Left omega algebras and regular equations

Georg Struth

Department of Computer Science, The University of Sheffield, UK

ARTICLE INFO

Article history:
Available online 6 June 2012

Keywords:
Kleene and omega algebra
Arden's rule
Regular equations
Automated and interactive theorem proving
Formalised mathematics

ABSTRACT

Left omega algebras, where one of the usual star induction axioms is absent, are studied in the context of recursive regular equations. Abstract conditions for explicitly defining the omega operation are presented. They are used for developing abstract side conditions on Arden's rule that are necessary for solving such equations. The definability and solvability results are refined to concrete models, to languages, traces and relations. It turns out, for instance, that the omega operation captures precisely the empty word property in regular languages and wellfoundedness in relational models. The approach also leads to simple new relative completeness results for left omega algebras, and for Salomaa's axioms for regular expressions. Since automated theorem proving and counterexample search within the theorem proving environment Isabelle/HOL are instrumental for this investigation, it is also an exercise in formalised mathematics.

© 2012 Elsevier Inc. All rights reserved.

1. Introduction

In the context of termination and nontermination analysis in regular algebras [8,14], I revisit Arden and Salomaa’s approaches to solving regular equations from the point of view of left omega algebras. Omega algebras were introduced as algebras of omega-regular events [6] to capture the equational theories of omega-regular expressions and languages. They expand the regular operations of union, concatenation and finite iteration, as axiomatised by regular algebras [7], by an operation for infinite iteration. Left omega algebras are based on left Kleene algebras, which are variants of Kozen’s Kleene algebras [16] in which the right star induction axiom is absent. Left Kleene algebras are sound and complete for the equational theory of regular expressions [4].

The two main topics of this work are conditions for explicitly defining the omega operation and for unique solvability of recursive regular equations. The topics are related and explored both abstractly and in concrete models. Another main contribution is the formalisation of all structures and conditions considered in the interactive theorem prover Isabelle/HOL [17], using its automated theorem proving and counterexample search facilities.

The omega operation is explicitly definable in models that are inherently finite, for instance regular languages, sets of finite paths or traces, or binary relations. This is not too surprising since, intuitively, omega is forced to be finite in these models. It is certainly less obvious that it carries an interesting meaning in this setting.

Arden's rule is the classical tool for solving systems of recursive language equations [1]. A side condition is the negated empty word property which holds if a language does not contain the empty word. It is particular to the language model. In Salomaa’s and Wagner’s axioms for regular and omega regular algebras [18,19], for instance, it is defined inductively on the term algebra. This is not entirely satisfactory algebraically, and it excludes interesting models of computation beyond regular languages. To solve recursive equations over arbitrary regular algebras, more abstract side conditions are needed.

The main technical results of this paper are as follows:

First, abstract sufficient conditions are presented for explicitly defining the omega operation in left Kleene algebras; thus for building left omega algebras as their conservative extensions.
Second, Arden’s rule is derived in left Kleene and omega algebras from four different algebraic side conditions. For left omega algebra with domain, wellfoundedness is established as a fifth one. Therefore, under these side conditions, certain linear recursive equations in the language of Kleene algebras have unique solutions. Moreover, it is shown that these side conditions have meaningful interpretations in various models.

Third, the abstract results are refined to three main models of regular algebra: languages, traces and relations. However, instead of fully formalising these models, we impose simple abstract algebraic conditions that sufficiently characterise each model. In the language model, the omega operation is Boolean-valued, hence a predicate. It characterises precisely the empty word property. Arden’s rule for regular languages can then be obtained from its abstract relative via a simple length-increase argument. Similar results are obtained for trace and path algebras. In the relational model, all side conditions studied become equivalent to wellfoundedness. Arden’s rule now specialises to a unique extension property [10], which has been derived, for the first time, in a first-order setting.

Fourth, the results obtained are applied in two simple relative completeness results. Deriving the axioms of left Kleene algebra from Salomaa’s axioms provides a new and simple completeness proof of the latter relative to Boffa’s result. Deriving Wagner’s axioms from left omega algebras shows that the latter are complete with respect to omega regular languages.

As so often with variants of Kleene algebras, the strength of the approach shows through the ratio of theorems per axiom, the simplicity of concepts and proofs, and the range of previously fragmented results that can be captured uniformly. Here, various model-specific solvability results are obtained by minor additions to a core algebra; quickly and simply by automated reasoning.

2. Automated theorem proving in Isabelle

Isabelle/HOL [17] is one of the most popular theorem proving environments in which mathematical theories can be implemented and mathematical proofs can be mechanised. Traditionally, Isabelle has used complex rewriting procedures and built-in proof engines for verifying proof steps given by a user. Proofs accepted by the tool are highly trustworthy since Isabelle’s logical core is small and simple. All additional procedures used have been verified relative to this core.

Isabelle has recently been revolutionised by integrating external automated theorem proving systems and counterexample generators in a trustworthy way (see [3] for an overview). Isabelle uses the Sledgehammer command to call the external theorem provers. It uses a relevance filter to gather hypotheses that might be useful for proving a certain goal. Based on the actual hypotheses used by the external tools it then reconstructs their proofs by replaying proof search with an internally verified theorem prover. Proof reconstruction is essential because automated theorem provers are orders or magnitudes more complex—hence less trustworthy—than Isabelle. Sledgehammer has been complemented by tools that search for finite counterexamples, notably Nitpick and Quickcheck. This new technology makes theorem proving in Isabelle much more simple and natural for mathematicians. In the context of algebra, and in this paper, Isabelle is often merely a theory manager for the automated theorem provers.

Beyond that, Isabelle offers additional advantages for algebraic reasoning. Isabelle’s locales and axiomatic type classes support the design of theory hierarchies. Omega algebras, for instance, can be implemented as expansions of Kleene algebras, models of Kleene algebras, for instance, can be formally linked with the axiomatic layer by instantiation. Theorems are automatically inherited across hierarchies, that is, from superclasses to subclasses, and from the axiomatic level to instances. Isabelle’s higher-order features support the formalisation of symmetries and dualities and the automatic generation of theorems by duality. Finally, Isabelle allows pretty printing of theories and its proof scripting language Isar yields human readable proofs.

The formalisation in this paper is relative to a large repository for Tarski–Kleene algebras in Isabelle [1] which covers most variants of regular and omega regular algebras and their most important models [11–13]. All calculational results in this paper could be obtained quickly and easily from this repository, and complemented by counterexamples. In fact, the entire paper itself is an executable Isabelle file which is formally verified during typesetting. This file is available at the repository web site for inspection and verification. To enhance this process this document uses a simplified image of the omega algebra theory hierarchy and only those theorems that are needed for this paper.

3. Left omega algebras

The algebras studied in this paper are based on idempotent semirings or dioids. Dioids expanded by an operation of finite iteration are known as Kleene algebras. Omega algebras are obtained by further expanding these by an operation of infinite iteration.

Formally, a dioid is a structure \((S, +, \cdot, 0, 1)\) over a set \(S\) such that \((S, +, 0)\) is a join semilattice with zero, \((S, \cdot, 0)\) is a monoid, multiplication distributes over addition from the left and right, and zero is a left and right annihilator with respect to multiplication. Every dioid is ordered by the usual order \(\leq\) on the join semilattice reduct \((S, +)\). The operations of addition and multiplication are isitone with respect to that order and 0 is its least element.

---

1. [http://www.dcs.shef.ac.uk/~georg/isa](http://www.dcs.shef.ac.uk/~georg/isa)
In Isabelle, dioids can be defined as axiomatic classes in a compositional way from join semilattices.

Class join-semilattice-zero = plus-ord + zero +
assumes add-assoc: \((x+y)+z = x+(y+z)\)
and add-comm: \(x+y = y+x\)
and add-idem: \(x+x = x\)
and add-zerol: \(0+x = x\)

Join semilattices themselves are subclasses of Isabelle’s built-in theory of order. The class plus-ord adds an operation of addition and postulates that \(x \leq y\) if and only if \(x+y = y\). zero is a class that simply provides the operation 0. It can be found in a Signature file that is loaded by the Isabelle theory file underlying this paper. Defining constants in this way avoids name clashes.

I have formally proved that all join semilattices are orders to make all Isabelle theorems about orders available in our context. By defining dioids as expansions of join semilattices, theorems are again inherited.

Class dioid = join-semilattice-zero + mult-op + one +
assumes mult-assoc: \((x\cdot y)\cdot z = x\cdot (y\cdot z)\)
and distr: \((x+y)\cdot z = xz+yz\)
and distl: \((x+y)\cdot z = xy+xz\)
and mult-onel: \(1\cdot x = x\)
and mult-oner: \(x\cdot 1 = x\)
and annil: \(0\cdot x = 0\)
and annm: \(x\cdot 0 = 0\)

Semirings and dioids satisfy a duality principle. The opposite of a semiring or dioid can be formed by swapping the order of multiplication. Since all axioms of semirings and dioids are transformed into axioms under opposition, the opposite of a semiring or dioid is again a semiring or dioid and theorems are preserved under opposition as well.

A left Kleene algebra is a dioid \(K\) expanded by a star operation \(^* : K \rightarrow K\) which satisfies an unfold axiom and an induction axiom.

Class left-kleene-algebra = dioid + star-op +
assumes star-unfoldl: \(1+x\cdot x^* \leq x^*\)
and star-inductl: \(z+x\cdot y \Rightarrow x^*\cdot z \leq y\)

The opposite of a left Kleene algebra is a right Kleene algebra. A right Kleene algebra satisfies the dual axioms \(1+x^*\cdot x = x^*\) and \(z + y \cdot x \leq y \Rightarrow z \cdot x^* \leq y\). A Kleene algebra is both a left and a right Kleene algebra. An example of a left Kleene algebra in which the right induction axiom fails has been given by Kozen [15].

A left omega algebra is a left Kleene algebra \(K\) expanded by an omega operation \(\omega : K \rightarrow K\) which satisfies an unfold axiom and a coinduction axiom.

Class left-omega-algebra = left-kleene-algebra + omega-op +
assumes omega-unfold: \(x^\omega \leq x\cdot x^\omega\)
and omega-coinduct: \(y \leq z+x\cdot y \Rightarrow y \leq x^\omega+x^\omega\cdot z\)

An omega algebra is a left omega algebra that is also a Kleene algebra.

Every omega algebra has a maximal element with respect to the natural order, namely \(1^\omega\), for which we henceforth write \(\top\), whereas Kleene algebras need not possess maximal elements. Such Kleene algebras cannot be expanded to omega algebras [14]. An important property is that all operations of (left) omega algebras are isotone with respect to the natural order. A peculiarity is that \(x^\omega = \top\) for all \(x \geq 1\).

4. Languages, traces, relations

This paper focusses on the regular language model, the regular trace model and the regular relational model of left Kleene algebras and left omega algebras, which of course are also models of Kleene algebras and omega algebras. An implementation of these models can be found in the repository. “Regular” means that only the regular operations—addition, multiplication, star—are used and only finite words and traces are considered. These models have been studied extensively in the literature.

Example. (Regular) languages form Kleene algebras. Let \(\Sigma\) be an alphabet and let \(\Sigma^*\) be the free monoid over \(\Sigma\). A language is a subset \(X\) of \(\Sigma^*\). The structure \((2^{\Sigma^*}, \cup, \cdot, \epsilon, \emptyset, \{\epsilon\})\) forms a Kleene algebra under the standard regular operations of formal language theory: addition is \(X \cup Y\), multiplication is the complex product \(XY = \{ xy \in \Sigma^* : x \in X \text{ and } y \in Y \}\) and the star is defined as \(X^* = \cup_{i\geq0} X^i\), where the powers of \(X\) are inductively defined. This Kleene algebra is the full language Kleene algebra over \(\Sigma\). Each subalgebra of a full language Kleene algebra is a language Kleene algebra. A language is regular if it can be inductively generated from the empty language, the empty string language \(\{\epsilon\}\) and the singleton languages \(\{a\}\) for each \(a \in \Sigma\) by applying the regular operations.

It is well known that regular languages can be represented by regular expressions, and I usually do not distinguish them. Regular expressions and Kleene algebras have the same signature. A classical result in Kleene algebra shows that the above
axioms for Kleene algebras are not only sound, but also complete for the equational theory of regular languages or regular expressions [16]. Hence regular expressions form the term algebras of Kleene algebras. Identities between Kleene algebra terms are valid if and only if the corresponding regular expressions are equivalent, where regular expression equivalence is induced by language identity. The equational theory of Kleene algebras can therefore be decided by finite automata. Boffa has shown soundness and completeness for the superclass of left Kleene algebras [4].

**Example.** Binary relations form Kleene algebras. Let $A$ be a set. The structure $(2^{A \times A}, \cup, \circ, *, \emptyset, 1_A)$ forms a Kleene algebra, where $2^{A \times A}$ denotes the set of all binary relations on $A$, $\cup$ is set union, $\circ$ is relative product, $*$ is the reflexive-transitive closure operation, $\emptyset$ is the empty relation and $1_A$ the identity relation on $A$. This Kleene algebra is called the full relation Kleene algebra over $A$. Each subalgebra of a full relation Kleene algebra is a Kleene algebra called a relation Kleene algebra over $A$.

It is known that language Kleene algebras and relation Kleene algebras have the same equational theories [16].

**Example.** Sets of traces form Kleene algebras. A trace over a (finite) set $P$ and a (finite) set $A$ is a finite sequence over $(P \cup A)^*$, in which the first and last letter are in $P$ and in which letters from $P$ and $A$ alternate. $(P, A)^*$ denotes the set of all traces over $P$ and $A$. The product of traces is a partial operation:

$$p_0 a_0 \ldots a_{m-1} p_m \cdot q_0 b_0 \ldots b_{n-1} q_n = p_0 a_0 \ldots a_{m-1} p_m b_0 \ldots b_{n-1} q_n$$

if $p_m = q_0$, and it is undefined otherwise. The product $T_1 \cdot T_2$ on sets of traces, which is total, and the remaining regular operations can be defined as in the language case. This turns $2^{(P, A)^*}$ into a Kleene algebra, the full trace Kleene algebra over $P$ and $A$. In particular, $P$ is the multiplicative unit of this algebra. Again, every subalgebra of a full trace Kleene algebra is a Kleene algebra called a trace Kleene algebra.

Path algebras can be obtained from trace algebras by “forgetting” the elements of $A$. They are very similar to trace algebras and therefore not discussed any further in this paper.

### 5. Defining omega

This section provides conditions for explicitly defining the omega operation in left omega algebras. These extend, simplify and generalise previous work on trace semirings [14].

An element $x$ of a dioid is dense if $x \leq xx$ holds.

**Definition (in dioid)**

```plaintext
dense :: 'a ⇒ bool
where dense x ⟷ x ≤ x • x
```

In particular, every multiplicatively idempotent element is dense, and every element above 1 is dense.

**Lemma** idem-dense: $x • x = x$ ⇒ dense $x$

by (metis dense-def le-less)

**Lemma** supid-dense: $1 \leq x$ ⇒ dense $x$

by (metis dense-def mult-isor mult-onel)

In the first lemma, the name idem-dense has been assigned to make this fact available for further proofs. The line after the lemma contains its proof. It states that the internal automatic theorem prover Metis can verify this lemma by using the previously proved facts dense-def—the definition of density—and le-less—a property of partial orders.

In the Isabelle proof cycle, the command sledgehammer has been typed after the lemma statement to call external theorem provers. The line after the lemma has been generated by the tool in response to the external provers. More information can be found in the Isabelle documentation.

Subidentities in dioids, and even in left omega algebras, however, need not be dense—Nitpick, Isabelle’s counterexample generator, found a counterexample with three elements—but they are dense in many models of interest, for instance the trace model. Also, it is not the case that every dense element is idempotent (two-element counterexample) or above 1 (four-element counterexample). All counterexamples can be found in the Isabelle theory file for this paper.

For dense elements, omega can be defined explicitly.

**Lemma** dense-top: dense $x$ ⇒ $x^\omega = x \cdot \top$

by (metis dense-def eq-iff max-element mult-isol mult-oner omega-simulation omega-unfold-eq top-def)

The omega of a dense element in a sum vanishes as well.
The converses of these two implications do not hold, as Nitpick shows.

The last proof is an example of a step-wise proof that uses Isabelle’s proof scripting language Isar. This lemma yields an explicit definition of omega if \( y^\omega = 0 \), that is, \( y \) is \( \omega \)-trivial.

Theorem split-lemma: dense \( x \land y^\omega = 0 \rightarrow (x+y)^\omega = y^\omega \cdot x \cdot T \)

by (metis add-zero dense-sum-top)

Corollary split-lemma-var: \( \forall x \cdot y \cdot z \cdot (x = y + z \land \text{dense } y \land z^\omega = 0 \rightarrow x^\omega = z^\omega \cdot y \cdot T) \)

by (metis split-lemma)

Theorem split-lemma and its corollary yield abstract sufficient conditions for explicitly defining omega. Nitpick could show that these conditions are not necessary.

Of course, \( 0 \) is both dense and \( \omega \)-trivial, hence in the above definition both \( y \) and \( z \) can be zero.

The following sections show that this splitting works in many important models, in particular language, trace and path models, but not easily in the relational model.

In left omega algebras, \( \omega \)-triviality can be related to equivalent conditions. First, I call an element \( x \) of a dioid deflationary if the following condition holds.

Definition (in dioid)

\[ \text{defl} :: \: a \rightarrow \text{bool} \]

\[ \text{where defl } x \leftrightarrow (\forall y . (y \leq x \cdot y \rightarrow y = 0)) \]

This is motivated by the fact that, in Bourbaki–Witt fixpoint theory, functions on posets that satisfy \( y \leq f(y) \) for all \( y \) are often called inflationary.\(^2\)

The following fact rules out that dense elements and elements greater than one are deflationary.

Lemma dense-defl: \( x \neq 0 \rightarrow \text{dense } x \rightarrow \neg \text{defl } x \)

by (metis defl-def dense-def)

Lemma supid-inflationary: \( 1 \leq x \rightarrow (\forall y . y \leq x \cdot y) \)

by (metis mult-isor mult-onel)

Lemma defl-super-id: \( 0 \neq l \rightarrow (l \leq x \rightarrow \neg \text{defl } x) \)

by (metis defl-def mult-onel)

The converses of these implications need not hold: elements that are not deflationary need neither be dense nor greater than 1. Nitpick also shows that the preconditions \( 0 \neq 1 \) in some of these lemmas are essential.

The following statement shows that an element of a left omega algebra is deflationary if and only if it is \( \omega \)-trivial.

This fact is already known for omega algebras [14].

Theorem defl-trivial: \( \text{defl } x \leftrightarrow x^\omega = 0 \)

by (metis add-comm add-zero defl-def eq-iff omega-coinduct-var omega-unfold-eq)

Next I give two alternative conditions. The first one is called strong deflationarity, the second one \( \omega \)-boundedness. In left omega algebras, they are equivalent to deflationarity and omega triviality. Hence all these conditions are equivalent.

Definition (in left-kleene-algebra)

\[ \text{s-defl} :: \: a \rightarrow \text{bool} \]

\[ \text{where s-defl } x \leftrightarrow (\forall y \cdot z . y \leq x \cdot y + z \rightarrow y \leq x^\omega \cdot z) \]

Definition (in left-omega-algebra)

\[ \text{om-bound} :: \: a \rightarrow \text{bool} \]

\[ \text{where om-bound } x \leftrightarrow (\forall z . x^\omega \leq x^\omega \cdot z) \]

Theorem alt-trivial: \( x^\omega = 0 \leftrightarrow \text{om-bound } x \)
by (metis add-comm add-zerol annihil leq-def om-bound-def)

Theorem defl-s-defl: \( x^\omega = 0 \leftrightarrow s\text{-defl } x \)
by (metis add-ub add-zerol annihil eq-iff omega-unfold-eq add-comm omega-coinduct s-defl-def)

It is interesting to consider the two variants of deflationarity already in the weaker setting of left Kleene algebras.

Theorem s-defl-to-defl: \( s\text{-defl } x \rightarrow \text{defl } x \)
by (metis add-comm add-zerol annihil defl-def s-defl-def leq-def)

For the converse direction, however, I neither obtain a proof nor a refutation within the running time bounds of Sledgehammer or Nitpick.

To sum up, I have obtained an abstract splitting condition under which the omega operation can be explicitly defined in left omega algebras. In that case, left Kleene algebras can be extended by definition. Then every statement of the restricted theory holds in its extension, and every statement in the language of the restricted theory which holds in the extended theory holds already in the restriction. In other word, extensions by definition do not add expressive power.

This condition is that an element is dense and omega trivial. Omega triviality has been shown to be equivalent in left omega algebras to three other conditions: deflationarity, strong deflationarity and omega boundedness. The two conditions have been separated: Nontrivial dense elements are not deflationary, but the converse need not always hold. Superidentities, that is, elements above 1 and multiplicative idempotents are always dense, but dense elements need neither be superidentities nor idempotents.

6. Arden’s rule abstractly

Arden’s rule is a fundamental tool of language theory [1]. To determine, for instance, the language accepted by the automaton

\[
\begin{array}{c}
0 \rightarrow a \rightarrow 1 \\
\end{array}
\]

it can be translated into a system of recursive language equations

\[
x_0 = (a + b)x_0 + ax_1 \quad x_1 = ax_2 \quad x_2 = 1
\]

Arden’s rule yields a way of solving this system. The solution to the first equation—which is recursive—is \( x_0 = (a + b)^*ax_1 \). Solutions for \( x_1 \) and then \( x_2 \)—which are not recursive—are obtained by substitution. This yields the solution \( x_0 = (a + b)^*aa1 = (a + b)^*aa \), which is, of course, the regular expression corresponding to the automaton.

More generally, Arden’s rule states that, whenever a language denoted by a regular expression \( x \) does not contain the empty word, if \( y = x \cdot y + z \) is valid, then \( y = x^* \cdot z \) is valid. In other words, if a language \( x \) does not have the empty word property, then the recursive equation \( y = x \cdot y + z \) has the unique solution \( y = x^* \cdot z \).

It should be evident that Arden’s rule is of general interest for modelling and reasoning about computing systems in terms of systems of recursive equations. The work of Salomaa [18] shows that Arden’s rule is the basis of a simple algebraic proof of one direction of Kleene’s theorem. Also Backhouse and Carré’s study of matrix algebras over regular algebras [2] relies heavily on it.

To provide a more general context for Arden’s rule I prove it abstractly in left omega algebras and discuss some of its consequences and variants at the algebraic level.

Theorem arden: \( x^\omega = 0 \rightarrow z + x \cdot y = y \rightarrow x^* \cdot z = y \)
by (metis add-zerol eq-iff antisym-conv omega-coinduct star-induct)

The precise relationship between \( \omega \)-triviality of \( x \) and the empty word property in the language model is explained in the following section.

It is evident that Arden’s rule also holds under the three equivalent conditions of the previous section. A two-element counterexample provided by Nitpick refutes the law \( z + x \cdot y = y \rightarrow x^* \cdot z = y \) without additional side conditions.

Isabelle easily obtains the following variants.

Lemma arden-equiv: \( x^\omega = 0 \rightarrow (z + x \cdot y = y \leftrightarrow x^* \cdot z = y) \)
by (metis arden distr mult-assoc mult-oneI star-unfold-eq)

Lemma arden-equiv-var: \( x^\omega = 0 \leftrightarrow (\forall y z. z + x \cdot y = y \rightarrow x^* \cdot z = y) \)
by (metis add-zerol annihil arden omega-unfold-eq)
Nitpick refutes the possibility of replacing the implication in the second lemma by an equivalence. Again these variants hold with \( \omega \)-triviality replaced by the three equivalent conditions.

Nitpick also refutes the conjecture that Arden’s rule might hold for elements that are not dense or not above 1 instead of \( \omega \)-trivial.

All these statements show that in left omega algebras, the equation \( y = x \cdot y + z \) has the unique solution \( y = x^* \cdot z \) if and only if \( x \) is \( \omega \)-trivial or satisfies an equivalent condition.

It is interesting to contrast Arden’s rule with the star induction rule which, in left Kleene algebras, is equivalent to the following rule.

**Lemma** \( z + x \cdot y = y \rightarrow x^* \cdot z \leq y \)

**by** (metis order-refl star-inductl)

Accordingly, \( x^*z \) is the least solution of the equation \( y = xy + z \) in every left Kleene algebra. But, as the next lemma shows, it need not be the only solution.

**Lemma** arden-sol: \( 1 \leq x \land z \leq w \rightarrow x \cdot x^* \cdot w + z = x^* \cdot w \)

**by** (metis add-comm leq-def order-trans star-ref star-unfoldl-eq supid-inflationary)

This shows that in every left Kleene algebra, if \( 1 \leq x \), then \( y = x \cdot y + z \) has solutions \( y = x^* \cdot w \) for all elements \( w \geq z \).

Nitpick also shows that in some left Kleene algebras, for some elements \( x \) and \( z \), the equation \( y = x \cdot y + z \) has more than one solution.

An interesting question is whether deflationarity or strong deflationarity, instead of \( \omega \)-triviality, implies Arden’s rule already in (left) Kleene algebras, that is, for a signature that contains only those operations that occur in the rule itself.

In fact, the following result holds in left Kleene algebras.

**Theorem** arden: \( s \cdot x \rightarrow (y = x \cdot y + z \rightarrow y = x^* \cdot z) \)

**by** (metis add-comm eq-iff star-inductl s-defl-def)

By contrast, in the setting of left Kleene algebra I could neither prove nor refute Arden’s rule using the condition of deflationarity within Isabelle’s time bounds. This is consistent with the facts from the previous section: Lemma \( s \cdot x \rightarrow \text{defl-def} \) shows that strong deflationarity implies deflationarity in the context of left Kleene algebras, whereas the converse implication could neither be proved nor refuted.

The next sections refine these results to the most important models of Kleene algebras, to languages, traces and relations.

7. Omega and regular languages

The results of the previous sections immediately specialise to language omega algebras. In this setting, the superidentities represent those languages that have the empty word property. Therefore, by Lemma supid-dense, languages that have the empty word property are dense and their omega is equal to \( \top \).

To capture the additional properties of the language model as far as needed in this paper, I add one single algebraic condition to left omega algebras.

**Class** lang-left-omega-algebra = left-omega-algebra +

**assumes** defl-eq-newp: \( \neg 1 \leq x \rightarrow \text{defl} x \)

This fact, namely that nonempty languages that do not satisfy the empty word property are deflationary, can easily be justified by the following length-increase argument: Suppose that a language \( x \neq 0 \) does not have the empty word property and let \( y \neq 0 \) be another language. Then any word of minimal length in \( x \cdot y \) must be strictly greater than any word of minimal length in \( y \) and therefore \( y \neq x \cdot y \), that is, \( x \) is deflationary.

I could have formally verified this simple argument within Isabelle by using the language model of left Kleene algebra from the repository. But, in contrast to the calculational results in this paper, the efforts of this formalisation is out of proportion with the simplicity of the result.

In language left omega algebras, deflationarity is now equivalent to the absence of the empty word property.

**Lemma** defl-ewp: \( 0 \neq 1 \rightarrow (\neg 1 \leq x \leftrightarrow \text{defl} x) \)

**by** (metis defl-eq-newp defl-super-id 1 defl-trivial)

This fact has already been used by Backhouse and Carré [2]. Their Theorem 5.2 proves the following fact (with respect to deflationary elements) for the special case of matrices over regular languages, from which the statement for regular languages follows as a subcase.

The conditions for explicitly defining the omega operation in Theorem split-lemma now refine to a simple case analysis, and the omega operation reduces to yet another way of measuring the empty word property.

**Theorem** lang-omega-def: \( x^\omega = (\text{if } 1 \leq x \text{ then } \top \text{ else } 0) \)

**by** (metis antisym defl-eq-newp defl-trivial max-element omega-iso top-def)

Thus every language (left) Kleene algebra can be uniquely expanded into a language (left) omega algebra; language omega algebras are extensions by definition of language Kleene algebras.
It is important that “extension” means that the omega operation is defined on regular languages, not that regular languages are expanded to omega-regular languages. Unfortunately, the uses of “extension” in model theory and universal algebra, where it means adding elements to models, and proof theory, where conservative extensions expand signatures, may be confusing. As a consequence of Theorem \text{lang-omega-def}, the absence of the empty word property can be defined as an identity in left language omega algebra, whereas this is not possible in Kleene algebra, where deflationarity, which is a quasi-identity, hence a universally quantified equational Horn formula, could be used.

**Corollary 1.** Language (left) omega algebras are conservative extensions of language (left) Kleene algebras.

Therefore, if a theorem in the language of Kleene algebras holds in all language omega algebras, it already holds in all language Kleene algebras.

Finally, Arden’s rule of formal language theory holds with the empty word property as a side condition.

**Theorem** \text{arden}: \( \neg \, l \leq x \rightarrow (z + x y = y \leftrightarrow x^* z = y) \)

**by** (metis \text{arden-equiv} defl-eq-newp defl-trivial)

Formally, of course, it can also be written as a quasi-identity, which is not possible in Kleene algebra. All the other abstract results from Section 5 and Section 6 hold in the language model, too.

### 8. Omega and traces

Trace omega algebras have, to some extent, been studied in [14]. The arguments are similar to, but slightly different from language omega algebras.

In trace models, the elements between 0 and 1 are the subsets of \( P \), that is, sets of traces of length one. They form a Boolean subalgebra and are therefore multiplicatively idempotent, hence dense. A set of traces is called test-free if it does not contain a subset of \( P \). Each set of traces can be split into a subset of \( P \) and a test-free subset (both possibly empty).

This is captured algebraically by defining two functions \( tp \) and \( tfp \) that project on the test part and the test-free part of an element \( x \) of a left omega algebra. The two functions are therefore assumed to be idempotent.

**Class** \text{trace-left-omega-algebra} = \text{left-omega-algebra} +

**fixes**

\( tp :: a \Rightarrow a \)

and \( tfp :: a \Rightarrow a \)

**assumes**

\( tp\text{-retract} : tp \, (tp \, x) = tp \, x \)

\( tfp\text{-retract} : tfp \, (tfp \, x) = tfp \, x \)

\( tp\text{-subid} : tp \, x \leq l \)

and \( tfp\text{-not-supid} : \neg \, l \leq tfp \, x \)

\( tp\text{-tfp} : x = tp \, x + tfp \, x \)

**and**

\( subid\text{-dense} : x \leq l \rightarrow (x \leq x \, x) \)

\( defl\text{-eq-tf} : tfp \, x = x \rightarrow (defl \, x) \)

It is also required that the test part of an element \( x \) is a subidentity and that the test-free part is not a superidentity. The test part joined with the test-free part of \( x \) must be equal to \( x \). In the concrete trace model, the subidentities form a Boolean subalgebra, but here it suffices to require that all subidentities are dense. Finally, all test-free elements are required to be deflationary (obviously, an element \( x \) is test-free if and only if it is equal to its test-free part). This last condition can again be verified by a length-increase argument similar to the language case [14].

It follows that test parts are multiplicatively idempotent, hence dense, and test-free parts are \( \omega \)-trivial.

**Lemma** \text{tpidem} : \( (tp \, x) = (tp \, x) \cdot (tp \, x) \)

**by** (metis \text{subid-dense} tp-subid mult-isor eq-iff)

**Lemma** \text{tfp-om} : \( (tp\, x)^\omega = 0 \)

**by** (metis defl-eq-tf defl-trivial tfp-retract)

Test-freeness and deflationarity (and also the other equivalent conditions) are again equivalent.

**Lemma** \text{defl-tfp} : defl \, x \leftrightarrow tfp \, x = x

**by** (metis add-ub defl-def defl-eq-tf leq-def min-zero mult-isor tp-tfp tpidem)

Moreover, the omega operation can again be explicitly defined.

**Theorem** \text{trace-om} : \( x^\omega = (tp\, x)^* \cdot (tp\, x)^- \cdot T \)

**by** (metis dense-def order-refl split-lemma tfp-om tp-tfp tpidem)

So trace omega algebras are extensions by definition of trace Kleene algebras with two projection functions and therefore conservative extensions. For every individual trace, in particular, the omega can be defined without these projections.

It is also obvious that a variant of Arden’s rule can be obtained for the trace model which can be used for solving recursive trace equations, for instance in the context of reactive system verification, where trace models are important.
As already mentioned, a special case of trace omega algebras are path omega algebras (cf. [2, 14]). In path algebras, the elements between 0 and 1 are the sets of paths of length one. Sets of paths can again be split into subsets of $P$ and test-free paths. The test-free paths are deflationary, and the omega of a set of paths is obtained like in the case of traces. All further results that hold of trace left omega algebras also hold of path left omega algebras. Additional results about paths dioids and their relationship to trace dioids can be found in [14].

9. Omega, relations and wellfoundedness

Relation omega algebras differ from trace and language omega algebras in that a length-increase argument for showing that an element is deflationary is impossible. In relational models, all elements above 1 are reflexive relations. Their omega is, of course, $\top$ (the full cartesian product). Also, as in the case of trace models, all subidentities are multiplicatively idempotent, hence dense. It follows from Lemma dense-top that each subidentity $R$ of a relation left omega algebra satisfies $R^\omega = R \circ \top$. Also, by Lemma defl-super-id1, deflationary or $\omega$-trivial elements must be irreflexive. It is also clear that each relation can again be split into a subidentity and an irreflexive part. But will $R^\omega$ vanish for all irreflexive relations?

**Lemma 1.** There exists a relation dioid in which some irreflexive relation is not deflationary.

**Proof.** Consider the full relation dioid over the Booleans $\mathbb{B} = \{0, 1\}$. The relation $R$ is depicted in the left-hand diagram below whereas the right-hand diagram shows $\top = \mathbb{B}^2$. It is easy to see that $R \circ \top = \top$ and, obviously, $\top \not= \emptyset$.

Hence the situation is more complex than in trace semirings.

Intuitively, $\omega$-triviality expresses a termination property—the absence of infinite iteration. In relational models it should therefore be related to wellfoundedness. This has already been explored in depth in previous work [8, 14]. The basis of this exploration is the notion of domain semiring [9], which can be defined as follows.

**Class** domain-semiring = dioid + d-op +

assumes d1: $x + (d(x) \cdot x) = d(x) \cdot x$

and d2: $d(x \cdot y) = d(x) \cdot d(y)$

and d3: $d(x) + i = 1$

and d4: $d(0) = 0$

and d5: $d(x + y) = d(x) + d(y)$

**Class** domain-left-kleene-algebra = domain-semiring + left-kleene-algebra

**Class** domain-left-omega-algebra = domain-semiring + left-omega-algebra

For relation semirings over a set a $A$, the domain operation models

$$d(R) = \{ (p, p) \in A \times A : (p, q) \in R \text{ for some } q \in A \},$$

which corresponds to the set of all states $p$ at which the relation $R$ is enabled. It can be shown for any domain semiring $S$ that the set $d(S)$ of all domain elements forms a bounded distributive lattice with minimal element 0 and maximal element 1. In the relation semiring, these elements can be identified with sets of states (formally, they are subidentities). State spaces that form Boolean algebras can be obtained from an alternative axiomatisation which entails the present one [9]. A large number of facts about domain semirings and related algebras can be found in the Isabelle repository and the literature [12, 13].

A Kleene star and omega operator can be added to the signature without any need of modifying the domain axioms.

An element $x$ of a domain semiring $S$ is wellfounded it satisfies the following condition.

**Definition (in domain-semiring)**

$\text{wf} :: a \Rightarrow \text{bool}$

where $\text{wf} x \leftrightarrow (\forall y. d(y) \leq d(x \cdot y) \Rightarrow d(y) = 0)$

The expression $d(x \cdot y) = d(x \cdot d(y))$ models the preimage of the set $d(y)$ under the (abstract) action $x$, that is, the set of all elements in $S$ which are related by $x$ to some element in $d(y)$. If $d(y) \not\subseteq d(x \cdot y)$, then the set $d(y)$ is closed under $x$-actions, hence no element in $d(y)$ can have $x$-maximal elements, that is, elements from which no further $x$ actions are possible. By the above formula, therefore, only the empty set can (vacuously) have $x$-maximal elements. But this means that, in the case of relations, $x$ is wellfounded (or rather Noetherian) in the set-theoretic sense (cf. [8] for further discussion).
First, it is easy to derive Arden’s rule in the context of domain left omega algebras.

**Theorem arden:** \( \text{wf} \ x \rightarrow (z + x \cdot y = y \rightarrow x^* z = y) \)

by (metis add-comm add-zerol annir arden d1 omega-unfold-eq order-refl wf-def)

This fact has already been proved in [10], but in a higher-order setting and using fixpoint fusion. To my knowledge, the present proof is the first one that is entirely within first-order logic and which could be obtained by automated reasoning. However, I could neither prove nor refute Arden’s rule in the weaker setting of domain left Kleene algebra (with \( \top \)). The formula expressing wellfoundedness is very similar to that expressing deflationarity. In fact, it can easily be shown that wellfoundedness implies \( \omega \)-triviality in all domain left omega algebras, but Nitpick refutes that the two conditions are equivalent.

**Lemma ewp-omega:** \( \text{wf} \ x \rightarrow x^0 = 0 \)

by (metis add-comm add-zerol annir d1 omega-unfold-eq order-refl wf-def)

However, wellfoundedness and deflationarity are equivalent under the additional condition \( d(x) \cdot \top = x \cdot \top \), which holds in the relational model (cf. [14]). The proof is simplified by an auxiliary lemma.

**Lemma top-zero:** \( x \cdot \top = 0 \leftrightarrow x = 0 \)

by (metis annir eq-iff max-element min-zero mult-oner top-def)

**Theorem ewp-defl-eq:** \( (\forall x. \ d(x) \cdot \top = x \cdot \top) \rightarrow (\text{defl} \ x \leftrightarrow \text{ewp} \ x) \)

by (smt defl-def leq-def distr top-zero annir defl-trivial omega-unfold-eq order-refl wf-def)

More generally, it can be shown [8, 14] that, in domain semirings,

\[
R^\omega = \nabla(R) \top,
\]

where \( \nabla(R) \) is an element of \( d(A \times A) \) that characterises all those elements of \( A \) from which infinite \( R \)-chains emanate. Hence \( R^\omega \) can again be defined explicitly in this setting. A formal account of this result can be found in the Isabelle repository.

### 10. Salomaa’s axioms

An abstract variant of Arden’s rule plays a prominent role as an axiom in Salomaa’s sound and complete axiomatisation of the algebra of regular events [18]. Salomaa essentially expands dioids by a star operation that satisfies three axioms. The third one uses the empty word property as a side condition. Strictly speaking, his axioms are defined schematically on terms, not axiomatically with first-order variables. As previously I give an algebraic reconstruction of Salomaa’s axiomatisation.

**Class salomaa = dioid + star-op +

fixes ewp :: 'a \Rightarrow bool

assumes ewp-form : ewp x \leftrightarrow (\exists y. x = t + y \land \neg ewp y)

and S11: \((1 + x)^{\ast} = x^{\ast}\)

and S12: \(1 + x^* \cdot x = x^*\)

and salomaa : \((\neg ewp y) \land x = x \cdot y + z \rightarrow x = z \cdot y^*\)

In Salomaa’s axiomatisation, the empty word property is defined as an inductive predicate with respect to term algebras or regular expressions. The expression 1 has the empty word property, whereas the expressions 0 and \( a \), for each constant \( a \), do not have it. The term \( s + t \) has the empty word property if and only if \( s \) or \( t \) has it, and \( s \cdot t \) has the empty word property if and only if \( s \) and \( t \) have it. Finally, \( s^* \) has the empty word property.

I do not give an inductive definition of the empty word property because this would require to axiomatise term algebras of Salomaa algebras, which is involved. Instead I use the abstract condition ewp-form which is well known from language theory. It states that every regular expression which has the empty word property can be written as the sum of 1 and some other term which does not have the empty word property. This reflects the fact that the empty word can be separated from any language that contains it. This condition suffices for the results in this section.

It is also easy to show that in the language model the inductive definition of the empty word property implies \( 1 \leq x \): if \( x \) has the empty word property, then \( 1 \leq x \) by ewp-form. This result also follows in the above class.

**Lemma ewp-one:** \( \text{ewp} \ x \rightarrow 1 \leq x \)

by (metis add-ub ewp-form)

The converse implication can be inductively verified in the language model. Hence in this model the two side conditions on Arden’s rule are equivalent.

It is interesting to explore the relationship between Salomaa’s side condition on Arden’s rule and \( \omega \)-triviality. The following lemma shows that \( \omega \)-triviality implies the absence of the empty word property in left omega algebras. However, even in language left omega algebras, the converse implication does not hold.
Lemma. \( \text{omega-triv-to-neg-ewp: } 0 \neq 1 \rightarrow x^\omega = 0 \rightarrow \neg (\exists y. x = 1 + y \land y^\omega = 0) \)

by (metis add-sub defl-super-id1 defl-trivial)

It is also well known that \( \omega \)-triviality of \( x \) and \( y \) does not imply \( \omega \)-triviality of their sum, which captures Salomaa’s condition of wellfoundedness for sums. Hence, rather unsurprisingly, left omega algebras are too weak to derive the inductive side condition for Salomaa’s axioms.

On the positive side, the axioms of right Kleene algebras can be derived from Salomaa’s axioms, which—up to duality—gives a new simple completeness proof for the latter relative to Boffa’s result mentioned in the introduction.

Since the unfold law of right Kleene algebra is a trivial consequence of Salomaa’s axioms, that satisfies the induction axiom

finally, Nitpick shows that the right induction axiom cannot be derived when in Salomaa’s axioms the empty word property is replaced by \( 1 \leq x \).

11. Completeness of left omega algebras

Regular languages can easily be generalised to \( \omega \)-regular languages by defining \( X^\omega = \{x_0 \cdot x_1 \cdots : x_i \in X\} \), where \( x_0 \cdot x_1 \cdots \) denotes a sequence of type \( \mathbb{N} \rightarrow X \). \( \omega \)-regular expressions can be defined as terms over Wagner algebras [19], which are axiomatised as follows.

A (semi)module is a structure \((S, L, \cdot)\) where \( S \) is a dioid, \( L \) a semilattice with zero \( 0_s \), and \( \cdot \) a scalar product of type \( S \times L \rightarrow L \) that satisfies

\[
\begin{align*}
x : (X + Y) & = x : X + x : Y, \\
(x + y) : X & = x : X + y : Y, \\
(x \cdot y) : X & = x : (y : X), \\
1 : X & = X, \\
0_s : X & = 0, \\
x : 0_w & = 0_w.
\end{align*}
\]

A Kleene module is a module \((K, L, \cdot)\) over a Kleene algebra \( K \) that satisfies the induction axiom

\[
Y + x : X \leq X \Rightarrow x^\omega : Y \leq X.
\]

A Wagner algebra is a Kleene module \((K, L, \cdot)\) expanded by an omega operation \( ^\omega : K \times L \rightarrow L \) that satisfies

\[
\begin{align*}
\neg \text{ewp}(x) & \Rightarrow x^\omega = (x \cdot x^\omega)^\omega, \\
\neg \text{ewp}(x) \land \neg \text{ewp}(y) & \Rightarrow (x \cdot y)^\omega = x : (y \cdot x)^\omega, \\
\neg \text{ewp}(x) & \Rightarrow (x + y)^\omega = y : (x + y)^\omega + X \Rightarrow (x + y)^\omega = y^\omega + y^\omega : X.
\end{align*}
\]
In fact, Wagner uses Salomaa’s axioms for regular expressions, but this makes no difference. Wagner has shown that these axioms are sound and complete for omega regular languages [19]. The empty word property is defined as by Salomaa, hence Wagner’s axioms are again defined on the term algebra.

It turns out that Wagner’s axioms without the side conditions can be derived in left omega algebra.

**Lemma W1:** \((x \cdot x^\omega)\omega = x^\omega\)

\text{by (metis mult-assoc omega-coinduct-var omega-unfold-eq order-refl star-slide-var star-trans-eq antisym star-omega-1)}

**Lemma W2:** \(x \cdot (y \cdot x)^\omega = (x \cdot y)^\omega\)

\text{by (metis omega-unfold-eq mult-isol eq-iff mult-assoc omega-simulation eq-iff mult-assoc)}

**Lemma W3:** \((x+y)^\omega = x \cdot (x+y)^\omega + z \rightarrow (x+y)^\omega = x^\omega + x^\omega \cdot z\)

\text{by (metis add-comm add-lub eq-iff omega-coinduct omega-subdist star-inductl)}

Proofs can be found in the \(\omega\)-algebra file of our Isabelle repository. Similar results for omega algebras can be found in Bolduc’s Master thesis [5].

This result establishes completeness of left omega algebras relative to Wagner’s result. First, by Boffa’s result, the left Kleene algebra axioms are complete for the algebra of regular expressions. They can therefore be used instead of Salomaa’s axioms as a basis for Wagner algebra.

Second, Wagner’s axioms can be derived in left omega algebra when the empty word property and sort constraints have been forgotten. Hence in particular they hold in algebras that satisfy these constraints.

12. Conclusion

Left omega algebras have been studied both abstractly and on regular models given by languages, traces, paths and relations. I introduced abstract side conditions for deriving Arden’s rule which is essential for solving systems of recursive regular equations, and conditions for defining the omega operation explicitly on interesting classes of models. I refined these conditions to meaningful properties in particular models by imposing simple algebraic conditions that hold in individual models. Finally, I gave new completeness proofs for Salomaa’s axioms and left omega algebras. As so often with Kleene algebra, a main achievement is certainly generality and simplicity.

An important model that could not be discussed in this paper is formed by the matrices over omega algebras, which themselves form omega algebras. Our abstract results are, of course, valid in this setting, but particular criteria for unique solvability of linear matrix equations certainly deserve further investigation.

Final open questions are whether a direct proof of Arden’s rule from deflationarity—instead of strong deflationarity—is possible in left Kleene algebras, and whether wellfoundedness implies Arden’s rule already in Kleene algebras with domain in the absence of omega.

Acknowledgements

I would like to thank Michael Laurence for his collaboration and the RAMICS referees for their comments on the conference version of this paper. I am grateful to Jules Desharnais who helped me to improve the journal version. I would also like to apologise to my family for having worked between Christmas and New Year 2011 to meet the special issue deadline on December 31.

References