. $\sqsupseteq$.

In Parnas's approach [44], the pairs (a,p) need to satisfy the restriction $p \le \ulcorner a$; it allows distinguishing the "must-termination" given by p from the "may-termination" given by $\ulcorner a$. However, it excludes "miraculous" program behaviour. Then there is no neutral element w.r.t. $\bigsqcap$, since the obvious candidate $(0,1)$ does not satisfy the restriction. So we do not have a full semiring structure. In Nelson's approach [39] this restriction is dropped, allowing miraculous programs like the pair $\mathsf{fail} = (0,1)$ that is guaranteed to terminate for all input states but at the same time never yields any output state. Now one obtains almost a semiring except that fail is only a left zero w.r.t. composition. This structure can be extended to a weaker form of modal Kleene algebra; the details are the subject of a forthcoming paper [38].

As an example for the use of the MKA laws in this setting, we prove associativity of composition. It is immediate that it suffices to consider the second components of the pairs, for which we calculate:

$$p\,|a](q\,|b]r) \;=\; p\,(|a]q)\,(|a]|b]r) \;=\; p\,(|a]q)\,(|ab]r).$$

The first step uses conjunctivity of $|a]$, the second one locality. The proofs of the other properties mentioned are slightly longer but again entirely straightforward calculations using the laws of modal Kleene algebra.

7 Beyond PDL: Predicate Transformer Algebras

Assume a test semiring $(K,+,,0,1)$. A *predicate transformer* is a function $f : \mathsf{test}(K) \to \mathsf{test}(K)$. It is *disjunctive* if $f(p+q) = f(p)+f(q)$ and *conjunctive* if $f(pq) = f(p)f(q)$. It is *strict* if $f(0) = 0$. Finally, id is the identity transformer and $\circ$ denotes function composition.

Let P be the set of *all* predicate transformers, M the set of isotone and D the set of strict and disjunctive ones. Under the pointwise ordering $f \le g \overset{\text{def}}{\Leftrightarrow} \forall\, p\,.\, f(p) \le g(p)$, P forms a lattice where the supremum $f + g$ and infimum $f \sqcap g$ of f and g are the standard pointwise liftings of $+$ and $\cdot$. We will also use the pointwise liftings of $-$ and $\to$ to the operator level. The least element of P (and M, D) is the constant 0-valued function $\mathbf{0}(p)$. The structure $(D,+,\circ,\mathbf{0},id)$ is an idempotent semiring. In fact, in its left argument $\circ$ even preserves arbitrary existing suprema and infima.

If $\mathsf{test}(K)$ is a complete Boolean algebra then P is a complete lattice with D as a complete sublattice. Hence we can extend D by a star operation via a least fixpoint definition:

$$f^* \overset{\text{def}}{=} \mu g\,.\, id + f \circ g,$$

where μ is the least-fixpoint operator. Now D satisfies the Kleene algebra axioms except the second star induction law ($*$-4). Only the subalgebra of universally disjunctive predicate transformers is a full Kleene algebra.

Many properties of modal operators can now be presented much more succinctly. First, the test-level Galois connections (3) can be lifted to operators $f, g : \mathsf{test}(K) \to \mathsf{test}(K)$:

$$|a\rangle f \le g \Leftrightarrow f \le [a|g, \qquad \langle a|f \le g \Leftrightarrow f \le |a]g, \tag{19}$$

for all $a \in K$. From this we get the cancellation and shunting laws

$$|a\rangle[a| \leq \langle 1\rangle \leq [a||a\rangle, \qquad \langle a||a] \leq \langle 1\rangle \leq |a]\langle a|, \tag{20}$$

$$f|a] \leq g \Leftrightarrow f \leq g\langle a| \qquad f[a| \leq g \Leftrightarrow f \leq g|a\rangle. \tag{21}$$

Semiring expressions inside of operators can be decomposed by the laws

$$\langle a+b\rangle = \langle a\rangle + \langle b\rangle, \qquad |ab\rangle = |a\rangle|b\rangle, \qquad \langle ab| = \langle b|\langle a|,$$

$$[a+b] = [a] \sqcap [b], \qquad |ab] = |a]|b], \qquad [ab| = [b|[a|.$$

The decomposition with respect to multiplication is covariant for forward modalities and contravariant for backward modalities. This results from the symmetry between domain and codomain via opposition. The decomposition can be used to transform expressions into normal form and to reason entirely at the level of modal algebra in the sense of [22].

Diamonds are isotone, i.e., $a \leq b$ implies $\langle a\rangle \leq \langle b\rangle$. Dually, boxes are antitone, i.e., $a \leq b$ implies $[b] \leq [a]$.

In the case of an MKA, the algebras of operators can be extended to KAs because of the following unfold and induction laws at the operator level (cf. [11]).

$$|1\rangle + |a\rangle|a^*\rangle \leq |a^*\rangle, \qquad |1\rangle + |a^*\rangle|a\rangle \leq |a^*\rangle, \tag{22}$$

$$f + |a\rangle g \leq g \Rightarrow |a^*\rangle f \leq g. \tag{23}$$

Setting $f = g = \langle 1\rangle$ we obtain, from the analogue of this for the backward diamond,

$$\langle a| \leq \langle 1\rangle \;\Rightarrow\; \langle a^*| \leq \langle 1\rangle. \tag{24}$$

These laws for the "inner star" induce an "outer star" $|a\rangle^*$ that coincides with $|a^*\rangle$ and turns the algebra of boxes into a left weak Kleene algebra. Analogous laws hold for the backward modal operators.

Next we give lifted versions of the commutation properties (13). The first of these becomes, for the forward diamond,

$$|b\rangle f \leq f|c\rangle \;\Rightarrow\; |b^*\rangle f \leq f|c^*\rangle; \tag{25}$$

it is easily shown using (23). The second one lifts only for the case where f is a diamond:

$$\langle a||b\rangle \leq |c\rangle\langle a| \;\Rightarrow\; \langle a||b^*\rangle \leq |c^*\rangle\langle a|. \tag{26}$$

This is established by shunting the two occurrences of $\langle a|$ in the conclusion of this implication to the respective other side of the inequation using (19) and (21) and then again using (23).

The restriction on the second semicommutation laws entails that not all desirable properties can be shown in a pointfree manner. We demonstrate this with the soundness proof for the while-rule of the propositional Hoare calculus. Proof obligation (18) translates into

$$\langle pa|f \leq f \;\Rightarrow\; \langle\neg p|\langle(pa)^*|f \leq \langle\neg p|f.$$

For a diamond f, but not generally, this follows by neutrality of $\langle 1\rangle$, (26) and isotonicity.

8 Beside PDL: Temporal Logic

While propositional dynamic logic contains explicit statements for actions or programs and therefore allows one to compare different programs, temporal logics reason about runs of one particular program at a time. This is particularly interesting for the analysis of concurrent programs and reactive systems, which need not terminate. Originally, temporal logics used Prior's future tense modality G with the reading "at all future states including the present one", F with the reading "at some future state including the present one" and X with the reading "at the next state". Later, the binary operator U was added with the reading $p\,\mathsf{U}q$ as "p until q", i.e., "q will eventually be true and till then p will be true". This system is also known as propositional linear temporal logic.

It is well known that these temporal operators can be defined in PDL, whence also in MKA. For abstract program a,

$$\mathsf{X} = |a\rangle, \tag{27}$$
$$\mathsf{F} = |a^*\rangle, \tag{28}$$
$$\mathsf{G} = |a^*], \tag{29}$$
$$p\,\mathsf{U} = |(pa)^*\rangle. \tag{30}$$

Of course, X, F and G can also be defined in Kleene modules, whereas U requires a product of a test and an action which cannot be expressed there. It is obvious that, interpreted over traces, these operators have the desired semantics. It follows immediately that $\mathsf{F} = 1\mathsf{U}$ that $\mathsf{G} = \neg\mathsf{F}\neg$ and that — by the unfold laws for the Kleene star — the following unfold laws for eventually and until hold:

$$\mathsf{F} = |1\rangle + \mathsf{X}\,\mathsf{F}, \tag{31}$$
$$p\,\mathsf{U} = |1\rangle + (|p\rangle \sqcap \mathsf{X}\,(p\,\mathsf{U})). \tag{32}$$

Manna and Pnueli [33] have axiomatized linear temporal logic (LTL). More recently, von Karger [55] has derived these axioms as theorems in a much leaner formalism called *temporal algebra* that defines modal operators by Galois connections similar to ours over a complete Boolean algebra and uses the Theorem of Knaster and Tarski to model iteration via fixpoints on this algebra. The reconstruction by von Karger also provides a nice modular presentation of the LTL axioms. Some of them are general laws of modal logic. They therefore hold a fortiori in MKA. Some further axioms, like the above until law, are fixpoint properties and hence hold not only for von Karger's calculus, but also for the more general case of MKA. In particular, all the laws that do not involve U even hold in Kleene modules. A particular instance of such a law is

$$|a^*](p \to |a]p) \leq |a^*](p \to |a^*]p), \tag{33}$$

which can be obtained by dualising the induction law (16).

There is, however, a series of LTL axioms that depend on the particular structure of models and the way that temporal formulas are interpreted over runs of a program. Also here, we can immediately generalise von Karger's reconstruction to MKA. Von Karger shows, for instance, that some further LTL axioms are *implied* in models that satisfy a confluence property. We will discuss this kind of property extensively in Section 9.5. Some further axioms are implied in models in which every state has precisely one successor state. This can be expressed using the well-known properties of being a partial function or *simple* (or *deterministic*) and being total or *entire* (cf. [20]). This can be expressed in MKA as

$$\langle a||a\rangle \leq \langle 1\rangle, \qquad\qquad \langle 1\rangle \leq |a\rangle\langle a|. \tag{34}$$

The element a is a *map* if it is simple and entire. For maps, in particular, $|a\rangle = |a]$, which is a direct translation of the LTL axiom $\neg\mathsf{X} = \mathsf{X}\neg$, and $|a\rangle 1 = 1$. A co-simplicity property is also imposed on backward modalities, whereas this is not the case for entirety. Just in contrast, the model of linear temporal logic is assumed to be a discrete linear ordering with a left but with no right endpoint. It remains to model the initial state.

Intuitively, a test p characterises the initial states of element a if it is contained in the complement of the codomain of a, i.e., $p \leq \neg\langle a|1 = [a|0$. Dually, as we have seen in Section 3.1, p characterises the terminal states of a if it is contained in the complement of the domain of a, i.e., $p \leq \neg|a\rangle 1 = |a]0$. Terminality, however, is of no further interest here. Let now init be the greatest such element, i.e.,

$$\mathsf{init}_a \stackrel{\text{def}}{=} [a|0.$$

This initiality test is important for modelling validity of a temporal implication $p \supset q$ as $\mathsf{init}_a \cdot p \leq q$.

Von Karger's completeness result for propositional linear temporal logic can then easily be generalised to modal Kleene algebra.

Theorem 8.1 *Modal Kleene algebra has the basic axioms of Manna and Pnueli for propositional linear temporal logic as theorems. The additional conditions for linearity of models and validity of temporal implication can be expressed in modal Kleene algebra.*

Von Karger also sketches a completeness result for computational tree logic; we conjecture that this can also be generalised to MKA.

9 Termination Analysis

9.1 Termination in Modal Kleene Algebra

We now deal with the question whether a transition system admits infinite transition paths. To this end we abstract a notion of termination for modal semirings from set-theoretic relations.

According to the standard definition, a relation R on a set A is well-founded iff every non-empty subset of A has an R-minimal element. In a $\ulcorner$- semiring S, the minimal part of $p \in \mathsf{test}(S)$ w.r.t. some $a \in K$ can algebraically be characterized as $p - \langle a|p$, i.e., as the set of points that have no a-predecessor in p. So, by contraposition, the well-foundedness condition holds iff for all $p \in \mathsf{test}(K)$ one has $p - \langle a|p \leq 0 \;\Rightarrow\; p \leq 0$. Using Boolean algebra we therefore obtain the following abstract characterization of well-foundedness and its dual, Noethericity.

Let S be a modal semiring. An element $a \in S$ is *well-founded* if for all $p \in \mathsf{test}(S)$,

$$p \leq \langle a|p \Rightarrow p \leq 0, \tag{35}$$

An element $a \in S$ is *Noetherian* if for all $p \in \mathsf{test}(S)$,

$$p \leq |a\rangle p \Rightarrow p \leq 0. \tag{36}$$

Similar definitions in related structures have been given in [1,16,22]. By de Morgan duality, a is Noetherian iff, for all $p \in \mathsf{test}(K)$,

$$|a]p \leq p \Rightarrow 1 \leq p. \tag{37}$$

It is easy to prove some of the well-known properties of well-founded and Noetherian relations in modal Kleene algebra [11]. First, 0 is the only Noetherian test. Second, the property of being Noetherian is downward closed. Third, every Noetherian element is irreflexive and non-dense, provided it is non-trivial. Fourth, an element is Noetherian iff its transitive closure is, but no reflexive transitive closure is Noetherian. Finally, Noethericity of a sum implies Noethericity of its components, whereas the converse direction does not hold in general. We will later present commutativity conditions that enforce this converse implication.

9.2 Termination via Löb's Formula

We now investigate two alternative equational characterisations of Noethericity. The first one uses the star. The second one is without the star. It holds for the special case of a *transitive* Kleenean element a, i.e., when $aa \leq a$.

Let K be a $^\ulcorner$-semiring. Consider the equations

$$|a\rangle \leq |a\rangle^{+}(|1\rangle - |a\rangle), \tag{38}$$

$$|a\rangle \leq |a\rangle(|1\rangle - |a\rangle). \tag{39}$$

The equation (39) is a translation of Löb's formula from modal logic (cf. [7]) which expresses well-foundedness in Kripke structures. We say that a is *pre-Löbian* if it satisfies (38). We say that a is *Löbian* if it satisfies (39).

In the relational model, Löb's formula states that a is transitive and that there are no infinite a-chains. We will now relate Löb's formula and Noethericity.

Theorem 9.1 *Assume a modal Kleene algebra.*

1. *Every Löbian and every pre-Löbian element is Noetherian.*
2. *Every Noetherian element is pre-Löbian.*
3. *Every transitive and Noetherian element is Löbian.*

As an example, we prove property 2. Proofs of the other two properties can be found in [12].

Proof. Let K be an MKA and let $a \in K$. Let $f = |a\rangle$ and $g(p) = p - f(p)$.

Let a be pre-Löbian, which is equivalent to $f - f^{+}g \leq |0\rangle$. Assume $p \leq f(p)$, i.e., $p - f(p) \leq 0$, i.e., $g(p) \leq 0$. We must show that $p \leq 0$. We calculate

$$p \leq f(p) = f(p) - f^{+}(0) = f(p) - f^{+}g(p) = 0.$$

The second step uses strictness of diamonds. The third step uses the assumption on g. The fourth step uses the assumption that a is pre-Löbian.

Let a be Noetherian. Then a is pre-Löbian if we can show that $f - f^{+}g \leq f(f - f^{+}g)$. We calculate

$$\begin{aligned} f - f^{+}g &= f - ff^{*}g \\ &\leq f(|1\rangle - f^{*}g) \\ &= f(|1\rangle - (|1\rangle + f^{+})g) \\ &= f(|1\rangle - (g + f^{+}g)) \\ &= f((|1\rangle - g) - f^{+}g) \\ &\leq f(f - f^{+}g). \end{aligned}$$

The first step uses the definition of f^+. The second step uses the identity $f(p) - f(q) \leq f(p-q)$ which holds for every additive mapping on a Boolean algebra. The fifth step uses the Boolean identity $p - (q + r) = (p - q) - r$. The last step uses isotonicity and the fact that $|1\rangle - g = |1\rangle - (|1\rangle - f) \leq f$. This follows from the Boolean identities $p - (p - q) = pq \leq q$. □

Properties 1. and 3. already hold in $^\ulcorner$-semirings. A closer analysis of the proof shows that in 3. it suffices to assume that a is *weakly transitive*, i.e,

$$|aa\rangle \leq |a\rangle. \tag{40}$$

This is a much weaker requirement than transitivity $aa \leq a$. To see this, view the Kleene elements again as sets of computation paths. If a consists of paths with exactly two states each (i.e., is isomorphic to a binary relation on states) then aa consists of paths with exactly three states, and so $aa \leq a$ holds only if $aa = 0$. But a is still weakly transitive if it is transitive considered as a binary relation.

The calculational translation between the Löb-formula and our definition of Noethericity is quite interesting for the correspondence theory of modal logic (see also Section 10). In this view, our property of Noethericity expresses a frame property, which is part of semantics, whereas the Löb formula stands for a modal formula, which is part of syntax. In modal semirings, we are able to express syntax and semantics in one and the same formalism. Moreover, while the traditional proof of the correspondence uses model-theoretic semantic arguments based on infinite chains, the algebraic proof is entirely calculational and avoids infinity. This can be quite beneficial for mechanisation.

9.3 Termination via Infinite Iteration

Cohen has extended Kleene algebra with an $^\omega$ operator for modelling infinite iteration [8]; he has also shown applications in concurrency control. In [53], this algebra has been used for calculating proofs of theorems from abstract rewriting that use simple termination assumptions.

Dually to the Kleene star, the omega operator is defined as a greatest post-fixpoint. An *ω-algebra* is a structure (K, ω) where K is a Kleene algebra and

$$a^\omega \leq aa^\omega, \tag{41}$$

$$c \leq ac + b \Rightarrow c \leq a^\omega + a^*b, \tag{42}$$

for all $a, b, c \in K$. Hence, a^ω is also the greatest fixpoint of $\lambda x.ax$.

Like in Section 7, for an MKA K it seems interesting to lift (41) and (42) to operator algebras, similar to the laws (22), and (23) for the star. This is very simple for (41): for $a \in K$,

$$|a^\omega\rangle \leq |a\rangle|a^\omega\rangle. \tag{43}$$

However, there is no law corresponding to (23) and (42). The proof of (23) uses (llp) and works, since the star occurs at the left-hand sides of inequalities. There is no similar law that allows us to handle an omega that occurs at right-hand sides of inequalities. But instead, one can axiomatise the greatest fixpoint $\nu|a\rangle$ of $|a\rangle$ for $a \in K$ by

$$\nu|a\rangle \leq |a\rangle\, \nu|a\rangle, \tag{44}$$

$$p \leq |a\rangle p + q \Rightarrow p \leq \nu|a\rangle + |a^*\rangle q. \tag{45}$$

If $\mathsf{test}(K)$ is complete then, by the Knaster-Tarski theorem, $\nu|a\rangle$ always exists, since $|a\rangle$ is isotone. Then one can use a weaker axiomatisation (see [22]) from which (45) follows by greatest fixpoint fusion.

The test $\nu|a\rangle$ measures potential infinity, whereas a^ω measures actual infinity.It even turns out (see the end of this section) that $\nu|a\rangle$ is more suitable for termination analysis than a^ω.

Since $|a\rangle p = \neg|a]\neg p$, existence of $\nu|a\rangle$ also implies existence of the least fixpoint $\mu|a]$ of $|a]$, since $\mu|a] = \neg\nu|a\rangle$. In the modal μ-calculus, $\mu|a]$ is known as the *halting predicate* (see, e.g., [23]). With the help of $\nu|a\rangle$ we can rephrase Noethericity more concisely as

$$\nu|a\rangle = 0. \tag{46}$$

As an immediate consequence of this we obtain

Corollary 9.2 *Define, for fixed $q \in \mathsf{test}(K)$ and $a \in K$, the function $f : \mathsf{test}(K) \to \mathsf{test}(K)$ by $f(p) = q + |a\rangle p$. If $\nu|a\rangle$ exists and a is Noetherian then f has the unique fixpoint $|a^*\rangle q$.*

A notion of guaranteed termination can easily be defined in ω-algebra as the absence of infinite iteration. We call *a ω-Noetherian* if $a^\omega \leq 0$.

We now study how Noethericity and ω-Noethericity relate. Analogously to (17) we call a $\ulcorner$-Kleene algebra K *extensional* if

$$|a\rangle \leq |b\rangle \Rightarrow a \leq b \tag{47}$$

holds for all $a, b \in K$. Note that the language model is not extensional. The following lemma shows that the relation between Noethericity and ω-Noethericity does not depend on extensionality. This is somewhat surprising, since set-theoretic relations are extensional and in the relational model the two notions coincide.

Lemma 9.3

1. *Every Noetherian element of an ω-algebra with domain is ω-Noetherian.*

2. *There is an ω-algebra with domain with an ω-Noetherian, but not Noetherian element.*
3. *There is a non-extensional ω-algebra with domain in which all ω-Noetherian elements are Noetherian.*
4. *There is an extensional ω-algebra with domain in which all ω-Noetherian elements are Noetherian.*
5. *In every extensional omega algebra K with domain one can extend the diamond algebra $|K\rangle$ to an omega algebra by setting $|a\rangle^\omega = |a^\omega\rangle$.*

For the proof see [12].

Thus ω-algebra does not entirely capture the standard notion of termination.

We now study the exhaustive finite iteration of an element $a \in K$, given by

$$\mathsf{exh}\, a \stackrel{\text{def}}{=} \mathsf{while}\, \ulcorner a\, \mathsf{do}\, a \;=\; a^* \neg \ulcorner a. \tag{48}$$

Then we can represent the set of points from which a terminal point can be reached via a-steps as

$$\ulcorner(\mathsf{exh}\, a) \;=\; \ulcorner(a^* \neg \ulcorner a) \;=\; |a^*\rangle \neg \ulcorner a. \tag{49}$$

Proposition 9.4 *If a is Noetherian then $\ulcorner(\mathsf{exh}\, a) = 1$, i.e., from every starting point a terminal point can be reached.*

For the proof see again [12]. This shows again that modal Kleene algebra is more adequate for termination analysis than omega algebra. To see this, consider the algebra LAN of formal languages which is both an omega algebra and an MKA with complete test algebra $\mathsf{test}(\mathrm{LAN}) = \{0, 1\}$. In LAN we have $|a\rangle 1 = \ulcorner a = 1 \neq 0$ when $a \neq 0$ and hence a is Noetherian iff $a = 0$. Moreover, distinguishing the cases $a = 0$ and $a \neq 0$, easy calculations show that in LAN we have $\mathsf{exh}\, a = \neg \ulcorner a$. This mirrors the fact that by totality of concatenation a nonempty language can be iterated indefinitely without reaching a terminal element. But we also have $a^\omega = 0$ whenever $1 \sqcap a = 0$. Therefore, unlike in the relational model, $a^\omega = 0 \;\not\Rightarrow\; \ulcorner(\mathsf{exh}\, a) = 1$, while still $\nu|a\rangle = 0 \Rightarrow \ulcorner(\mathsf{exh}\, a) = 1$. Hence, for termination analysis in KAs more general than the relational model, the element $\nu|a\rangle$ seems more adequate than a^ω.

9.4 Additivity of Termination

It has been shown that many statements of abstract rewriting that depend on termination assumptions can be proved in ω-algebra [53], among them an abstract variant of the Bachmair/Dershowitz well-founded union theorem [2], but also many of the so-called cooperation theorems. It seems that Kleene algebra and ω-algebra capture the regular fragment of abstract rewriting. However, many other properties of abstract rewriting require context-free reasoning. We will show in this and the following section that modal Kleene algebra provides ways of reasoning also in this larger fragment. Moreover, as we have seen in the previous section, there is a gap between termination in ω-algebra and in $^{\ulcorner}$-Kleene algebra. Here, we provide a proof of the Bachmair/Dershowitz theorem in $^{\ulcorner}$-Kleene algebra.

Consider a Kleene algebra K and $a, b \in K$. We say that *a semi-commutes* over b if $ba \leq a^+b^*$ and that *a quasi-commutes* over b if $ba \leq a(a+b)^*$. Semi-commutation and quasi-commutation state conditions for permuting certain steps to the left of others. In general, sequences with a-steps and b-steps can be split into a "good" part with all a-steps occurring to the left of b-steps and into a "bad" part where both kinds of steps are mixed. Semi-commutation implies quasi-commutation; if a is Noetherian then the reverse implications holds as well (see [53] for proofs).

One of the main results in this area is the Bachmair/Dershowitz well-founded union theorem; it generalizes in the following way from relations to modal Kleene algebra.

Theorem 9.5 *Let K be an extensional modal Kleene algebra. For all $a, b \in K$, let a quasi-commute over b. Then a and b are Noetherian iff their sum is Noetherian.*

The proof in modal Kleene algebra takes about one page of algebraic calculation, see [12]. This shows that modal Kleene algebra provides proofs for abstract rewriting that are as simple as those in omega algebra. Note that the proofs in [2] are rather informal, while also previous diagrammatic proofs (e.g. [21]) suppress many elementary steps. In contrast, the algebraic proofs are complete, formal and still simple. An extensive discussion of the relation between the proofs in omega algebra and their diagrammatic counterparts can be found in [17]. In particular, the algebraic proofs mirror precisely the diagrammatic ones. This also holds for the modal proofs we present here.

9.5 Newman's Lemma

We now turn from semi-commutation to commutation and confluence. For their direct algebraic characterisation one either has to use converse at the element level or a combination of forward and backward modalities at the operator level. Since we do not have converse available, we have to choose the second alternative.

We say that $b \in K$ *commutes* over $a \in K$ if $\langle b^*||a^*\rangle \leq |a^*\rangle\langle b^*|$, and *locally commutes* over a if $\langle b||a\rangle \leq |a^*\rangle\langle b^*|$. The more standard notions of confluence and local confluence are recovered by setting $a = b$. Newman's Lemma, originally stated for a single rewrite relation, says that a locally confluent and Noetherian rewrite relation is even confluent. It has been generalised to two relations in [50] for a theory of term-rewriting with non-symmetric relations that extends the traditional equational case. The generalisation of the equational Church-Rosser theorem is similar. While the Church-Rosser case has already been proved in Kleene algebra in [52], it has been argued in [53] that a proof of Newman's lemma does not work in pure Kleene or omega algebra, since these structures capture only the regular fragment of abstract rewriting while the standard proof of Newman's lemma requires context-free recursion in the centre of a formula with left and right contexts.

In contrast to previous approaches [16,47], modal Kleene algebra allows a calculational proof that mirrors precisely the previous diagrammatic one given in [50].

Theorem 9.6 *Let K be a modal Kleene algebra with complete test algebra. If $a + b$ is Noetherian and a and b locally commute then a and b commute.*

Proof. (Sketch) The central idea of our proof is to use a generalised predicate (rc stands for "restricted commutation")

$$rc(p, a, b) \Leftrightarrow \langle b^*|\langle p\rangle|a^*\rangle \leq |a^*\rangle\langle b^*|.$$

$rc(p, a, b)$ states that b commutes over a on all points characterized by the test p. We use the notation $\langle p\rangle$ to enhance the symmetry of the formulation; this is justified, since $|p\rangle = \langle p|$ for all tests p. Clearly, b commutes over a iff $rc(1, a, b)$, so that commutation can be retrieved as a special case. Then the predicate

$$r = \mathsf{sup}\,\{p \mid rc(p, a, b)\}$$

characterizes the set of all points on which b commutes over a; it is contracted by $|a + b]$, so that, by the second form (37) of Noethericity, we are done. $\square$

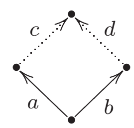
Again, the actual calculations take less than a page. For full details see [12].

Additionally, exhaustive iteration (48) and simplicity (34) can be used to show uniqueness of term normal forms for confluent actions. In the proof, the points in $(\mathsf{exh}\,a)^{\urcorner}$ represent term normal forms whereas uniqueness is expressed by simplicity. A proof can be found in [12].

10 Modal Kleene Algebra and Correspondence Theory

An important part of modal logic is correspondence theory [7,45]. It studies translations between relational and modal characterisations of certain properties of the underlying Kripke frames. Usually, the correctness proofs for these translations are done at the semantic level, frequently by pointwise arguments. In this section we give some examples of purely algebraic translations in the calculus of MKA.

We start with the commutation formulas used in the previous section and study diagrams of the type



In abstract relational algebra this is expressed as $a^{\smile}b \leq cd^{\smile}$. In an extensional MKA with converse, this can be translated into

$$a^{\smile}b \leq cd^{\smile} \;\Leftrightarrow\; \langle a||b\rangle \leq |c\rangle\langle d|. \tag{50}$$

But even in non-extensional MKAs the formula $\langle a||b\rangle \leq |c\rangle\langle d|$ is an adequate formulation; it expresses that any two transition paths along a and b that emanate from a common starting point can be joined by extending them by c and d transition paths, respectively.

In many modal logics, only forward or only backward modalities are available. So it is interesting which type of formulas can be expressed using only one sort of modality. For the above commutation property this is possible, resulting in the *Geach formula* [7,45]:

Lemma 10.1 *In general* MKA*s*

$$\langle a||b\rangle \leq |c\rangle\langle d| \;\Leftrightarrow\; |b\rangle|d] \leq |a]|c\rangle.$$

Hence, in extensional MKA*s with converse*

$$a^{\smile}b \leq cd^{\smile} \;\Leftrightarrow\; |b\rangle|d] \leq |a]|c\rangle.$$

Proof. Starting from the right-hand side of (50) this is shown very concisely at the operator level using the shunting rules (19) and (21):

$$\langle a||b\rangle \leq |c\rangle\langle d| \;\Leftrightarrow\; |b\rangle \leq |a]|c\rangle\langle d| \;\Leftrightarrow\; |b\rangle|d] \leq |a]|c\rangle.$$

□

Consequently, commutation and local commutation are equivalent to the following formulas:

$$|a^*\rangle|b^*] \leq |b^*]|a^*\rangle, \qquad |a\rangle|b^*] \leq |b]|a^*\rangle.$$

However, these are much less intuitive than our original ones. But the proof of Newman's Lemma can be carried out in this unidirectional form as well.

Special cases of commutation type properties are determinacy or simplicity $\langle a||a\rangle \leq \langle 1\rangle$ and totality or entireness$\langle 1\rangle \leq |a\rangle\langle a|$ (which is easily shown to be equivalent to $\ulcorner a = 1$) (see (34)).

Although the technique we have shown for translating modal validity is, of course, generally applicable, e.g., to Löb's formula, we refrain from treating further examples in this survey.

11 Greedy-Like Algorithms

11.1 Looping for Optimality

We conclude this survey by applying the theory to another algorithm derivation that ties in well with generalised confluence and exhaustive iteration.

A greedy algorithm solves an optimisation problems by proceeding in a stepwise fashion without backtracking. At each step it has a set of choices from which it always takes the one that seems best at the moment, i.e., it works locally without lookahead to the global optimum that is to be found eventually. Instances of this scheme are shortest path and minimum spanning tree problems in graphs, the construction of Huffman codes and scheduling problems. Of course, the greedy approach only works for certain types of problems: as is well-known from hiking in the mountains, always choosing the steepest path will rarely lead to the highest summit of the whole area. The central correctness requirement for the greedy scheme is that *a local choice must not impair reaching the global optimum*.

We now use modal Kleene algebra for deriving general conditions under which a loop satisfies this principle. It turns out that local optimality is inessential; so we study a more general class of loops that we call *greedy-like*. In [36] a relational derivation was abstracted to modal Kleene algebra via the Geach formula (cf. Lemma 101), whence avoiding backward modalities. While this corresponds to the standard approach that a modal logician would take, modal Kleene algebra offers the additional flexibility of simple combined reasoning with forward and backward modalities via Galois connections. Then the development of greediness conditions can be based again on commutation properties that, like in abstract rewriting, immediately reflect the choices that are taken at each step of a run of a greedy-like algorithm. Here, we briefly describe this commutation-based development.

We start with a specification element t that represents a relation between inputs and admissible outputs and an element c that represents a comparison relation on outputs capturing the notion of (global) optimality. The derivation will exhibit the precise requirements on c.

An element r *improves* t with respect to c if it always relates inputs to outputs that are at least as good as those prescribed by t. If r and t are relations this reads formally $t^{\smile} r \leq c$, which in MKA immediately translates into the predicate

$$imp(r,t,c) \stackrel{\text{def}}{\Leftrightarrow} \langle t||r\rangle \leq |c\rangle .$$

Since then 0 trivially improves t, we are interested in the greatest improvement. In REL this always exists and is given by the residual $t^{\smile} \backslash c$. However, since we want to avoid residuals, we will not make use of this representation.

An implementation of specification t that always produces optimal solutions then is an element r that refines and improves t. So we define

$$opt(r,t,c) \stackrel{\text{def}}{\Leftrightarrow} r \leq t \wedge imp(r,t,c)$$

and want to calculate a sufficient criterion under which a loop $w \stackrel{\text{def}}{=} \mathsf{while}\ p\ \mathsf{do}\ s$ with loop condition $p \in \mathsf{test}(K)$ and body $s \in K$ satisfies $opt(w,t,c)$, i.e.,

$$w \leq t, \qquad (51) \qquad\qquad imp(w,t,c), \qquad (52)$$

where we defer the treatment of (51) to the next section.

Spelling out the definitions in (52) results in $\langle t||(ps)^*\neg p\rangle \leq |c\rangle$. We abstract a bit and try to answer the question when, for $q \in \mathsf{test}(K)$ and $a \in K$, we have $\langle t||a^*q\rangle \leq c$. By the lifted semi-commutation property (26) in Section 7, this can be established if

$$\langle t||a\rangle \leq |c\rangle\langle t|, \qquad (53) \qquad\qquad \langle t|\langle q\rangle \leq |c\rangle, \qquad (54)$$

since then by locality

$$\langle t||a^*q\rangle = \langle t||a^*\rangle|q\rangle \leq |c^*\rangle\langle t|\langle q\rangle \leq |c^*\rangle|c\rangle \;=\; |c^+\rangle .$$

If we now assume c to be weakly transitive (40), which is reasonable for a comparison relation, we have $|c^+\rangle \leq |c\rangle$ and can draw the desired conclusion.

How can we, in turn, establish (53) and (54), at least in our special case? Translating back we get the proof obligations

$$\langle t||ps\rangle \leq |c\rangle\langle t|, \qquad (55) \qquad\qquad \langle t|\langle\neg p\rangle \leq |c\rangle . \qquad (56)$$

Condition (55) means that every pass through the loop body s preserves the possibility of obtaining a solution that is at least as good as all possible solutions before; (56) means that upon loop termination no possible solution is better than the termination value.

11.2 Iterating Through the Problem Domain

We now decompose the specification relation t into the exhaustive iteration of an element e of a set of elementary steps between points in the problem domain.

We admit arbitrary inputs as initial approximations but only terminal elements, from which no further elementary steps are possible, as outputs. Therefore we assume now that t has the special shape (48)

$$t = \mathsf{exh}\, e = e^* \,;\, \neg\ulcorner e = \mathsf{while}\ \ulcorner e\ \mathsf{do}\ e. \tag{57}$$

Such a problem structure is found, e.g., in matroids and greedoids [24,28] where it is additionally assumed that t is a discrete strict-order and that all terminal (or maximal) elements, the *bases*, have the same height (also known as *rank* or *dimension*) in the associated Hasse diagram.

We try to calculate an implementation that traverses the problem domain without backtracking. This suggests trying $ps \leq e$. Now, by isotonicity of the star operation, proof obligation (51) can be fulfilled if additionally we can achieve $\neg p \leq \neg\ulcorner e$ or, equivalently, $\ulcorner e \leq p$. Sufficient conditions for these properties are

$$ps \leq e \ \wedge\ \ulcorner(ps) \geq \ulcorner e. \tag{58}$$

These are reasonable requirements, since they prevent that the iteration blocks at a non-terminal element. They even imply $\ulcorner(ps) = \ulcorner e$.

Next, we tackle proof obligation (56), assuming (57). We calculate

$$\begin{aligned}
\langle t|\langle\neg\ulcorner e\rangle \leq |c\rangle &\Leftrightarrow \langle\neg\ulcorner e\rangle|t\rangle \leq \langle c| \\
&\Leftrightarrow \langle\neg\ulcorner e\rangle|e^*\rangle\langle\neg\ulcorner e\rangle \leq \langle c| \\
&\Leftrightarrow \langle\neg\ulcorner e\rangle(\langle 1\rangle + |e\rangle|e^*\rangle)\langle\neg\ulcorner e\rangle \leq \langle c| \\
&\Leftrightarrow \langle\neg\ulcorner e\rangle \leq |c\rangle.
\end{aligned}$$

Step one employs shunting (19,21) and de Morgan duality. Step two uses (57). Step three unfolds the star. Step four uses distributivity, locality, $\neg\ulcorner e\, e = 0$, idempotence of $\neg\ulcorner e$ and equality of backward and forward diamonds of a test.

So (56) is established if c is *weakly reflexive* on terminal elements, i.e., if $\langle\neg\ulcorner e\rangle \leq |c\rangle$. This holds, in particular, if c is fully reflexive, i.e., a pre-order. But in some applications one may choose to leave c partially reflexive. E.g., when constructing a Huffman code, the non-terminal elements are proper forests, for which a comparison relation is not given as easily as for the terminal elements, which are single code trees.

As for proof obligation (55), it is a generic condition that has to be considered individually in each case. Our derivation can be summed up as follows.

Theorem 11.1 *Suppose that c is weakly reflexive on $\neg\ulcorner e$ and weakly transitive, and that $t = \mathsf{exh}\, e$. Then*

$$ps \leq e \wedge \ulcorner(ps) \geq \ulcorner e \wedge \langle t||ps\rangle \leq |c\rangle\langle t| \ \Rightarrow\ opt(\mathsf{while}\ \ulcorner e\ \mathsf{do}\ s, t, c).$$

So far we still have a general scheme that does not specifically mention greediness. But we can further refine s to choose in every step a locally optimal element. To this end we need another pre-order l and stipulate $imp(s, e, l)$. This now provides a truly greedy algorithm, the correctness of which is already shown by Theorem 111. It corresponds to Curtis's "Best-Global" algorithm [10].

In [36] we provide a full reconstruction of Curtis's classification of Greedy algorithms [10] in the abstract setting of MKA, even using forward modalities only. The reason for this is that converse enters the derivation only in the limited way of general commutation properties which can be expressed by forward modalities only, using the Geach formula of Lemma 101. The modal approach again leads to considerably more concise proofs than the original relational/allegorical ones.

12 Conclusion

We have outlined the calculus of modal Kleene algebra and discussed several applications, most of them in the field of semantics, system calculi and development of programs and algorithms. The proofs that are needed in these examples are abstract, concise and entirely calculational.

Together with previous work [52,53], our case study in abstract rewriting, for instance, shows that large parts of this theory can easily be reconstructed in modal Kleene algebra. This is probably a novel idea. Other practical results, for instance the soundness proof of propositional Hoare logic or the reconstruction of temporal logics, are strongly based on previous work. Here, the main contribution is that modal Kleene algebra may serve as a convenient uniform framework. Sometimes, however, it even yields a drastic cut with Occam's razor: in the cases of propositional dynamic logic and linear temporal logic we can significantly reduce the number of axioms.

Relational algebraists may claim that most of the results presented in this paper could as well be treated in their formalism. While this is certainly true, since relations form a special instance of MKA, we believe that modal Kleene algebra still provides some advantages. It has fewer operations and it is algorithmically more tractable.

This often leads to a more concise and readable notation. Finally, the lifting to the modal operator algebras provides an additional level of abstraction that is not present in relational algebra.

There is one particular application of modal Kleene algebra that has not been discussed in this survey. Ehm has extended our approach to a calculus for the analysis of pointer algorithms [18]. He has combined modal Kleene algebra with techniques from fuzzy set theory to model the projection onto particular substructures of a given pointer structure. The reachability analysis performed by pointer algorithms, however, works to a large extent in pure modal Kleene algebra. Giving a full account of these results is beyond the scope of this paper.

So far, all our proofs are by paper and pencil. However, the simplicity of these proofs makes them ideal candidates for mechanisation. Our case studies in rewriting show that much less structure is needed for formalising proofs with a proof assistant than with previous approaches (e.g. [41,48]). We expect similar results when modal Kleene algebra is integrated into a formal method. Note that a considerable part of formal reasoning with popular methods like Z [49] or B [1] is essentially relational. In particular, Kleene algebra has strong connections to automata-theoretic decision procedures.

The results presented in this paper establish modal Kleene algebra as a formalism for safe cross-theory reasoning and therefore interoperability between different calculi for program and system analysis, modal or relational. We have tried to support this claim both from the syntactic and the semantic point of view. In the future, we plan extensive case studies, among others in the areas of program and protocol analysis. Due to its simplicity and flexibility, we believe that modal Kleene algebra offers a considerable potential that deserves further exploration.

Acknowledgment We are grateful to Roland Backhouse, Ernie Cohen, Sharon Curtis, Michael Ebert, Thorsten Ehm, Hitoshi Furusawa, Wolfram Kahl, Dexter Kozen, Hans Leiss, Oege de Moor, Gunther Schmidt, Michel Sintzoff and Joakim von Wright for valuable comments and discussions.

> Propose to an Englishman any principle, or any instrument, however admirable, and you will observe that the whole effort of the English mind is directed to find a difficulty, a defect or an impossibility in it. If you speak to him of a machine for peeling a potato, he will pronounce it impossible: if you peel a potato with it before his eyes, he will declare it useless, because it will not slice a pineapple.
>
> *Charles Babbage 1852*

References

1. J.-R. Abrial. *The B-Book.* Cambridge University Press, 1996.
2. L. Bachmair and N. Dershowitz. Commutation, transformation, and termination. In J. H. Siekmann, editor, *8th International Conference on Automated Deduction*, volume 230 of *LNCS*, pages 5–20. Springer, 1986.
3. R. Backhouse and D. Michaelis. Fixed-point characterisation of winning strategies in impartial games. In R. Berghammer, B. Möller, and G. Struth, editors, *Relational and Kleene-Algebraic Methods in Computer Science*, volume 3051 of *LNCS*, pages 34–47. Springer, 2004.
4. R. Berghammer and H. Zierer. Relational algebraic semantics of deterministic and nondeterministic programs. *Theoretical Computer Science*, 43:123–147, 1986.
5. C. Brink. Boolean modules. *Journal of Algebra*, 71:291–313, 1981.

6. M. Broy, R. Gnatz, and M. Wirsing. Semantics of nondeterministic and non-continuous constructs. In F.L. Bauer and M. Broy, editors, *Program Construction*, volume 69 of *LNCS*, pages 553–592. Springer, 1979.
7. B. F. Chellas. *Modal Logic: An Introduction*. Cambridge University Press, 1980.
8. E. Cohen. Separation and reduction. In R. Backhouse and J. N. Oliveira, editors, *Proc. of Mathematics of Program Construction, 5th International Conference, MPC 2000*, volume 1837 of *LNCS*, pages 45–59. Springer, 2000.
9. E. Cohen, D. Kozen, and F. Smith. The complexity of Kleene algebra with tests. Technical Report 96-1598, Computer Science Department, Cornell University, July 1996.
10. S.A. Curtis. The classification of greedy algorithms. *Science of Computer Programming*, 49:125–157, 2003.
11. J. Desharnais, B. Möller, and G. Struth. Kleene algebra with domain. *ACM Transaction on Computational Logic*, 2004. Preliminary version: Universität Augsburg, Institut für Informatik, Report No. 2003-07, June 2003.
12. J. Desharnais, B. Möller, and G. Struth. Termination in modal Kleene algebra. In J.-J. Lévy, E. Mayr, and J. Mitchell, editors, *Proc. IFIP TCS 2004*, pages 653–666. Kluwer, 2004.
13. J. Desharnais, B. Möller, and F. Tchier. Kleene under a demonic star. In T. Rus, editor, *Algebraic Methodology and Software Technology*, volume 1816 of *LNCS*, pages 355–370. Springer, 2000.
14. J. Desharnais, B. Möller, and F. Tchier. Kleene under a modal demonic star. *Journal on Logic and Algebraic Programming, Special Issue on Relation Algebra and Kleene Algebra*, 2004. (to appear).
15. H. Doornbos. A relational model of programs without the restriction to Egli-Milner-monotone constructs. In E.-R. Olderog, editor, *Programming Concepts, Methods and Calculi*, pages 363–382. North-Holland, 1994.
16. H. Doornbos, R. Backhouse, and J. van der Woude. A calculational approach to mathematical induction. *Theoretical Computer Science*, 179:103–135, 1997.
17. M. Ebert and G. Struth. Diagram chasing in relational system development. *ENTCS*, 2004. (to appear).
18. T. Ehm. Pointer Kleene algebra. In R. Berghammer, B. Möller, and G. Struth, editors, *Relational and Kleene-Algebraic Methods in Computer Science*, LNCS, pages 99–111. Springer, 2004.
19. T. Ehm, B. Möller, and G. Struth. Kleene modules. In R. Berghammer, B. Möller, and G. Struth, editors, *Relational and Kleene-Algebraic Methods in Computer Science*, volume 3051 of *LNCS*, pages 112–123. Springer, 2004.
20. P. Freyd and A. Scedrov. *Categories, allegories*. North-Holland, 1990.
21. A. Geser. *Relative termination*. PhD thesis, Fakultät für Mathematik und Informatik, Universität Passau, 1990.
22. R. Goldblatt. An algebraic study of well-foundedness. *Studia Logica*, 44(4):422–437, 1985.
23. D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.
24. P. Helman, B.M.E. Moret, and H.D. Shapiro. An exact characterization of greedy structures. *SIAM Journal on Discrete Mathematics*, 6:274–283, 1993.
25. M. Hollenberg. Equational axioms of test algebra. In M. Nielsen and W. Thomas, editors, *Computer Science Logic, 11th International Workshop, CSL '97*, volume 1414 of *LNCS*, pages 295–310. Springer, 1997.
26. N. Jacobson. *Basic Algebra*, volume I,II. Freeman, New York, 1985.
27. B. Jónsson and A. Tarski. Boolean algebras with operators, Part I. *American Journal of Mathematics*, 73:891–939, 1951.
28. B. Korte, L. Lovász, and R. Schrader. *Greedoids*. Springer, 1991.
29. D. Kozen. A representation theorem for $*$-free PDL. Technical Report RC7864, IBM, 1979.
30. D. Kozen. A completeness theorem for Kleene algebras and the algebra of regular events. *Information and Computation*, 110(2):366–390, 1994.
31. D. Kozen. Kleene algebra with tests. *Trans. Programming Languages and Systems*, 19(3):427–443, 1997.
32. Hans Leiß. Kleenean semimodules and linear languages. In Zoltán Ésik and Anna Ingólfsdóttir, editors, *FICS'02 Preliminary Proceedings*, number NS-02-2 in BRICS Notes Series, pages 51–53. Univ. of Aarhus, 2002.

33. Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems — Specification*. Springer, 1991.
34. B. Möller. Derivation of graph and pointer algorithms. In B. Möller, H.A. Partsch, and S.A. Schuman, editors, *Formal program development*, volume 755 of *LNCS*, pages 123–160. Springer, 1993.
35. B. Möller. Lazy Kleene algebra. In D. Kozen, editor, *Mathematics of Program Construction*, volume 3125 of *LNCS*, pages 252–273. Springer, 2004.
36. B. Möller and G. Struth. Greedy-like algorithms in modal Kleene algebra. In R. Berghammer, B. Möller, and G. Struth, editors, *Relational and Kleene-Algebraic Methods in Computer Science*, volume 3051 of *LNCS*, pages 202–214. Springer, 2004.
37. B. Möller and G. Struth. Modal Kleene algebra and partial correctness. In C. Rattray, S. Maharaj, and C. Shankland, editors, *Algebraic Methods and Software Technology*, volume 3116 of *LNCS*, pages 379–393. Springer, 2004.
38. B. Möller and G. Struth. wp is wlp. Technical Report 2004-14, Institut für Informatik, Universität Augsburg, October 2004. (to appear).
39. G. Nelson. A generalization of Dijkstra's calculus. *ACM Transactions on Programming Languages and Systems*, 11:517–561, 1989.
40. I. Németi. Dynamic algebras of programs. In *Proc. FCT'81 — Fundamentals of Computation Theory*, volume 117 of *LNCS*, pages 281–291. Springer, 1981.
41. T. Nipkow. More Church-Rosser proofs (in Isabelle/HOL). *J. Automated Reasoning*, 26(1):51–66, 2001.
42. Mathematics of Program Construction Group. Fixed point calculus. *Information Processing Letters*, 53:131–136, 1995.
43. B. Paige and S. Koenig. Finite differencing of computable expressions. *ACM Transactions on Programming Languages and Systems*, 4(3):402– 454, 1986.
44. D. Parnas. A generalized control structure and its formal definition. *Communications of the ACM*, 26:572–581, 1983.
45. S. Popkorn. *First Steps in Modal Logic*. Cambridge University Press, 1994.
46. V. Pratt. Dynamic algebras: Examples, constructions, applications. *Studia Logica*, 50:571–605, 1991.
47. G. Schmidt and T. Ströhlein. *Relations and Graphs*. EATCS Monographs in Computer Science. Springer, 1993.
48. N. Shankar. A mechanical proof of the Church-Rosser theorem. *Journal of the ACM*, 35(3):475–522, 1988.
49. J. M. Spivey. *Understanding Z*. Cambrigde University Press, 1988.
50. G. Struth. Non-symmetric rewriting. Technical Report MPI-I-96-2-004, Max-Planck-Institut für Informatik, 1996.
51. G. Struth. *Canonical Transformations in Algebra, Universal Algebra and Logic*. PhD thesis, Institut für Informatik, Universität des Saarlandes, 1998.
52. G. Struth. Calculating Church-Rosser proofs in Kleene algebra. In H.C.M. de Swart, editor, *Relational Methods in Computer Science, 6th International Conference*, volume 2561 of *LNCS*, pages 276–290. Springer, 2002.
53. G. Struth. Abstract abstract rewriting. *Journal on Logic and Algebraic Programming, Special Issue on Relation Algebra and Kleene Algebra*, 2004. (to appear).
54. V. Trnkova and J. Reiterman. Dynamic algebras with tests. *Journal of Computer and Systems Science*, 35:229–242, 1987.
55. B. von Karger. Temporal algebra. *Mathematical Structures in Computer Science*, 8:277–320, 1998.
56. J. von Wright. From Kleene algebra to refinement algebra. In E. Boiten and B. Möller, editors, *Mathematics of Program Construction*, volume 2386 of *LNCS*, pages 233–262. Springer, 2002.

Journal on Relational Methods in Computer Science, Vol. 1, 2004, pp. 93 - 131
Received by the editors March 14, 2004, and, in revised form, October 12, 2004.

Published on December 10, 2004.

This article may be accessed via WWW at http://www.jormics.org.