Syntactic Monoids in a Category

Jiří Adámek, Stefan Milius*, and Henning Urbat

1 Institut für Theoretische Informatik
Technische Universität Braunschweig, Germany
2 Lehrstuhl für Theoretische Informatik
Friedrich-Alexander Universität Erlangen-Nürnberg, Germany

Abstract

The syntactic monoid of a language is generalized to the level of a symmetric monoidal closed category \( \mathcal{D} \). This allows for a uniform treatment of several notions of syntactic algebras known in the literature, including the syntactic monoids of Rabin and Scott (\( \mathcal{D} = \text{sets} \)), the syntactic semirings of Polák (\( \mathcal{D} = \text{semilattices} \)), and the syntactic associative algebras of Reutenauer (\( \mathcal{D} = \text{vector spaces} \)). Assuming that \( \mathcal{D} \) is a commutative variety of algebras, we prove that the syntactic \( \mathcal{D} \)-monoid of a language \( L \) can be constructed as a quotient of a free \( \mathcal{D} \)-monoid modulo the syntactic congruence of \( L \), and that it is isomorphic to the transition \( \mathcal{D} \)-monoid of the minimal automaton for \( L \) in \( \mathcal{D} \). Furthermore, in the case where the variety \( \mathcal{D} \) is locally finite, we characterize the regular languages as precisely the languages with finite syntactic \( \mathcal{D} \)-monoids.

1998 ACM Subject Classification F.4.3 Formal Languages

Keywords and phrases Syntactic monoid, transition monoid, algebraic automata theory, duality, coalgebra, algebra, symmetric monoidal closed category, commutative variety

1 Introduction

One of the successes of the theory of coalgebras is that ideas from automata theory can be developed at a level of abstraction where they apply uniformly to many different types of systems. In fact, classical deterministic automata are a standard example of coalgebras for an endofunctor. And that automata theory can be studied with coalgebraic methods rests on the observation that formal languages form the final coalgebra.

The present paper contributes to a new category-theoretic view of algebraic automata theory. In this theory one starts with an elegant machine-independent notion of language recognition: a language \( L \subseteq X^* \) is recognized by a monoid morphism \( e : X^* \to M \) if it is the preimage under \( e \) of some subset of \( M \). Regular languages are then characterized as precisely the languages recognized by finite monoids. A key concept, introduced by Rabin and Scott [20] (and earlier in unpublished work of Myhill), is the syntactic monoid of a language \( L \). It serves as a canonical algebraic recognizer of \( L \), namely the smallest \( X \)-generated monoid recognizing \( L \). Two standard ways to construct the syntactic monoid are:

(1) as a quotient of the free monoid \( X^* \) modulo the syntactic congruence of \( L \), which is a two-sided version of the well-known Myhill-Nerode equivalence, and
(2) as the transition monoid of the minimal automaton for \( L \).

In addition to syntactic monoids there are several related notions of syntactic algebras for (weighted) languages in the literature, most prominently the syntactic idempotent semirings of Polák [19] and the syntactic associative algebras of Reutenauer [21], both of which admit

* Stefan Milius acknowledges support by the Deutsche Forschungsgemeinschaft (DFG) under project MI 717/5-1
constructions similar to (1) and (2). A crucial observation is that monoids, idempotent semirings and associative algebras are precisely the monoid objects in the categories of sets, semilattices and vector spaces, respectively. Moreover, these three categories are symmetric monoidal closed w.r.t. their usual tensor product.

The main goal of our paper is thus to develop a theory of algebraic recognition in a general symmetric monoidal closed category \( \mathcal{D} = (\mathcal{D}, \otimes, I) \). Following Goguen [12], a language in \( \mathcal{D} \) is a morphism \( L : X^* \to Y \) where \( X \) is a fixed object of inputs, \( Y \) a fixed object of outputs, and \( X^* \) denotes the free \( \mathcal{D} \)-monoid on \( X \). And a \( \mathcal{D} \)-automaton is given by the picture below: it consists of an object of states \( Q \), a morphism \( i \) representing the initial state, an output morphism \( f \), and a transition morphism \( \delta \) which may be presented in its curried form \( \lambda \delta \).

\[
\begin{array}{c}
X \otimes Q \\
i i \\
\downarrow \delta \\
Q \\
\downarrow f \\
Y \\
\end{array}
\]

This means that an automaton is at the same time an algebra \( I + X \otimes Q \xrightarrow{[i, \delta]} Q \) for the functor \( FQ = I + X \otimes Q \), and a coalgebra \( Q \xrightarrow{(f, \lambda \delta)} Y \times [X, Q] \) for the functor \( TQ = Y \times [X, Q] \). It turns out that much of the classical (co-)algebraic theory of automata in the category of sets extends to this level of generality. Thus Goguen [12] demonstrated that the initial algebra for \( F \) coincides with the free \( \mathcal{D} \)-monoid \( X^* \), and that every language is accepted by a unique minimal \( \mathcal{D} \)-automaton. We will add to this picture the observation that the final coalgebra for \( T \) is carried by the object of languages \( [X^*, Y] \), see Proposition 2.21.

In Section 3 we introduce the central concept of our paper, the syntactic \( \mathcal{D} \)-monoid of a language \( L : X^* \to Y \), which by definition is the smallest \( X \)-generated \( \mathcal{D} \)-monoid recognizing \( L \). Assuming that \( \mathcal{D} \) is a commutative variety of algebras, we will show that the above constructions (1) and (2) for the classical syntactic monoid adapt to our general setting: the syntactic \( \mathcal{D} \)-monoid is (1) the quotient of \( X^* \) modulo the syntactic congruence of \( L \) (Theorem 3.14), and (2) the transition \( \mathcal{D} \)-monoid of the minimal \( \mathcal{D} \)-automaton for \( L \) (Theorem 4.6). As special instances we recover the syntactic monoids of Rabin and Scott (\( \mathcal{D} = \text{sets} \)), the syntactic semirings of Polák (\( \mathcal{D} = \text{semilattices} \)) and the syntactic associative algebras of Reutenauer (\( \mathcal{D} = \text{vector spaces} \)). Furthermore, our categorical setting yields new types of syntactic algebras “for free”. For example, we will identify monoids with zero as the algebraic structures representing partial automata (the case \( \mathcal{D} = \text{pointed sets} \)), which leads to the syntactic monoid with zero for a given language. Similarly, by taking as \( \mathcal{D} \) the variety of algebras with an involutive unary operation we obtain syntactic involution monoids.

Most of the results of our paper apply to arbitrary languages. In Section 5 we will investigate \( \mathcal{D} \)-regular languages, that is, languages accepted by \( \mathcal{D} \)-automata with a finitely presentable object of states. Under suitable assumptions on \( \mathcal{D} \), we will prove that a language is \( \mathcal{D} \)-regular iff its syntactic \( \mathcal{D} \)-monoid is carried by a finitely presentable object (Theorem 5.4). We will also derive a dual characterization of the syntactic \( \mathcal{D} \)-monoid which is new even in the “classical” case \( \mathcal{D} = \text{sets} \): if \( \mathcal{D} \) is a locally finite variety, and if moreover some other locally finite variety \( \mathcal{C} \) is dual to \( \mathcal{D} \) on the level of finite objects, the syntactic \( \mathcal{D} \)-monoid of \( L \) dualizes to the local variety of languages in \( \mathcal{C} \) generated by the reversed language of \( L \).

Due to space limitations most proofs are omitted or sketched. See [1] for an extended version of this paper.

**Related work.** Our paper gives a uniform treatment of various notions of syntactic algebras known in the literature [19, 20, 21]. Another categorical approach to (classical)
syntactic monoids appears in the work of Ballester-Bolinches, Cosme-Llopes and Rutten [5]. These authors consider automata in the category of sets specified by equations or dually by coequations, which leads to a construction of the automaton underlying the syntactic monoid of a language. The fact that it forms the transition monoid of a minimal automaton is also interpreted in that setting. In the present paper we take a more general and conceptual approach by studying algebraic recognition in a symmetric monoidal closed category $\mathcal{D}$. One important source of inspiration for our categorical setting was the work of Goguen [12].

In the recent papers [2, 4] we presented a categorical view of varieties of languages, another central topic of algebraic automata theory. Building on the duality-based approach of Gehrke, Grigorieff and Pin [11], we generalized Eilenberg’s variety theorem and its local version to the level of an abstract (pre-)duality between algebraic categories. The idea to replace monoids by monoid objects in a commutative variety $\mathcal{D}$ originates in this work.

When revising this paper we were made aware of the ongoing work of Bojanczyk [8]. He considers, in lieu of commutative varieties, categories of Eilenberg-Moore algebras for an arbitrary monad on sorted sets, and defines syntactic congruences in this more general setting. Our Theorem 3.14 is a special case of [8, Theorem 3.1].

## 2 Preliminaries

Throughout this paper we work with deterministic automata in a commutative variety $\mathcal{D}$ of algebras. Recall that a variety of algebras is an equational class of algebras over a finitary signature. It is called commutative (or entropic) if, for any two objects $A$ and $B$ of $\mathcal{D}$, the set $\mathcal{D}(A,B)$ of all homomorphisms from $A$ to $B$ carries a subobject $[A,B] \hookrightarrow B^{|A|}$ of the product of $|A|$ copies of $B$. Commutative varieties are precisely the categories of Eilenberg-Moore algebras for a commutative finitary monad on the category of sets, see [13, 16]. We fix an object $X$ (of inputs) and an object $Y$ (of outputs) in $\mathcal{D}$.

- **Example 2.1.** 1. Set is a commutative variety with $[A,B] = B^A$.

2. A pointed set $(A, \bot)$ is a set $A$ together with a chosen point $\bot \in A$. The category Set$_\bot$ of pointed sets and point-preserving functions is a commutative variety. The point of $[(A, \bot_A), (B, \bot_B)]$ is the constant function with value $\bot_B$.

3. An involution algebra is a set with an involutive unary operation $x \mapsto \tilde{x}$, i.e. $\tilde{\tilde{x}} = x$. We call $\tilde{x}$ the complement of $x$. Morphisms are functions $f$ with $f(\tilde{x}) = \tilde{f(x)}$. The variety Inv of involution algebras is commutative. Indeed, the set $[A,B]$ of all homomorphisms is an involution algebra with pointwise complementation: $\tilde{f}$ sends $x$ to $\tilde{f(x)}$.

4. All other examples we treat in our paper are varieties of modules over a semiring. Given a semiring $S$ (with 0 and 1) we denote by $\text{Mod}(S)$ the category of all $S$-modules and module homomorphisms (i.e. $S$-linear maps). Three interesting special cases of $\text{Mod}(S)$ are:

   a. $S = \{0, 1\}$, the boolean semiring with $1 + 1 = 1$: the category $\text{JSL}_0$ of join-semilattices with 0, and homomorphisms preserving joins and 0;

   b. $S = \mathbb{Z}$: the category $\text{Ab}$ of abelian groups and group homomorphisms;

   c. $S = K$ (a field): the category $\text{Vec}(K)$ of vector spaces over $K$ and linear maps.

- **Notation 2.2.** We denote by $\Psi : \text{Set} \to \mathcal{D}$ the left adjoint to the forgetful functor $[-] : \mathcal{D} \to \text{Set}$. Thus $\Psi X_0$ is the free object of $\mathcal{D}$ on the set $X_0$.

- **Example 2.3.** 1. We have $\Psi X_0 = X_0$ for $\mathcal{D} = \text{Set}$ and $\Psi X_0 = X_0 + \{\bot\}$ for $\mathcal{D} = \text{Set}_\bot$. 
2. For $\mathcal{D} = \text{Inv}$ the free involution algebra on $X_0$ is $\Psi X_0 = X_0 + \overline{X}_0$ where $\overline{X}_0$ is a copy of $X_0$ (whose elements are denoted $\overline{x}$ for $x \in X_0$). The involution swaps the copies of $X_0$, and the universal arrow $X_0 \to \Psi X_0 + \overline{X}_0$ is the left coproduct injection.

3. For $\mathcal{D} = \text{Mod}(S)$ the free module $\Psi X_0$ is the submodule of $S^{X_0}$ on all functions $X_0 \to S$ with finite support. Equivalently, $\Psi X_0$ consists of formal linear combinations $\sum_{i=1}^{n} s_i x_i$ with $s_i \in S$ and $x_i \in X_0$. In particular, $\Psi X_0 = \mathcal{P}_f X_0$ (finite subsets of $X_0$) for $\mathcal{D} = \text{JSL}_0$, and $\Psi X_0$ is the vector space with basis $X_0$ for $\mathcal{D} = \text{Vec}(K)$.

**Definition 2.4.** Given objects $A$, $B$ and $C$ of $\mathcal{D}$, a bimorphism from $A$, $B$ to $C$ is a function $f : |A| \times |B| \to |C|$ such that the maps $f(a,-) : |B| \to |C|$ and $f(-,b) : |A| \to |C|$ carry morphisms of $\mathcal{D}$ for every $a \in |A|$ and $b \in |B|$. A tensor product of $A$ and $B$ is a universal bimorphism $t : |A| \times |B| \to |A \otimes B|$, which means that for every bimorphism $f : |A| \times |B| \to |C|$ there is a unique morphism $f' : A \otimes B \to C$ in $\mathcal{D}$ with $f' \cdot t = f$.

**Theorem 2.5** (Banachiewski and Nelson [6]). Every commutative variety $\mathcal{D}$ has tensor products, making $\mathcal{D} = (\mathcal{D}, \otimes, I)$ with $I = \Psi 1$ a symmetric monoidal closed category. That is, we have the following bijective correspondence of morphisms, natural in $A, B, C \in \mathcal{D}$:

$$f : A \otimes B \to C$$

$$\lambda f : A \to \{B, C\}$$

**Remark 2.6.** Recall that a monoid $(M, m, i)$ in a monoidal category $(\mathcal{D}, \otimes, I)$ (with tensor product $\otimes : \mathcal{D} \times \mathcal{D} \to \mathcal{D}$ and tensor unit $I \in \mathcal{D}$) is an object $M$ equipped with a multiplication $m : M \otimes M \to M$ and unit $i : I \to M$ satisfying the usual associative and unit laws. Due to $\otimes$ and $I$ representing bimorphisms, this categorical definition is equivalent to the following algebraic one in our setting: a $\mathcal{D}$-monoid is a triple $(M, \cdot, i)$ where $M$ is an object of $\mathcal{D}$ and $(|M|, \cdot, i)$ is a monoid in $\text{Set}$ with $\cdot : |M| \times |M| \to |M|$ a bimorphism of $\mathcal{D}$. A morphism $h : (M, \cdot, i) \to (M', \cdot', i')$ of $\mathcal{D}$-monoids is a morphism $h : M \to M'$ of $\mathcal{D}$ such that $|h| : |M| \to |M'|$ is a monoid morphism in $\text{Set}$. We denote by $\text{Mon}(\mathcal{D})$ the category of $\mathcal{D}$-monoids and their homomorphisms. In the following we will freely work with $\mathcal{D}$-monoids in both categorical and algebraic disguise.

**Example 2.7.** 1. In $\text{Set}$ the tensor product is the cartesian product, $I = \{\ast\}$, and $\text{Set}$-monoids are ordinary monoids.

2. In $\text{Set}_1$ we have $I = \{\bot, \ast\}$, and the tensor product of pointed sets $(A, \bot_A)$ and $(B, \bot_B)$ is $A \otimes B = (A \setminus \{\bot_A\}) \times (B \setminus \{\bot_B\}) \cup \{\bot\}$. $\text{Set}_1$-monoids are precisely monoids with zero. Indeed, given a $\text{Set}_1$-monoid structure on $(A, \bot)$ we have $x \cdot \bot = \bot = \bot \cdot x$ for all $x$ because $\bot$ is a bimorphism, i.e. $\bot$ is a zero element. Morphisms of $\text{Mon}(\text{Set}_1)$ are zero-preserving monoid morphisms.

3. An $\text{Inv}$-monoid (also called an involution monoid) is a monoid equipped with an involution $x \mapsto \overline{x}$ such that $x \overline{y} = \overline{y} \overline{x} = x \overline{y}$. For example, for any set $A$ the power set $\mathcal{P} A$ naturally carries the structure of an involution monoid: the involution takes complements, $\overline{\overline{S}} = A \setminus \overline{S}$, and the monoid multiplication is the symmetric difference $S \oplus T = (S \setminus T) \cup (T \setminus S)$.

4. $\text{JSL}_0$-monoids are precisely idempotent semirings (with 0 and 1). Indeed, a $\text{JSL}_0$-monoid on a semilattice (i.e. a commutative idempotent monoid) $(D, +, 0)$ is given by a unit 1 and a monoid multiplication that, being a bimorphism, distributes over $+$ and 0.

5. More generally, a $\text{Mod}(S)$-monoid is precisely an associative algebra over $S$: it consists of an $S$-module together with a unit 1 and a monoid multiplication that distributes over $+$ and 0 and moreover preserves scalar multiplication in both components.

**Notation 2.8.** We denote by $X^{\otimes n}$ ($n < \omega$) the $n$-fold tensor power of $X$, recursively defined by $X^{\otimes 0} = I$ and $X^{\otimes (n+1)} = X \otimes X^{\otimes n}$.
Proposition 2.9 (see Mac Lane [15]). The forgetful functor \( \text{Mon}(\mathcal{D}) \to \mathcal{D} \) has a left adjoint assigning to every object \( X \) the free \( \mathcal{D} \)-monoid \( X^* = \coprod_{n<\omega} X^\otimes n \). The monoid structure \((X^*, m_X, i_X)\) is given by the coproduct injection \( i_X : I = X^\otimes 0 \to X^* \) and \( m_X : X^* \otimes X^* \to X^* \), where \( X^* \otimes X^* = \coprod_{n,k<\omega} X^\otimes n \otimes X^\otimes k \) and \( m_X \) has as its \((n,k)\)-component the \((n+k)\)-th coproduct injection. The universal arrow \( \eta_X : X \to X^* \) is the first coproduct injection.

Proposition 2.10. The free \( \mathcal{D} \)-monoid on \( X = \Psi X_0 \) is \( X^\bullet = \Psi X_0^\bullet \). Its monoid multiplication extends the concatenation of words in \( X_0^\bullet \), and its unit is the empty word \( \varepsilon \).

Example 2.11. 1. In \( \text{Set} \) we have \( X^\bullet = X^\bullet \). In \( \text{Set}_\perp \) with \( X = \Psi X_0 = X_0 + \{ \perp \} \) we get \( X^\bullet = X_0^\bullet + \{ \perp \} \). The product \( x \cdot y \) is concatenation for \( x, y \in X_0^\bullet \), and otherwise \( \perp \).

2. In \( \text{Inv} \) with \( X = \Psi X_0 = X_0 + X_0 \) we have \( X^\bullet = X_0^\bullet + X_0^\bullet \). The multiplication restricted to \( X_0^\bullet \) is concatenation, and is otherwise determined by \( \tilde{u} \cdot \tilde{v} = \tilde{u} \cdot \tilde{v} \) for \( u, v \in X_0^\bullet \).

3. In \( \text{JSL}_0 \) with \( X = \Psi X_0 = \mathcal{P}_I X_0 \) we have \( X^\bullet = \mathcal{P}_I X_0^\bullet \), the semiring of all finite languages over \( X_0 \). Its addition is union and its multiplication is the concatenation of languages.

4. More generally, in \( \text{Mod}(S) \) with \( X = \Psi X_0 \) we get \( X^\bullet = \Psi X_0^\bullet = [X_0^\bullet] \), the module of all finite \( S \)-weighted languages over the alphabet \( X_0 \). Hence the elements of \([X_0^\bullet]\) are functions \( c : X_0^\bullet \to S \) with finite support, which may be expressed as polynomials
\[
\sum_{i=1}^n c(w_i) w_i \text{ with } w_i \in X_0^\bullet \text{ and } c(w_i) \in S.
\]
The \( S \)-algebraic structure of \([X_0^\bullet]\) is given by the usual addition, scalar multiplication and product of polynomials.

Definition 2.12 (Goguen [12]). A \( \mathcal{D} \)-automaton \((Q, \delta, i, f)\) consists of an object \( Q \) (of states) and morphisms \( \delta : X \otimes Q \to Q \), \( i : I \to Q \) and \( f : Q \to Y \); see Diagram (1). An automata homomorphism \( h : (Q, \delta, i, f) \to (Q', \delta', i', f') \) is a morphism \( h : Q \to Q' \) preserving transitions as well as initial states and outputs, i.e. making the following diagrams commute:

\[
\begin{array}{ccc}
X \otimes Q & \xrightarrow{i} & Q \\
\downarrow \delta & & \downarrow h \\
X \otimes Q' & \xrightarrow{i'} & Q'
\end{array}
\quad
\begin{array}{ccc}
I & \xrightarrow{i} & Q \\
\downarrow \delta & & \downarrow h \\
I' & \xrightarrow{i'} & Q'
\end{array}
\quad
\begin{array}{ccc}
Y & \xrightarrow{f} & Y \\
\downarrow & & \downarrow \\
Y & \xrightarrow{f'} & Y
\end{array}
\]

The above definition makes sense in any monoidal category \( \mathcal{D} \). In our setting, since \( I = \Psi 1 \), the morphism \( i \) chooses an initial state in \([Q]\). Moreover, if \( X = \Psi X_0 \) for some set \( X_0 \) (of inputs), the morphism \( \delta \) amounts to a choice of endomorphisms \( \delta_a : Q \to Q \) for \( a \in X_0 \), representing transitions. This follows from the bijections

\[
\begin{array}{ccc}
\Psi X_0 \otimes Q & \to & Q \\
\downarrow \text{in } \mathcal{D} & & \downarrow \text{in } \mathcal{D} \\
\Psi X_0 \otimes [Q, Q] & \xrightarrow{\delta} & [Q, Q] \\
\downarrow \text{in } \mathcal{D}(Q, Q) & & \downarrow \text{in } \text{Set}
\end{array}
\]

Example 2.13. 1. The classical deterministic automata are the case \( \mathcal{D} = \text{Set} \) and \( Y = \{0, 1\} \). Here \( f : Q \to \{0, 1\} \) defines the set \( F = f^{-1}[1] \subseteq Q \) of final states. For general \( Y \) we get deterministic Moore automata with outputs in \( Y \).

2. The setting \( \mathcal{D} = \text{Set}_\perp \) with \( X = X_0 + \{ \perp \} \) and \( Y = \{ \perp, 1 \} \) gives partial deterministic automata. Indeed, the state object \((Q, \perp)\) has transitions \( \delta_a : (Q, \perp) \to (Q, \perp) \) for \( a \in X_0 \) preserving \( \perp \), that is, \( \perp \) is a sink state. Equivalently, we may consider \( \delta_a \) as a partial transition map on the state set \( Q \setminus \{ \perp \} \). The morphism \( f : (Q, \perp) \to \{ \perp, 1 \} \) again determines a set of final states \( F = f^{-1}[1] \) (in particular, \( \perp \) is non-final). And the morphism \( i : \{ \perp, * \} \to (Q, \perp) \) determines a partial initial state: either \( i(*) \) lies in \( Q \setminus \{ \perp \} \), or no initial state is defined.
3. In \( \mathcal{D} = \text{Inv} \) let us choose \( X = X_0 + \tilde{X}_0 \) and \( Y = \{0, 1\} \) with \( \tilde{0} = 1 \). An \( \text{Inv} \)-automaton is a deterministic automaton with complementary states \( x \mapsto \tilde{x} \) such that (i) for every transition \( p \xrightarrow{a} q \) there is a complementary transition \( \tilde{p} \xrightarrow{\tilde{a}} \tilde{q} \) and (ii) a state \( q \) is final iff \( \tilde{q} \) is non-final.

4. For \( \mathcal{D} = \text{JSL}_0 \) with \( X = \mathcal{P}_J X_0 \) and \( Y = \{0, 1\} \) (the two-chain) an automaton consists of a semilattice \( Q \) of states, transitions \( \delta_a : Q \to Q \) for \( a \in X_0 \) preserving finite joins (including 0), an initial state \( i \in Q \) and a homomorphism \( f : Q \to \{0, 1\} \) which defines a prime upset \( F = f^{-1}[1] \subseteq Q \) of final states. The latter means that a finite join of states is final iff one of the states is. In particular, 0 is non-final.

5. More generally, automata in \( \mathcal{D} = \text{Mod}(\mathcal{S}) \) with \( X = \Psi X_0 \) and \( Y = \mathcal{S} \) are \( \mathcal{S} \)-weighted automata. Such an automaton consists of an \( \mathcal{S} \)-module \( Q \) of states, linear transitions \( \delta_a : Q \to Q \) for \( a \in X_0 \), an initial state \( i \in Q \) and a linear output map \( f : Q \to \mathcal{S} \).

\( \triangleright \) \textbf{Remark 2.14.} 1. An algebra for an endofunctor \( F \) of \( \mathcal{D} \) is a pair \((Q, \alpha)\) of an object \( Q \) and a morphism \( \alpha : FQ \to Q \). A homomorphism \( h : (Q, \alpha) \to (Q', \alpha') \) of \( F \)-algebras is a morphism \( h : Q \to Q' \) with \( h \cdot \alpha = \alpha' \cdot Fh \). Throughout this paper we work with the endofunctor \( FQ = I + X \otimes Q \); its algebras are denoted as triples \((Q, \delta, i)\) with \( \delta : X \otimes Q \to Q \) and \( i : I \to Q \). Hence \( \mathcal{D} \)-automata are precisely \( F \)-algebras equipped with an output morphism \( f : Q \to Y \). Moreover, automata homomorphisms are precisely \( F \)-algebra homomorphisms preserving outputs.

2. Analogously, a \( \text{coalgebra} \) for an endofunctor \( T \) of \( \mathcal{D} \) is a pair \((Q, \gamma)\) of an object \( Q \) and a morphism \( \gamma : Q \to TQ \). Throughout this paper we work with the endofunctor \( TQ = Y \times [X, Q] \); its coalgebras are denoted as triples \((Q, \tau, f)\) with \( \tau : Q \to [X, Q] \) and \( f : Q \to Y \). Hence \( \mathcal{D} \)-automata are precisely pointed \( T \)-coalgebras, i.e. \( T \)-coalgebras equipped with a morphism \( i : I \to Q \). Indeed, a pointed coalgebra \( I \xrightarrow{i} Q \xrightarrow{(f, \tau)} Y \times [X, Q] \), the morphism \( Q \xrightarrow{\delta} [X, Q] \) is the curried form of a morphism \( Q \otimes X \xrightarrow{\delta} X \otimes Q \xrightarrow{\tau} Q \). Automata homomorphisms are \( T \)-coalgebra homomorphisms preserving initial states.

\( \triangleright \) \textbf{Definition 2.15.} Given a \( \mathcal{D} \)-monoid \((M, m, i)\) and a morphism \( e : X \to M \) of \( \mathcal{D} \), the \textit{\( F \)-algebra associated to \( M \) and \( e \)} has carrier \( M \) and structure

\[
[i, \delta] = (I + X \otimes M \xrightarrow{1 + \otimes M} I + M \otimes M \xrightarrow{[i, m]} M).
\]

In particular, the \( F \)-algebra associated to the free monoid \( X^\ast \) (and its universal arrow \( \eta_X \)) is

\[
[i_X, \delta_X] = (I + X \otimes X^\ast \xrightarrow{1 + \eta_X \otimes X^\ast} I + X^\ast \otimes X^\ast \xrightarrow{[i_X, m_X]} X^\ast).
\]

\( \triangleright \) \textbf{Example 2.16.} In \( \text{Set} \) every monoid \( M \) together with an “input” map \( e : X \to M \) determines an \( F \)-algebra with initial state \( i \) and transitions \( \delta_a = - \cdot e(a) \) for all \( a \in X \). The \( F \)-algebra associated to \( X^\ast \) is the usual automaton of words: its initial state is \( e \) and the transitions are given by \( w \xrightarrow{a} wa \) for \( a \in X \).

\( \triangleright \) \textbf{Proposition 2.17} (Goguen [12]). For any symmetric monoidal closed category \( \mathcal{D} \) with countable coproducts, \( X^\ast \) is the initial algebra for \( F \).

\( \triangleright \) \textbf{Remark 2.18.} Given any \( F \)-algebra \((Q, \delta, i)\) the unique \( F \)-algebra homomorphism \( \epsilon_Q : X^\ast \to Q \) is constructed as follows: extend the morphism \( \lambda \delta : X \to [Q, Q] \) to a \( \mathcal{D} \)-monoid morphism \( \lambda \delta^+ : X^\ast \to [Q, Q] \). Then

\[
\epsilon_Q = (X^\ast \cong X^\ast \otimes I \xrightarrow{\lambda \delta^+ \otimes 1} [Q, Q] \otimes Q \xrightarrow{\sigma} Q),
\]

where \( \sigma \) is the ‘evaluation morphism’, i.e. the counit of the adjunction \(- \otimes Q \vdash [Q, -]\).
\textbf{Notation 2.24.} \(\delta^*: X^* \otimes Q \to Q\) denotes the uncurried form of \(\lambda \delta^+: X^* \to [Q, Q]\).

\textbf{Remark 2.20.} Recall from Rutten [22] that the final coalgebra for the functor \(TQ = \{0, 1\} \times Q^2\) on \(\text{Set}\) is the coalgebra \(\mathcal{P}X^* \cong [X^*, \{0, 1\}])\) of all languages over \(X\). Given any coalgebra \(Q\), the unique coalgebra homomorphism from \(Q\) to \(\mathcal{P}X^*\) assigns to every state \(q\) the language accepted by \(q\) (as an initial state). These observations generalize to our present setting. The object \([X^*, Y]\) of \(D\) carries the following \(T\)-coalgebra structure: its transition morphism \(\tau_{[X^*, Y]}: [X^*, Y] \to [X, [X^*, Y]]\) is the two-fold currying of

\[
\begin{array}{c}
[X^*, Y] \otimes X \otimes X^* \xrightarrow{[X^*, Y] \otimes x \otimes X^*} [X^*, Y] \otimes X^* \otimes X^* \\
\xrightarrow{[X^*, Y] \otimes m \otimes x} [X^*, Y] \otimes X^* \overset{ev}{\to} Y,
\end{array}
\]

and its output morphism \(f_{[X^*, Y]}: [X^*, Y] \to Y\) is

\[
f_{[X^*, Y]} = ([X^*, Y] \otimes I) \xrightarrow{[X^*, Y] \otimes x} [X^*, Y] \otimes X^* \overset{ev}{\to} Y.
\]

\textbf{Proposition 2.21.} \([X^*, Y]\) is the final coalgebra for \(T\).

\textbf{Proof sketch.} Given any coalgebra \((Q, \tau, f)\), let \(\delta : X \otimes Q \to Q\) be the uncurried version of \(\tau : Q \to [X, Q]\), see Remark 2.14. Then the unique coalgebra homomorphism into \([X^*, Y]\) is \(\lambda h : Q \to [X^*, Y]\), where \(h = (Q \otimes X^* \cong X^* \otimes Q \overset{\delta}{\to} Q \overset{f}{\to} Y)\).

\textbf{Definition 2.22 (Goguen [12]).} A \emph{language} in \(D\) is a morphism \(L : X^* \to Y\).

Note that if \(X = \Psi X_0\) (and hence \(X^* = \Psi X_0^*\)) for some set \(X_0\), one can identify a language \(L : X^* = \Psi X_0^* \to Y\) in \(D\) with its adjoint transpose \(\tilde{L} : X_0^* \to |Y|\), via the adjunction \(\Psi \\dashv |\dashv : D \to \text{Set}\). In the case where \(|Y|\) is a two-element set, \(\tilde{L}\) is the characteristic function of a “classical” language \(L_0 \subseteq X_0^*\).

\textbf{Example 2.23.} 1. In \(D = \text{Set}\) (with \(X^* = X^*\) and \(Y = \{0, 1\}\)) one represents \(L_0 \subseteq X^*\) by its characteristic function \(L : X^* \to \{0, 1\}\).

2. In \(D = \text{Set}_\bot\) (with \(X = X_0 + \{\bot\}\), \(X^* = X_0^* + \{\bot\}\) and \(Y = \{\bot, 1\}\)) one represents \(L_0 \subseteq X_0^*\) by its extended characteristic function \(L : X_0^* + \{\bot\} \to \{\bot, 1\}\) where \(L(\bot) = \bot\).

3. In \(D = \text{Inv}\) (with \(X = X_0 + \tilde{X}_0\), \(X^* = X_0^* + \tilde{X}_0^*\) and \(Y = \{0, 1\}\)) one represents \(L_0 \subseteq X_0^*\) by \(L : X_0^* + \tilde{X}_0^* \to \{0, 1\}\) where \(L(w) = 1\) iff \(w \in L_0\) and \(L(\tilde{w}) = 1\) iff \(w \notin L_0\) for all words \(w \in X_0^*\).

4. In \(D = \text{JSL}_0\) (with \(X = \mathcal{P}X_0\), \(X^* = \mathcal{P}X_0^*\) and \(Y = \{0, 1\}\)) one represents \(L_0 \subseteq X_0^*\) by \(L : \mathcal{P}X_0^* \to \{0, 1\}\) where \(L(U) = 1\) iff \(U \cap L_0 \neq \emptyset\).

5. In \(D = \text{Mod}(S)\) (with \(X = \Psi X_0\), \(X^* = S[X_0]\) and \(Y = S\)) an \(S\)-weighted language \(L_0 : X_0^* \to S\) is represented by its free extension to a module homomorphism

\[
L : S[X_0^*] \to S, \quad L \left( \sum_{i=1}^n c(w_i)w_i \right) = \sum_{i=1}^n c(w_i)L_0(w_i).
\]

\textbf{Definition 2.24 (Goguen [12]).} The \emph{language accepted} by a \(D\)-automaton \((Q, \delta, i, f)\) is \(L_Q = (X^* \overset{\delta_i}{\to} Q \overset{f}{\to} Y)\), where \(c_Q\) is the \(F\)-algebra homomorphism of Remark 2.18.

\textbf{Example 2.25.} 1. In \(D = \text{Set}\) with \(Y = \{0, 1\}\), the homomorphism \(e_Q : X^* \to Q\) assigns to every word \(w\) the state it computes in \(Q\), i.e. the state the automaton reaches on input \(w\). Thus \(L_Q(w) = 1\) iff \(Q\) terminates in a final state on input \(w\), which is precisely the standard definition of the accepted language of an automaton. For general \(Y\), the function \(L_Q : X^* \to Y\) is the behavior of the Moore automaton \(Q\), i.e. \(L_Q(w)\) is the output of the last state in the computation of \(w\).
2. For $\mathcal{D} = \text{Set}_\perp$ with $X = X_0 + \{\perp\}$ and $Y = \{\perp, 1\}$, we have $e_Q : X_0^* + \{\perp\} \to (Q, \perp)$ sending $\perp$ to $\perp$, and sending a word in $X_0^*$ to the state it computes (if any), and to $\perp$ otherwise. Hence $L_Q : X_0^* + \{\perp\} \to \{\perp, 1\}$ defines (via the preimage of 1) the usual language accepted by a partial automaton.

3. In $\mathcal{D} = \text{Inv}$ with $X = X_0 + \widetilde{X}_0$ and $Y = \{0, 1\}$, the map $L_Q : X_0^* + \widetilde{X}_0^* \to \{0, 1\}$ sends $w \in X_0^*$ to 1 iff $w$ computes a final state, and it sends $\widetilde{w} \in \widetilde{X}_0^*$ to 1 iff $\widetilde{w}$ computes a non-final state.

4. In $\mathcal{D} = \text{JSL}_0$ with $X = \mathcal{P}X_0$ and $Y = \{0, 1\}$, the map $L_Q : \mathcal{P}X_0^* \to \{0, 1\}$ assigns to $U \in \mathcal{P}X_0^*$ the value 1 iff the computation of at least one word in $U$ ends in a final state.

5. In $\mathcal{D} = \text{Mod}(S)$ with $X = \Psi X_0$ and $Y = S$, the map $L_Q : S[X_0^*] \to S$ assigns to $\sum_{i=1}^n c(w_i)w_i$ the value $\sum_{i=1}^n c(w_i)y_i$, where $y_i$ is the output of the state $Q$ reaches on input $w_i$. Taking $Q = S^n$ for some natural number $n$ yields a classical $n$-state weighted automaton, and in this case one can show that the restriction of $L_Q$ to $X_0^*$ is the usual language of a weighted automaton.

> **Remark 2.26.** By Remark 2.14 every $\mathcal{D}$-automaton $(Q, \delta, i, f)$ is an $F$-algebra as well as a $T$-coalgebra. Our above definition of $L_Q$ was purely algebraic. The corresponding coagebraic definition uses the unique coalgebra homomorphism $c_Q : Q \to [X^*, Y]$ into the final $T$-coalgebra and precomposes with $i : I \to Q$ to get a morphism $c_Q \cdot i : I \to [X^*, Y]$ (choosing a language, i.e. an element of $[X^*, Y]$). Unsurprisingly, the results are equal:

> **Proposition 2.27.** The language $L_Q : X^* \to Y$ of an automaton $(Q, \delta, i, f)$ is the uncurried form of the morphism $c_Q \cdot i : I \to [X^*, Y]$.

### 3 Algebraic Recognition and Syntactic $\mathcal{D}$-Monoids

In classical algebraic automata theory one considers recognition of languages by (ordinary) monoids in lieu of automata. One key concept is the **syntactic monoid** which is characterized as the smallest monoid recognizing a given language. There are also related concepts of canonical algebraic recognizers in the literature, e.g. the syntactic idempotent semiring and the syntactic associative algebra. In this section we will give a uniform account of algebraic language recognition in our categorical setting. Our main result is the definition and construction of a minimal algebraic recognizer, the **syntactic $\mathcal{D}$-monoid** of a language.

> **Definition 3.1.** A $\mathcal{D}$-monoid morphism $e : X^* \to M$ recognizes the language $L : X^* \to Y$ if there exists a morphism $f : M \to Y$ of $\mathcal{D}$ with $L = f \cdot e$.

> **Example 3.2.** We use the notation of Example 2.23.

1. $\mathcal{D} = \text{Set}$ with $X^* = X^*$ and $Y = \{0, 1\}$: given a monoid $M$, a function $f : M \to \{0, 1\}$ defines a subset $F = f^{-1}[1] \subseteq M$. Hence a monoid morphism $e : X^* \to M$ recognizes $L$ via $f$ (i.e. $L = f \cdot e$) iff $L_0 = e^{-1}[F]$. This is the classical notion of recognition of a language $L_0 \subseteq X^*$ by a monoid, see e.g. Pin [18].

2. $\mathcal{D} = \text{Set}_\perp$ with $X = X_0 + \{\perp\}$, $X^* = X_0^* + \{\perp\}$ and $Y = \{\perp, 1\}$: given a monoid with zero $M$, a $\text{Set}_\perp$-morphism $f : M \to \{\perp, 1\}$ defines a subset $F = f^{-1}[1]$ of $M \setminus \{0\}$. A zero-preserving monoid morphism $e : X_0^* + \{\perp\} \to M$ recognizes $L$ via $f$ iff $L_0 = e^{-1}[F]$.

3. $\mathcal{D} = \text{Inv}$ with $X = X_0 + \widetilde{X}_0$, $X^* = X_0^* + \widetilde{X}_0^*$ and $Y = \{0, 1\}$: for an involution monoid $M$ to give a morphism $f : M \to \{0, 1\}$ means to give a subset $F = f^{-1}[1] \subseteq M$ satisfying $m \in F$ iff $\widetilde{m} \notin F$. Then $L$ is recognized by $e : X_0^* + \widetilde{X}_0^* \to M$ via $f$ iff $L_0 = X_0^* \cap e^{-1}[F]$.
4. \( \mathcal{D} = \text{JSL}_0 \) with \( X = \mathcal{P}_I X_0, X^* = \mathcal{P}_I X_0^* \) and \( Y = \{0,1\} \): for an idempotent semiring \( M \) a morphism \( f : M \to Y \) defines a prime upset \( F = f^{-1}[1] \), see Example 2.13. Hence \( L \) is recognized by a semiring homomorphism \( e : \mathcal{P}_I X_0^* \to M \) via \( i f \) iff \( L_0 = X_0^* \cap e^{-1}[F] \). Here we identify \( X_0^* \) with the set of all singleton languages \( \{w\}, w \in X_0^* \). This is the concept of language recognition introduced by Polák [19] (except that he puts \( F = f^{-1}[0] \), so 0 and 1 must be swapped, as well as \( F \) and \( M \setminus F \).

5. \( \mathcal{D} = \text{Mod}(\mathcal{S}) \) with \( X = \Psi X_0, X^* = \mathcal{S}[X_0] \) and \( Y = \mathcal{S} \): given an associative algebra \( M \), the language \( L \) is recognized by \( e : \mathcal{S}[X_0] \to M \) via \( f : M \to \mathcal{S} \) iff \( L = f \cdot e \). For the case where the semiring \( \mathcal{S} \) is a ring, this notion of recognition is due to Reutenauer [21].

**Remark 3.3.** 1. Since \( \mathcal{D} \) and \( \text{Mon}(\mathcal{D}) \) are varieties, we have the usual factorization system of regular epimorphisms (= surjective homomorphisms) and monomorphisms (= injective homomorphisms). *Quotients* and *subobjects* are understood w.r.t. this system.

2. By an \( X \text{-generated } \mathcal{D} \text{-monoid} \) we mean a quotient \( e : X^* \to M \) in \( \text{Mon}(\mathcal{D}) \). For two such quotients \( e_i : X^* \to M_i, i = 1,2, \) we say, as usual, that \( e_1 \) is smaller or equal to \( e_2 \) (notation: \( e_1 \leq e_2 \)) if \( e_1 \) factorizes through \( e_2 \). Note that if \( X = \Psi X_0 \), the free \( \mathcal{D} \text{-monoid} \) \( X^* = \Psi X_0^* \) on \( X \) is also the free \( \mathcal{D} \text{-monoid} \) on the set \( X_0 \) (w.r.t. the forgetful functor \( \text{Mon}(\mathcal{D}) \to \text{Set} \)), see Proposition 2.10. In this case, to give a quotient \( e : X^* \to M \) is equivalent to giving a set of generators for the \( \mathcal{D} \text{-monoid} \) induced by \( X_0 \) – which is why \( M \) may also be called an \( X_0 \text{-generated } \mathcal{D} \text{-monoid} \).

**Lemma 3.4.** The language recognized by an \( X \text{-generated } \mathcal{D} \text{-monoid} \( e : X^* \to M \) via \( f : M \to Y \) is the language accepted by its derived automaton.

We are now ready to give an abstract account of syntactic algebras in our setting. In classical algebraic automata theory the syntactic monoid of a language is characterized as the smallest monoid recognizing that language. We will use this property as our definition of the syntactic \( \mathcal{D} \text{-monoid} \).

**Definition 3.5.** The syntactic \( \mathcal{D} \text{-monoid} \) of language \( L : X^* \to Y \), denoted by \( \text{Syn}(L) \), is the smallest \( X \text{-generated } \mathcal{D} \text{-monoid} \) recognizing \( L \).

In more detail, the syntactic \( \mathcal{D} \text{-monoid} \) is an \( X \text{-generated } \mathcal{D} \text{-monoid} \( e_L : X^* \to \text{Syn}(L) \) together with a morphism \( f_L : \text{Syn}(L) \to Y \) of \( \mathcal{D} \) such that (i) \( e_L \) recognizes \( L \) via \( f_L \), and (ii) for every \( X \text{-generated } \mathcal{D} \text{-monoid} \( e : X^* \to M \) recognizing \( L \) via \( f : M \to Y \) we have \( e_L \leq e \), that is, the left-hand triangle below commutes for some \( \mathcal{D} \text{-monoid morphism} \) \( h \):

\[
\begin{array}{ccc}
X^* & \xrightarrow{e} & M \\
\downarrow{e_L} & & \downarrow{f_L} \\
\text{Syn}(L) & \xrightarrow{f} & Y
\end{array}
\]

Note that the right-hand triangle also commutes since \( e \) is epimorphic and \( f \cdot e = L = f_L \cdot e_L \). The universal property determines \( \text{Syn}(L), e_L, \) and \( f_L \) uniquely up to isomorphism. A construction of \( \text{Syn}(L) \) is given below (Construction 3.13). We first consider a special case:

**Example 3.6.** In \( \mathcal{D} = \text{Set} \) with \( Y = \{0,1\} \), the syntactic monoid of a language \( L \subseteq X^* \) can be constructed as the quotient of \( X^* \) modulo the syntactic congruence, see e.g. [18]:

\[
\text{Syn}(L) = X^*/\sim, \quad \text{where } u \sim v \text{ iff for all } x,y \in X^*: xuy \in L \iff xvy \in L.
\]
We aim to generalize this construction to our categorical setting. First note the following

> **Lemma 3.7.** Let $\mathcal{D}$ be any symmetric monoidal closed category with countable coproducts. Then the forgetful functor $\text{Mon}(\mathcal{D}) \to \mathcal{D}$ preserves reflexive coequalizers.

> **Notation 3.8.** Let $(M, m, i)$ be a $\mathcal{D}$-monoid and $x : I \to M$. We write $x \cdot -$ and $- \cdot x$ for the following morphisms, respectively:

\[
M \cong I \otimes M \xrightarrow{\mu \otimes M} M \otimes M \xrightarrow{m} M \quad \text{and} \quad M \cong M \otimes I \xrightarrow{M \otimes x} M \otimes M \xrightarrow{m} M.
\]

Recall that in our setting, where $\mathcal{D}$ is a commutative variety, we have $I = \Psi 1$ and so the morphism $x$ is the adjoint transpose of an element of $M$ (see Remark 2.6). In the following we shall often write $x \cdot y$, identifying $x, y : I \to M$ with their corresponding elements of $M$.

> **Definition 3.9.** The syntactic congruence of a language $L : X^* \to Y$ is the following relation on the underlying set of $X^*$:

\[
E = \{(u, v) \in X^* \times X^* \mid \forall x, y \in X^* : L(x \cdot u \cdot y) = L(x \cdot v \cdot y)\}
\]

The projection maps are denoted by $l, r : E \to X^*$.

> **Lemma 3.10.** The set $E$ carries a canonical $\mathcal{D}$-algebraic structure making it a $\mathcal{D}$-object.

**Proof sketch.** Just observe that $E = \bigcap E_{x, y}$ where for fixed $x, y \in X^*$ the object $E_{x, y}$ is the kernel of the $\mathcal{D}$-morphism $X^* \xrightarrow{x} X^* \xrightarrow{\mu} X^* \xrightarrow{L} Y$.  

That the name syntactic congruence makes sense follows from Lemma 3.11 below. First recall that a $\mathcal{D}$-monoid congruence on a given $\mathcal{D}$-monoid $M$ is an equivalence relation in $\text{Mon}(\mathcal{D})$, that is, a jointly monic pair $c_1, c_2 : C \to M$ of $\mathcal{D}$-monoid morphisms (equivalently a $\mathcal{D}$-submonoid $(c_1, c_2) : C \hookrightarrow M \times M$) which is reflexive, symmetric and transitive. Congruences on $M$ are ordered as subobjects of $M \times M$, i.e. via inclusion.

> **Lemma 3.11.** $E$ is a $\mathcal{D}$-monoid congruence on $X^*$.

We can give an alternative, more conceptual, description of $E$:

> **Lemma 3.12.** Let $l_0, r_0 : K \to X^*$ be the kernel pair of $L : X^* \to Y$ in $\mathcal{D}$. Then $l, r : E \to X^*$ is the largest $\mathcal{D}$-monoid congruence contained in $K$.

> **Construction 3.13.** Let $L : X^* \to Y$ be a language and $l, r : E \to X^*$ its syntactic congruence. We construct the $\mathcal{D}$-monoid $\text{Syn}(L)$ as the coequalizer of $l$ and $r$ in $\text{Mon}(\mathcal{D})$:

\[
E \xrightarrow{l} X^* \xrightarrow{e_L} \text{Syn}(L).
\]

We need to show that $\text{Syn}(L)$ has the universal property of Definition 3.5, which first requires the morphism $f_L : \text{Syn}(L) \to Y$ with $L = f_L \cdot e_L$. To this end consider the diagram below, where $l_0, r_0$ is the kernel pair of $L$ and $m$ witnesses that $E$ is contained in $K$, i.e. $l = l_0 \cdot m$ and $r = r_0 \cdot m$ (see Lemma 3.12).

\[
K \xrightarrow{l_0} X^* \xrightarrow{L} Y \quad \quad E \xrightarrow{r} X^* \xrightarrow{\mu} X^* \xrightarrow{L} Y \quad \quad \text{Syn}(L) \xrightarrow{f_L} Y
\]

By Lemma 3.7 the morphism $e_L$ is also a coequalizer of $l$ and $r$ in $\mathcal{D}$. Since $L \cdot l = L \cdot r$ by the above diagram, this yields a unique $f_L : \text{Syn}(L) \to Y$ with $L = f_L \cdot e_L$. In other words, $\text{Syn}(L)$ recognizes $L$ via $f_L$. 

**Diagram:**

```
  K  \xrightarrow{l_0} X^*  \xrightarrow{L} Y \\
    \downarrow m \quad \quad \downarrow f_L \\
    E  \xrightarrow{r_0} X^*  \xrightarrow{\mu} X^*  \xrightarrow{L} Y \\
        \quad \text{Syn}(L)
```
Theorem 3.14. Syn(L) together with ε_L and f_L forms the syntactic D-monoid of L.

Proof sketch. This follows from the correspondence between kernel pairs and regular quotients: since l, r : E → X* is the largest congruence contained in the kernel pair of L by Lemma 3.12, the coequalizer ε_L of l, r is the smallest quotient of X* recognizing L.

Remark 3.15. Our proof of Theorem 3.14 is quite conceptual and works in a general symmetric monoidal closed category D with enough structure. On this level of generality one would use Lemma 3.12 to define the syntactic congruence E as the largest D-monoid congruence contained in the kernel of L : X* → Y. However, it is unclear whether such a congruence exists in this generality and so its existence might have to be taken as an assumption. Hence we restricted ourselves to the setting of a commutative variety D.

Example 3.16. Using the notation of Example 2.23 we obtain the following concrete syntactic algebras:

1. In Set with X = X₀ + {⊥} and Y = {⊥, 1} the syntactic monoid with zero of a language L₀ ⊆ X₀* is (X₀* + {⊥})/∼ where, for all u, v ∈ X₀*:
   \[ u \sim v \text{ iff for all } x, y ∈ X₀^* : xuy ∈ L₀ \iff xvy ∈ L₀. \]
   The zero element is the congruence class of ⊥.

2. In Inv with X = X₀ + X₀* and Y = {0, 1} the syntactic involution monoid of a language L₀ ⊆ X₀* is the quotient of X₀ + X₀* modulo the congruence ∼ defined for words u, v ∈ X₀* as follows:
   \[ (i) \ u \sim v \text{ iff } \hat{u} \sim \hat{v} \text{ iff for all } x, y ∈ X₀^* : xuy ∈ L₀ \iff xvy ∈ L₀; \]
   \[ (ii) \ u \sim \hat{v} \text{ iff } \hat{u} \sim v \text{ iff for all } x, y ∈ X₀^* : xuy ∈ L₀ \iff xvy \notin L₀. \]

3. In Mod(S) with X = ΨX₀ and Y = S the syntactic associative S-algebra of a weighted language L₀ : X₀* → S is the quotient of S[X₀] modulo the congruence defined for U, V ∈ S[X₀] as follows:
   \[ U \sim V \text{ iff for all } x, y ∈ X₀^* : L(xUy) = L(xVy) \] (3)
   Indeed, since L : S[X₀] → S is linear, (3) implies L(PUQ) = L(PVQ) for all P, Q ∈ S[X₀], which is the syntactic congruence of Definition 3.9.

4. In particular, for D = JSI₀ with X = ΨX₀ and Y = {0, 1}, we get the syntactic (idempotent) semiring of a language L₀ ⊆ X₀* introduced by Polák [19]: it is the quotient \[ P \gamma X₀^*/∼ \] where for U, V ∈ P \gamma X₀* we have
   \[ U \sim V \text{ iff for all } x, y ∈ X₀^* : (xUy) \cap L₀ \neq \emptyset \iff xVy \cap L₀ \neq \emptyset. \]

5. For D = Vec(K) with X = ΨX₀ and Y = K, the syntactic K-algebra of a K-weighted language L₀ : X₀* → K is the quotient K[X₀]/I of the K-algebra of finite weighted languages modulo the ideal
   \[ I = \{ v ∈ K[X₀] \mid \text{ for all } x, y ∈ X₀^* : L(xVy) = 0 \}. \]
   Indeed, the congruence this ideal I generates (U \sim V iff U − V ∈ I) is precisely (3). Syntactic K-algebras were studied by Reutenauer [21].

6. Analogously, for D = Ab with X = ΨX₀ and Y = Z, the syntactic ring of a Z-weighted language L₀ : X₀* → Z is the quotient Z[X₀]/I, where I is the ideal of all V ∈ Z[X₀] with L(xVy) = 0 for all x, y ∈ X₀*.
4 Transition \( \mathcal{D} \)-Monoids

Here we present another construction of the syntactic \( \mathcal{D} \)-monoid of a language: it is the transition \( \mathcal{D} \)-monoid of the minimal \( \mathcal{D} \)-automaton for this language. Recall that for any object \( Q \) of a closed monoidal category \( \mathcal{D} \), the object \([Q, Q]\) forms a \( \mathcal{D} \)-monoid w.r.t. composition.

> **Definition 4.1.** The transition \( \mathcal{D} \)-monoid \( T(Q) \) of an \( F \)-algebra \((Q, \delta, i)\) is the image of the \( \mathcal{D} \)-monoid morphism \( (\lambda \delta)^+ : \mathcal{X}^* \to [Q, Q] \) extending \( \lambda \delta : X \to [Q, Q]\):

\[
\begin{array}{ccc}
\mathcal{X}^* & \xrightarrow{(\lambda \delta)^+} & [Q, Q] \\
\epsilon_{T(Q)} & & \downarrow m_{T(Q)} \\
T(Q) & \xrightarrow{\epsilon_{T(Q)}} & [Q, Q]
\end{array}
\]

> **Example 4.2.** 1. In \( \text{Set} \) the transition monoid of an \( F \)-algebra \( Q \) (i.e. an automaton without final states) is the monoid of all extended transition maps \( \delta_w = \delta_{a_n} \cdots \delta_{a_1} : Q \to Q \) for \( w = a_1 \cdots a_n \in \mathcal{X}^* \), with unit \( \text{id}_Q = \delta_e \) and composition as multiplication.

2. In \( \text{Set}_\bot \) with \( X = X_0 + \{ \bot \} \) (the setting for partial automata) this is completely analogous, except that we add the constant endomap of \( Q \) with value \( \bot \).

3. In \( \text{Inv} \) with \( X = X_0 + \tilde{X}_0 \) we get the involution monoid of all \( \delta_w \) and \( \tilde{\delta}_w \). Again the unit is \( \delta_e \), and the multiplication is determined by composition plus the equations \( x \tilde{y} = \tilde{y} \tilde{x} = \tilde{x} y \).

4. In \( \text{JSL}_0 \) with \( X = \mathcal{P}_f X_0 \) the transition semiring consists of all finite joins of extended transitions, i.e. all semilattice homomorphisms of the form \( \delta_{w_1} \cdots \delta_{w_n} \) for \( \{w_1, \ldots, w_n\} \in \mathcal{P}_f X_0^* \). The transition semiring was introduced by Polák [19].

5. In \( \text{Mod}(\mathcal{S}) \) with \( X = \Psi X_0 \) the associative transition algebra consists of all linear maps of the form \( \sum_{i=1}^n s_i \delta_{w_i} \) with \( s_i \in \mathcal{S} \) and \( w_i \in X_0^* \).

Recall from Definition 2.12 that a \( \mathcal{D} \)-automaton is an \( F \)-algebra \( Q \) together with an output morphism \( f : Q \to Y \). Hence we can speak of the transition \( \mathcal{D} \)-monoid of a \( \mathcal{D} \)-automaton.

> **Proposition 4.3.** The language accepted by a \( \mathcal{D} \)-automaton \((Q, \delta, f, i)\) is recognized by the \( \mathcal{D} \)-monoid morphism \( \epsilon_{T(Q)} : \mathcal{X}^* \to T(Q) \).

**Proof sketch.** The desired morphism \( f_{T(Q)} : T(Q) \to Y \) with \( L_Q = f_{T(Q)} \cdot \epsilon_{T(Q)} \) is

\[
f_{T(Q)} = (T(Q) \xrightarrow{m_{T(Q)}} [Q, Q]) \cong [Q, Q] \otimes \mathcal{I} \xrightarrow{[Q, Q] \otimes i} [Q, Q] \otimes Q \xrightarrow{\epsilon_Q} Q \xrightarrow{f} Y.
\]

> **Definition 4.4.** A \( \mathcal{D} \)-automaton \((Q, \delta, f, i)\) is called **minimal** if it is

(a) reachable: the unique \( F \)-algebra homomorphism \( \mathcal{X}^* \to Q \) is surjective;

(b) simple: the unique \( T \)-coalgebra homomorphism \( Q \to [X^*, Y] \) is injective.

> **Theorem 4.5** (Goguen [12]). Every language \( L : \mathcal{X}^* \to Y \) is accepted by a minimal \( \mathcal{D} \)-automaton \( \text{Min}(L) \), unique up to isomorphism. Given any reachable automaton \( Q \) accepting \( L \), there is a unique surjective automata homomorphism from \( Q \) into \( \text{Min}(L) \).

This leads to the announced construction of syntactic \( \mathcal{D} \)-monoids via transition \( \mathcal{D} \)-monoids. The case \( \mathcal{D} = \text{Set} \) is a standard result of algebraic automata theory (see e.g. Pin [18]), and the case \( \mathcal{D} = \text{JSL}_0 \) is due to Polák [19].

> **Theorem 4.6.** The syntactic \( \mathcal{D} \)-monoid of a language \( L : \mathcal{X}^* \to Y \) is isomorphic to the transition \( \mathcal{D} \)-monoid of its minimal \( \mathcal{D} \)-automaton:

\( \text{Syn}(L) \cong T(\text{Min}(L)) \).

**Proof sketch.** Using reachability and simplicity of \( \text{Min}(L) \), one proves that the quotients \( e_L : \mathcal{X}^* \to \text{Syn}(L) \) and \( \epsilon_{T(\text{Min}(L))} : \mathcal{X}^* \to T(\text{Min}(L)) \) have the same kernel pair, namely the syntactic congruence of \( L \). This implies the statement of the theorem.
Our results so far apply to arbitrary languages in $\mathcal{D}$. In the present section we focus on regular languages, which in $\mathcal{D} = \text{Set}$ are the languages accepted by finite automata, or equivalently the languages recognized by finite monoids. For arbitrary $\mathcal{D}$ the role of finite sets is taken over by finitely presentable objects. Recall that an object $D$ of $\mathcal{D}$ is finitely presentable if the hom-functor $\mathcal{D}(D, -) : \mathcal{D} \rightarrow \text{Set}$ preserves filtered colimits. Equivalently, $D$ is an algebra presentable with finitely many generators and relations.

**Definition 5.1.** A language $L : X \rightarrow Y$ is called $\mathcal{D}$-regular if it is accepted by some $\mathcal{D}$-automaton with a finitely presentable object of states.

To work with this definition, we need the following

**Assumptions 5.2.** We assume that the full subcategory $\mathcal{D}_f$ of finitely presentable objects of $\mathcal{D}$ is closed under subobjects, strong quotients and finite products.

**Example 5.3.**
1. Recall that a variety is locally finite if all finitely presentable algebras (equivalently all finitely generated free algebras) are finite. Every locally finite variety satisfies the above assumptions. This includes our examples $\text{Set}$, $\text{Set}_\mathbb{K}$, $\text{Inv}$ and $\text{JSL}_0$.
2. A semiring $S$ is called Noetherian if all submodules of finitely generated $S$-modules are finitely generated. In this case, as shown in [10], the category $\text{Mod}(S)$ satisfies our assumptions. Every field is Noetherian, as is every finitely generated commutative ring, so $\text{Vec}(K)$ and $\text{Ab} = \text{Mod}(\mathbb{Z})$ are special instances.

**Theorem 5.4.** For any language $L : X \rightarrow Y$ the following statements are equivalent:

(a) $L$ is $\mathcal{D}$-regular.
(b) The minimal $\mathcal{D}$-automaton $\text{Min}(L)$ has finitely presentable carrier.
(c) $L$ is recognized by some $\mathcal{D}$-monoid with finitely presentable carrier.
(d) The syntactic $\mathcal{D}$-monoid $\text{Syn}(L)$ has finitely presentable carrier.

**Proof sketch.** This follows immediately from the universal properties of $\text{Syn}(L)$ and $\text{Min}(L)$ and the assumed closure properties of $\mathcal{D}_f$.

Just as the collection of all languages is internalized by the final coalgebra $[X^*, Y]$, see Proposition 2.21, we can internalize the regular languages by means of the rational coalgebra.

**Definition 5.5.** The rational coalgebra $\rho T$ for $T$ is the colimit (taken in the category of $T$-coalgebras and homomorphisms) of all $T$-coalgebras with finitely presentable carrier.

**Proposition 5.6.** There is a one-to-one correspondence between $\mathcal{D}$-regular languages and elements $I \in \rho T$ of the rational coalgebra.

We conclude this section with an interesting dual perspective on syntactic monoids, based on our previous work [2, 4]. For lack of space we restrict to the case $\mathcal{D} = \text{Set}$. This category is predual to the category $\mathcal{BA}$ of boolean algebras in the sense that the full subcategories of finite sets and finite boolean algebras are dually equivalent. Indeed, this is a restriction of the well-known Stone duality: the dual equivalence functor assigns to a finite boolean algebra $B$ the set $\text{At}(B)$ of its atoms, and to a boolean homomorphism $h : A \rightarrow B$ the map $\text{At}(h) : \text{At}(B) \rightarrow \text{At}(A)$ sending $b \in \text{At}(B)$ to the unique atom $a \in \text{At}(A)$ with $h a \geq b$.

How do the concepts we investigated in $\text{Set}$ – languages, automata and monoids – dualize to $\mathcal{BA}$? Observe that $\text{Reg}(X)$, the boolean algebra of regular languages over the alphabet
Syntactic Monoids in a Category

$X$, can be viewed as a deterministic automaton: its final states are the regular languages containing the empty word, and the transitions are given by $L \xrightarrow{a} a^{-1}L$ for $a \in X$, where $a^{-1}L = \{ w \in X^* : aw \in L \}$ is the left derivative of $L$ w.r.t. the letter $a$. (Similarly, the right derivative of $L$ w.r.t. $a$ is $La^{-1} = \{ w \in X^* : wa \in L \}$.) This makes $\text{Reg}(X)$ a coalgebra for the endofunctor $\mathcal{T} = (\{0, 1\} \times \text{Id}_X)$ on $\text{BA}$. Since the two-chain $\{0, 1\}$ is dual to the singleton set 1, finite coalgebras for $\mathcal{T}$ dualize to finite algebras for the functor $F = 1 + X \times \text{Id} \cong 1 + \bigcup X \text{Id}$ on $\text{Set}$. Based on this, we proved in [2] that further (i) finite $\mathcal{T}$-subcoalgebras of $\text{Reg}(X)$ dualize to finite quotient algebras of the initial $F$-algebra $X^*$, and (ii) finite local varieties of languages (i.e. finite $\mathcal{T}$-subcoalgebras of $\text{Reg}(X)$ closed under right derivatives) dualize to those $F$-algebras associated to $X$-generated monoids, see Definition 2.15. For a regular language $L \subseteq X^*$ the $F$-algebras associated to the minimal automaton $\text{Min}(L)$ and the syntactic monoid $\text{Syn}(L)$ are finite. Their dual $\mathcal{T}$-coalgebras are characterized as follows:

> **Theorem 5.7.** Let $L \subseteq X^*$ be a regular language, and $L^{\text{rev}}$ its reversed language.

(a) $\text{Min}(L)$ is dual to the smallest subcoalgebra of $\text{Reg}(X)$ containing $L^{\text{rev}}$.

(b) $\text{Syn}(L)$ is dual to the smallest local variety of languages containing $L^{\text{rev}}$.

Part (a) of this theorem adds to the recently developed dual view of minimal automata, see [7] and also [17, 3]. All the above considerations generalize from $\text{BA}/\text{Set}$ to arbitrary pairs $\mathcal{C}/\mathcal{D}$ of preducal locally finite varieties of algebras. Examples include the self-preducal varieties $\mathcal{C} = \mathcal{D} = \text{JSL}_0$ and $\mathcal{C} = \mathcal{D} = \text{Vec}(\mathbb{K})$ for a finite field $\mathbb{K}$.

6 Conclusions and Future Work

We proposed the first steps of a categorical theory of algebraic language recognition. Despite our assumption that $\mathcal{D}$ is a commutative variety, the bulk of our definitions, constructions and proofs works in any symmetric monoidal closed category with enough structure. However, the construction of the syntactic monoid via the syntactic congruence, and the proof that it coincides with a transition monoid, required the concrete algebraic setting. It remains an open problem to develop a genuinely abstract framework for our theory. In particular, such a generalized setting should provide the means for incorporating ordered algebras, e.g. the syntactic ordered monoids of $\text{Pin}$ [18]. We expect this can be achieved by working with (order-)enriched categories, where the coequalizer in our construction of the syntactic monoid is replaced by a coinserter. A more general theory of recognition might also open the door to treating algebraic recognizers for additional types of behaviors, including Wilke algebras [23] (representing $\omega$-languages) and forest algebras [9] (representing tree and forest languages).

One of the leading themes of algebraic automata theory is the classification of languages in terms of their syntactic algebras. For instance, by Schützenberger’s theorem a language is star-free iff its syntactic monoid is aperiodic. We hope that our conceptual view of syntactic monoids (notably their dual characterization in Theorem 5.7) can contribute to a duality-based approach to such results, leading to generalizations and new proof techniques.

References


A Proofs

This Appendix contains all proofs and additional details we omitted due to space limitations.

Details for Example 2.1.4.

A semiring $S = (S, +, \cdot, 0, 1)$ consists of a commutative monoid $(S, +, 0)$ and a monoid $(S, \cdot, 1)$ such that

$$0x = x0 = 0, \quad x(y + z) = xy + xz \quad \text{and} \quad (x + y)z = xy + yz.$$

A module over a semiring $S$ is a commutative monoid $(M, \cdot, 0)$ together with a scalar multiplication $\cdot : S \times M \to M$ such that the following laws hold:

$$(r + s)a = ra + sa, \quad r(a + b) = ra + rb, \quad (rs)a = r(sa),$$

$$0a = 0, \quad 1a = 1, \quad r0 = 0.$$

Proof of Proposition 2.10

A constructive proof can be found in [2]. Here we give a more conceptual argument, using the universal property of the tensor product. Observe that the functor $\Psi : \mathbf{Set} \to \mathcal{D}$ is strongly monoidal, i.e., it preserves the unit and tensor product up to natural isomorphism. Indeed, we have $\Psi 1 = I$ by definition. To see that $\Psi(A \times B) \cong \Psi A \otimes \Psi B$ for all sets $A$ and $B$, consider the following bijections (natural in $\mathcal{D}$):

$$X \cong \prod_{n < \omega} X^\otimes n \cong \prod_{n < \omega} \Psi X^n_0 \cong \Psi(\prod_{n < \omega} X^n_0) = \Psi X^*_0.$$

This implies $\Psi(A \times B) \cong \Psi A \otimes \Psi B$ by the Yoneda lemma. Using the fact that $\Psi$ preserves coproducts, being a left adjoint, we conclude

$$X^* \cong \prod_{n < \omega} X^\otimes n \cong \prod_{n < \omega} \Psi X^n_0 \cong \Psi(\prod_{n < \omega} X^n_0) = \Psi X^*_0.$$

Alternatively one can show that the right adjoint $[-] : \mathcal{D} \to \mathbf{Set}$ is a monoidal functor. This implies that $\Psi$ preserves free monoids.

Details for Remark 2.18

First we recall the $\mathcal{D}$-monoid structure on $[Q, Q]$. Let $\iota_Q : Q \to I \otimes Q$ be the left unit isomorphism. Then the unit $j : I \to [Q, Q]$ and multiplication $m : [Q, Q] \otimes [Q, Q] \to [Q, Q]$ are the unique morphisms making the following diagram commutative, respectively:

$$\begin{array}{ccc}
[Q, Q] \otimes Q & \xrightarrow{\text{ev}} & Q \\
\downarrow_{j \otimes Q} & & \uparrow_{\text{ev}} \\
I \otimes Q & \xrightarrow{\iota_Q} & [Q, Q] \otimes [Q, Q] \otimes [Q, Q] \otimes [Q, Q] \otimes [Q, Q]
\end{array}$$
Now all we have to show is that the morphism in (2) is an $F$-algebra homomorphism. It then follows that it is the unique one $e_Q$. Indeed, the diagram below commutes:

\[
\begin{array}{c}
\xymatrix{
X \otimes X^* \ar[r]^-{\eta_X \otimes X^*} & X^* \otimes X^* \ar[r]^-{m_X} & X^* \ar[l]_-{\iota_X} & I \\
X \otimes X^* \otimes I \ar[r]^-{\eta_X \otimes X^* \otimes I} & X^* \otimes X^* \otimes I \ar[r]^-{m_X \otimes I} & X^* \otimes I \ar[l]_-{\iota_X \otimes I} & I \otimes I \\
X \otimes [Q, Q] \otimes Q \ar[r]^-{\lambda \otimes [Q, Q] \otimes Q} & [Q, Q] \otimes [Q, Q] \otimes Q \ar[r]^-{m \otimes Q} & [Q, Q] \otimes Q \ar[l]_-{\iota \otimes Q} & I \otimes Q \\
X \otimes Q \ar[r]^-{\lambda \otimes Q} & [Q, Q] \otimes Q \ar[r]^-{ev} & Q \ar[l]_-{\iota} & I \otimes Q \\
\end{array}
\]

**Proof of Proposition 2.21**

Given any coalgebra $(Q, \tau, f)$, consider the morphism $\delta = (X \otimes Q \xrightarrow{\cong} Q \otimes X \xrightarrow{\beta} Q)$ where $\beta$ is the uncurried form of $\tau : Q \to [X, Q]$, and denote by $\delta^* : X^* \otimes Q \to Q$ the extension of $\delta$ as in Remark 2.18. We claim that the unique coalgebra homomorphism into $[X^*, Y]$ is $\lambda h : Q \to [X^*, Y]$, where

\[
h = (Q \otimes X^* \cong X^* \otimes Q \xrightarrow{\delta^*} Q \xrightarrow{f} Y).
\]

Let us first prove that $h$ is indeed a coalgebra homomorphism. Preservation of outputs is shown by the following commutative diagram:

\[
\begin{array}{c}
\xymatrix{
Q \ar[r]^-{\cong} \ar[d]^-{\lambda h} & I \otimes Q \ar[r]^-{\iota_X \otimes Q} & X^* \otimes Q \ar[d]^-{h} & Q \ar[r]^-{f} & Y \\
Q \ar[r]^-{\cong} & Q \otimes I \ar[r]^-{\iota} & Q \otimes X^* & Q \ar[r]^-{ev} & [X^*, Y] \\
[X^*, Y] \ar[r]^-{\cong} & [X^*, Y] \otimes I \ar[r]^-{\lambda h \otimes i_X} & [X^*, Y] \otimes X^* \ar[r]^-{\lambda h \otimes X^*} & [X^*, Y] \otimes X^* \\
\end{array}
\]

For preservation of transitions it suffices to show that the following diagram commutes, where $\tau : [X^*, Y] \otimes X \to [X^*, Y]$ is the uncurried coalgebra structure of $[X^*, Y]$:

\[
\begin{array}{c}
\xymatrix{
Q \otimes X \ar[r]^-{\beta} & Q \\
[X^*, Y] \otimes X \ar[r]_-{\tau} & [X^*, Y] \\
& [X^*, Y] \ar[u]_-{\lambda h \otimes X} \\
\end{array}
\]
But the above diagram is precisely the curried version of the following one (where we omit writing $\otimes$ for space reasons):

The part $(\ast)$ follows easily from the definition of $\delta^*$, and the other parts are clear. Now suppose that any coalgebra homomorphism of $\lambda h : Q \rightarrow [X^*, Y]$ is given. We show that $h : Q \otimes X^* \rightarrow Y$ is determined by the composites $(Q \otimes X^* \otimes i_n) Q \otimes X^* \rightarrow Y$, $n < \omega$, where $i_n : X^* \rightarrow X$ is the $n$-th coproduct injection. This proves the uniqueness of $\lambda h$: since $\otimes$ preserves coproducts, the morphisms $(Q \otimes i_n)_{n<\omega}$ form a coproduct cocone. For $n = 0$, the claim is proved by the diagram:

And the following diagram shows that $h \cdot (Q \otimes i_{n+1})$ is determined by $h \cdot (Q \otimes i_n)$ (again we omit $\otimes$, in particular we write $X^n$ for $X^\otimes n$):
Note that part $(*)$ commutes by the definition of the coalgebra structure on $[X^*,Y]$ and all other parts are easy to see.

**Proof of Proposition 2.27**

Recall from the proof of Proposition 2.21 that the uncurried version of $c_Q$ is the morphism

$$Q \otimes X^* \cong X^* \otimes Q \xrightarrow{\phi} Q \xrightarrow{f} Y.$$ 

Hence $c_Q \cdot i : I \to [X^*,Y]$ determines the language

$$X^* \cong X^* \otimes I \xrightarrow{X^* \otimes \eta_X} X^* \otimes Q \xrightarrow{\phi} Q \xrightarrow{f} Y,$$

and this is precisely $L_Q$, as shown by the diagram below (where $r_X$ is the $F$-algebra structure of $X^*$):

\[
\begin{array}{ccc}
X^* & \xrightarrow{id} & X^* \otimes I \\
\downarrow & \ldots & \ldownarrow \\
X^* \otimes I & \xrightarrow{X^* \otimes \eta_X} & X^* \otimes X^* \\
\downarrow & \ldots & \downarrow \\
X^* \otimes Q & \xrightarrow{\phi} & Q \\
\downarrow & \ldots & \downarrow \\
Y & \xrightarrow{f} & Y
\end{array}
\]

This completes the proof.

**Proof of Lemma 3.4**

Let $e : X^* \to M$ be an $X$-generated $D$-monoid and let $f : M \to Y$. Then this recognizes the language $L = f \cdot e$. We are done once we prove that $e$ is the unique $F$-algebra morphism from $X^*$ to the $F$-algebra associated to $M$ and $e \cdot \eta_X$ (cf. Remark 3.3). Recall from Proposition 2.17 that the initial $F$-algebra is the $F$-algebra associated to the free $D$-monoid $X^*$ and $\eta_X$. Then the following diagram clearly commutes since $e$ is a $D$-monoid morphism:

\[
\begin{array}{ccc}
FX^* = I + X \otimes X^* & \xrightarrow{I + \eta_X \otimes X^*} & I + X^* \otimes X^* \\
\downarrow & \downarrow & \downarrow \\
FM = I + X \otimes M & \xrightarrow{I + (e \cdot \eta_X) \otimes M} & I + M \otimes M \\
\end{array}
\]

This completes the proof.

**Proof of Lemma 3.7**

First of all we know from Lack [14, Theorem 2] that the forgetful functor $U : \text{Mon}(D) \to D$ is monadic, and by Proposition 2.9 the monad is $Id^D = \coprod_{n \geq 0} Id^{D^n}$. It suffices to show that $Id^D$ preserves reflexive coequalizers: it then follows that $U$ preserves (in fact, creates) them. For that it is sufficient to prove that each $Id^{D^n}$ preserves reflexive coequalizers. This follows from (the proof of) [14, Lemma 1] using that in our setting both $X \otimes -$ and $- \otimes X$ preserve all colimits (being left-adjoints).
Proof of Lemma 3.10

Observe that \( E = \bigcap E_{x,y} \) where for fixed \( x, y \in X^* \), \( E_{x,y} \) is the kernel of the \( \mathcal{D} \)-morphism

\[
X^* \xrightarrow{x} X^* \xrightarrow{y} X^* \xrightarrow{L} Y.
\] (4)

Since all limits in \( \mathcal{D} \) (and in particular, kernels and intersections) are created by the forgetful functor \([-]\) : \( \mathcal{D} \rightarrow \text{Set} \) we see that \( E \) has a canonical structure of a \( \mathcal{D} \)-object as desired.

Proof of Lemma 3.11

It suffices to prove that pairs in \( E \) are closed under the monoid operation of \( X^* \). Clearly \( (\varepsilon, \varepsilon) \in E \), where \( \varepsilon : 1 \rightarrow X^* \) is the adjoint transpose of the unit \( I = \Psi 1 \rightarrow X^* \) of the free \( \mathcal{D} \)-monoid on \( X \). Given \( (u, v) \) and \( (u', v') \) in \( E \) we show that \( (u \cdot u', v \cdot v') \) is in \( E \), too. Indeed, we have for all \( x, y \in X^* \) that

\[
L(x \cdot u \cdot u' \cdot y) = L(x \cdot v \cdot u' \cdot y) = L(x \cdot v \cdot v' \cdot y).
\]

Proof of Lemma 3.12

To see that \( E \) from Definition 3.9 satisfies this property let \( l', r' : E' \rightarrow X^* \) be any \( \mathcal{D} \)-monoid congruence contained in \( K \) via \( m' : E' \rightarrow K \) with \( l_0 \cdot m' = l' \) and \( r_0 \cdot m' = r' \). Since \( l' \) is a \( \mathcal{D} \)-monoid morphism it is easy to see that for every \( x : I \rightarrow X^* \) the following square commutes (note that \( (x, x) \in E' \) since \( E' \) is reflexive):

\[
\begin{array}{ccc}
E' & \xrightarrow{(x,x) \cdot -} & E' \\
\downarrow l' & & \downarrow r' \\
X^* & \xrightarrow{- \cdot -} & X^*
\end{array}
\]

and similarly for \( r' \) in lieu of \( l' \) and/or \( - \cdot y \) in lieu of \( x \cdot - \). It follows that the morphism (4) in the proof of Lemma 3.10 merges \( l' \) and \( r' \):

\[
L \cdot (- \cdot y) \cdot (x \cdot -) \cdot l' = L \cdot l' \cdot (- \cdot (y, y) \cdot ((x, x) \cdot -)) = L \cdot l_0 \cdot m' \cdot (- \cdot (y, y) \cdot ((x, x) \cdot -)) = L \cdot r_0 \cdot m' \cdot (- \cdot (y, y) \cdot ((x, x) \cdot -)) = L \cdot r' \cdot (- \cdot (y, y) \cdot ((x, x) \cdot -)) = L \cdot (- \cdot y) \cdot (x \cdot -) \cdot r'.
\]

Hence \( E' \) is contained in any \( E_{x,y} \), and therefore it is contained in their intersection \( E \).

Proof of Theorem 3.14

Suppose that we have an \( X \)-generated \( \mathcal{D} \)-monoid \( e : X^* \rightarrow M \) and a morphism \( f : M \rightarrow Y \) recognizing \( L \), i.e. such that \( L = f \cdot e \). Let \( l_0, r_0 : K \rightarrow X^* \) be the kernel pair of \( L \) as in Lemma 3.12 and take the kernel pair \( l_M, r_M : K_M \rightarrow X^* \) of \( e \). Now clearly we have

\[
L \cdot l_M = f \cdot e \cdot l_M = f \cdot e \cdot r_M = L \cdot r_M.
\]

Hence, there is a unique \( n : K_M \rightarrow K \) such that \( l_0 \cdot n = l_M \) and \( r_0 \cdot n = r_M \). It follows that \( n \) is monomorphic and so \( K_M \) is a \( \mathcal{D} \)-monoid congruence contained in \( K \). Consequently
$K_M$ is contained in $E$ (the largest $D$-monoid congruence contained $K$ by Lemma 3.12) via $o : K_M \to E$ with $l_M = l \cdot o$ and $r_M = r \cdot o$. Then we obtain that
\[
e_L \cdot l_M = e_L \cdot l \cdot o = e_L \cdot r \cdot o = e_L \cdot r_M.
\]
Thus, using that $e : X^* \to M$ is the coequalizer of its kernel pair $l_M, r_M$ we obtain a unique $h : M \to \text{Sym}(L)$ with $e_L = h \cdot e$, as desired.

**Proof of Proposition 4.3**

Let $(Q, \delta, i, f)$ be a $D$-automaton. By definition it accepts the language $L_Q = (X^* \xrightarrow{e_Q} Q \xrightarrow{f} Y)$ where $e_Q$ is the unique $F$-algebra morphism. Consider the morphism that evaluates any endomorphism of $Q$ at the initial state:
\[
ev_i = ([Q, Q] \cong [Q, Q] \otimes I \xrightarrow{[Q, Q] \otimes \iota} [Q, Q] \otimes Q \xrightarrow{\epsilon} Q).
\]
Now let
\[
T(Q) = (T(Q) \xrightarrow{\mu_{T(Q)}} [Q, Q] \xrightarrow{\epsilon} Q \xrightarrow{f} Y).
\]
With this morphism $T(Q)$ recognizes $L$; indeed, using the canonical isomorphism $\epsilon_Z : Z \to Z \otimes I$ we compute:
\[
L_Q = f \cdot e_Q
= f \cdot \epsilon_{Z} \cdot ((\lambda \delta)^+ \otimes \iota_{X^*}) \quad \text{(see Remark 2.18)}
= f \cdot \epsilon_{Z} \cdot \iota_{[Q, Q]} \cdot ((\lambda \delta)^+ \otimes I) \cdot \iota_{X^*} \quad \text{(def. of $\epsilon_{Z}$)}
= f \cdot \epsilon_{Z} \cdot (\lambda \delta)^+ \quad \text{(naturality of $\iota$)}
= f \cdot \epsilon_{Z} \cdot m_{T(Q)} \cdot e_{T(Q)} \quad \text{(see Definition 4.1)}
= f_{T(Q)} \cdot e_{T(Q)} \quad \text{(def. of $f_{T(Q)}$)}
\]
This completes the proof.

**Proof of Theorem 4.6**

Let $\text{Min} (L) = (Q, \delta, i, f)$, and write $\delta_x : Q \to Q$ for $e_{T(Q)} (x)$ ($x \in X^*$). Note that $\delta_{x \cdot y} = \delta_y \cdot \delta_x$ for all $x, y \in X^*$ since $e_{T(Q)}$ is a $D$-monoid morphism. Observe also that the unique $F$-algebra homomorphism $e_Q : X^* \to Q$ assigns to $x \in X^*$ the element $\delta_x : I \to Q$, and the unique $T$-coalgebra homomorphism $m_Q : Q \to [X^*, Y]$ assigns to a state $q : I \to Q$ the language $x \mapsto f \cdot \delta_x \cdot q$. It suffices to show that the kernel of $e_{T(Q)}$ is the syntactic congruence of $L$, that is, for all $u, v \in X^*$ one has
\[
\delta_u = \delta_v \quad \text{iff} \quad \forall x, y \in X^* : L(x \cdot u \cdot y) = L(x \cdot v \cdot y).
\]
To see this, we reason as follows:
\[
\delta_u = \delta_v \quad \iff \quad \forall x : \delta_u \cdot e_Q (x) = \delta_v \cdot e_Q (x) \quad \text{(eq \ surjective)}
\]
\[
\iff \forall x : \delta_u \cdot \delta_x \cdot i = \delta_u \cdot \delta_x \cdot i \quad \text{(def. $e_Q$)}
\]
\[
\iff \forall x : m_Q \cdot \delta_u \cdot \delta_x \cdot i = m_Q \cdot \delta_u \cdot \delta_x \cdot i \quad \text{(mQ \ injective)}
\]
\[
\iff \forall x, y : f \cdot \delta_y \cdot \delta_u \cdot \delta_x \cdot i = f \cdot \delta_y \cdot \delta_u \cdot \delta_x \cdot i \quad \text{(def. $m_Q$)}
\]
\[
\iff \forall x, y : f \cdot \delta_{x \cdot y} \cdot i = f \cdot \delta_{x \cdot y} \cdot i \quad \text{(def. $\delta(\cdot)$)}
\]
\[
\iff \forall x, y : f \cdot e_Q (x \cdot u \cdot y) = f \cdot e_Q (x \cdot v \cdot y) \quad \text{(def. $e_Q$)}
\]
\[
\iff \forall x, y : L(x \cdot u \cdot y) = L(x \cdot v \cdot y) \quad \text{(L = L_Q)}
\]
Proof of Theorem 5.4

Remark.

1. The functor $F Q = I + X \otimes Q$ preserves strong epimorphisms because the left adjoint $X \otimes -$ does and strong epimorphisms are closed under coproducts. Therefore every $F$-algebra homomorphism factorizes into a surjective homomorphism (carried by a strong epimorphism in $\mathcal{D}$) and an injective one (carried by a monomorphism in $\mathcal{D}$). By the reachable part $Q_r$ of an automaton $(Q, \delta, f, i)$ we mean the image of the initial $F$-algebra homomorphism, i.e., $e_Q = (X^* \xrightarrow{e_T} Q_r \xrightarrow{m_r} Q)$. Putting $f_r := f : m_r : Q_r \to Y$, the $F$-algebra $Q_r$ becomes an automaton, and $m_r$ an automata homomorphism. 

2. In the following automata, (co-)algebras and monoids with finitely presentable carrier are referred to as $fp$-automata, $fp$-(co-)algebras and $fp$-monoids, respectively.

Now for the proof of the theorem. (a)$\Leftrightarrow$(b) follows from Theorem 4.5 and the closure of $\mathcal{D}_f$ under subobjects and strong quotients. Similarly, (c)$\Leftrightarrow$(d) follows from the universal property of the syntactic monoid (see Definition 3.5) and again closure of $\mathcal{D}_f$ under subobjects and strong quotients. (c)$\Rightarrow$(a) is a consequence of Lemma 3.4. To prove (a)$\Rightarrow$(c), let $Q$ be any $fp$-automaton accepting $L$. Then by Proposition 4.3 the transition monoid $T(Q) = [Q, Q]$ recognizes $L$, so by closure of $\mathcal{D}_f$ under subobjects it suffices to show that $[Q, Q]$ is a finitely presentable object of $\mathcal{D}$. Assuming that $Q$ has $n$ generators as an algebra of $\mathcal{D}$, the map $[Q, Q] \to Q^n$ defined by restriction to the set of generators is an injective $\mathcal{D}$-morphism. Since $\mathcal{D}_f$ is closed under subobjects and finite products, it follows that $[Q, Q]$ is finitely presentable.

Proof of Proposition 5.6

We describe mutually inverse maps

$$(I \xrightarrow{x} gT) \mapsto (X^* \xrightarrow{L} Y) \text{ and } (X^* \xrightarrow{L} Y) \mapsto (I \xrightarrow{x} gT)$$

between the elements of $gT$ and the $\mathcal{D}$-regular languages. Let $h_Q : Q \to gT$ be the injections of the colimit $gT$, where $Q = (Q, \delta_Q, f_Q)$ ranges over all fp-coalgebras. Note that this colimit is filtered since $\mathcal{D}_f$ is closed under finite colimits. Moreover, since colimits of coalgebras are formed in the underlying category, the morphisms $h_Q$ also form a colimit cocone in $\mathcal{D}$. Given an element $I \xrightarrow{x} gT$ of the rational coalgebra we define a $\mathcal{D}$-regular language $L_x : X^* \to Y$ as follows: since $I = \Psi 1$ is finitely presentable, there exists an fp-coalgebra $Q$ and a morphism $i_Q : I \to Q$ such that $x = h_Q \cdot i_Q$. For the $F$-algebra $(Q, \delta_Q, i_Q)$ we have the unique $F$-algebra homomorphism $e_Q : X^* \to Q$, and we put $L_x := f_Q \cdot e_Q$.

We need to show that $L_x$ is well-defined, that is, for any other factorization $x = h_{Q'} \cdot i_{Q'}$ we have $f_Q \cdot e_Q = f_{Q'} \cdot e_{Q'}$. Given such a factorization, since the $h_Q$ form a filtered colimit, there exists an fp-coalgebra $Q'' = (Q'', \delta_{Q''}, f_{Q''})$ and coalgebra homomorphisms $h_{Q''} : Q \to Q''$ and $h_{Q''} \cdot i_{Q''} = h_{Q''} \cdot i_{Q'} = i_{Q''}$. Then for the $F$-algebra $(Q'', \delta_{Q''}, i_{Q''})$ we have the unique homomorphism $e_{Q''} : X^* \to Q''$. Moreover, $h_{Q''} \cdot e_{Q''}$ are also homomorphisms of $F$-algebras. If follows that

$$f_Q \cdot e_Q = f_{Q''} \cdot h_{Q''} \cdot e_Q \quad \text{(h}_{Q''} \text{ coalgebra homomorphism)}$$
$$= f_{Q''} \cdot e_{Q''} \quad \text{(h}_{Q''} \text{ F-algebra homom., X* initial)}$$

and analogously $f_{Q'} \cdot e_{Q'} = f_{Q''} \cdot e_{Q''}$. Hence $f_Q \cdot e_Q = f_{Q'} \cdot e_{Q'}$, as claimed.
Conversely, let \( L : \mathcal{X}^* \to Y \) be a \( \mathcal{D} \)-regular language. Then there exists an fp-automaton \( (Q, \delta_Q, i_Q, f_Q) \) with \( L = f_Q \cdot i_Q \). Put \( x_L := h_Q \cdot i_Q : I \to \gamma^T \). To prove the well-definedness of \( x_L \), consider the automata homomorphisms

\[
Q \xrightarrow{m} Q_r \xrightarrow{\epsilon} \text{Min}(L)
\]

of Theorem 4.5. Then

\[
\begin{align*}
 h_Q \cdot i_Q &= h_Q \cdot m \cdot i_Q, \\
 &= h_Q_r \cdot i_Q_r, \\
 &= h_{\text{Min}(L)} \cdot e \cdot i_Q_r, \\
 &= h_{\text{Min}(L)} \cdot i_{\text{Min}(L)}.
\end{align*}
\]

Hence \( x_L = h_Q \cdot i_Q \) is independent of the choice of \( Q \). It now follows immediately from the definitions that \( x \mapsto L_x \) and \( L \mapsto x_L \) are mutually inverse and hence define the desired bijective correspondence.

**B Dual Characterization of Syntactic Monoids**

Here we give a more detailed account of the dual view of syntactic monoids indicated in Section 5. This section are largely based on results from our papers \([2, 4]\) where a categorical generalization of Eilenberg’s variety theorem was proved. We work with the following

- **Assumptions B.1.** From now on \( \mathcal{D} \) is a locally finite entropic variety whose epimorphisms are surjective. Moreover, we assume that there is another locally finite variety \( \mathcal{C} \) such that the full subcategories \( \mathcal{C}_f \) and \( \mathcal{D}_f \) of finite algebras are dually equivalent. (Two such varieties \( \mathcal{C} \) and \( \mathcal{D} \) are called *predual*.)

The action of the equivalence functor \( \mathcal{C}_{op}^f \Rightarrow \mathcal{D}_f \) on objects and morphisms is written \( Q \mapsto \hat{Q} \) and \( f \mapsto \hat{f} \). Letting \( T \in \mathcal{C}_f \) denote the free one-generated object of \( \mathcal{C} \) we choose the output object \( Y \in \mathcal{D}_f \) to be the dual object of \( T \). Moreover, let \( \overline{Y} \in \mathcal{C}_f \) be the dual object of \( I \in \mathcal{D}_f \), the free one-generated object of \( \mathcal{D} \). Finally, we put \( X = \Psi X_0 \) for a finite alphabet \( X_0 \). Note that the underlying sets of \( \overline{Y} \) and \( Y \) are isomorphic:

\[
|\overline{Y}| \cong \mathcal{C}(T, \overline{Y}) \cong \mathcal{D}(I, Y) \cong |Y|.
\]

To simplify the presentation, we will assume in the following that \( \overline{Y} \) and \( Y \) have a two-element underlying set \( \{0, 1\} \). This is, however, inessential – see Remark B.14 at the end of this section.

- **Example B.2.** The categories \( \mathcal{C} \) and \( \mathcal{D} \) in the table below satisfy our assumptions.

<table>
<thead>
<tr>
<th>( \mathcal{C} )</th>
<th>( \mathcal{D} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>BA</td>
<td>Set</td>
</tr>
<tr>
<td>BR</td>
<td>Set (_\perp)</td>
</tr>
<tr>
<td>JSL(_0)</td>
<td>JSL(_0)</td>
</tr>
<tr>
<td>Vec((\mathbb{Z}_2))</td>
<td>Vec((\mathbb{Z}_2))</td>
</tr>
</tbody>
</table>

The case \( \text{BA/Set} \) is a restriction of Stone duality: the dual equivalence functor \( \text{BA}_{op}^f \Rightarrow \text{Set}_f \) assigns to a finite boolean algebra \( B \) the set \( \text{At}(B) \) of its atoms, and to a homomorphism \( h : A \to B \) the map \( \text{At}(h) : \text{At}(B) \to \text{At}(A) \) sending \( b \in \text{At}(B) \) to the unique atom \( a \in \text{At}(A) \).
with \( ha \geq b \). Using a similar Stone-type duality, we proved in [4] that the category \( \text{BR} \) of non-unital boolean rings (i.e., rings without 1 satisfying the equation \( x \cdot x = x \)) is pre dual to \( \text{Set}_L \). The other two examples above correspond to the well-known self-duality of finite semilattices and finite-dimensional vector spaces, respectively. We refer to [4] for details.

On \( \mathcal{C} \) we consider the endofunctor \( TQ = Y \times X_0^{X_0} \). Its coalgebras are precisely deterministic automata in \( \mathcal{C} \) without an initial state, represented as triples \( (Q, \gamma_a, f) \) with transition morphisms \( \gamma_a : Q \to Q \ (a \in X_0) \) and an output morphism \( f : Q \to \bar{Y} \).

**Example B.3.** In \( \mathcal{C} = \text{BA} \) a \( T \)-coalgebra is a deterministic automaton with a boolean algebra \( Q \) of states, boolean transitions morphisms \( \gamma_a \), and an output map \( f : Q \to \{0, 1\} \) which specifies (via the preimage of 1) an ultrafilter \( F \subseteq Q \) of final states.

The rational coalgebra \( gT \) of \( T \) (i.e., the colimit of all finite \( T \)-coalgebras) has as states the regular languages over \( X_0 \). The final state predicate \( f : gT \to \bar{Y} = \{0, 1\} \) sends a language to 1 if it contains the empty word \( \epsilon \), and the transitions \( \gamma_a : gT \to gT \) for \( a \in X_0 \) are given by \( \gamma_a(L) = a^{-1}L \). Here \( a^{-1}L = \{w \in X_0^*: aw \in L\} \) denotes the left derivative of \( L \). Similarly, the right derivatives of \( L \) are defined by \( La^{-1} = \{w \in X_0^*: wa \in L\} \) for \( a \in X_0 \).

**Example B.4.** In \( \mathcal{C} = \text{BA} \) the rational \( T \)-coalgebra is the boolean algebra of all regular languages over \( X_0 \) (w.r.t. union, intersection and complement), equipped with the above transitions and final states. Note that the transition map \( a^{-1}(-) \) is indeed a boolean homomorphism because left derivatives preserve all boolean operations. Moreover, the final states — viz. the set of all regular languages containing the empty word — form a (principal) ultrafilter.

The coalgebra \( gT \) is characterized by a universal property: every finite \( T \)-coalgebra has a unique coalgebra homomorphism into it (which sends a state to the language it accepts in the classical sense of automata theory). A finite \( T \)-coalgebra is called a subcoalgebra of \( gT \) if this unique morphism is injective, i.e., distinct states accept distinct languages. In [2] we related finite \( T \)-coalgebras in \( \mathcal{C} \) to finite \( F \)-algebras in the predual category \( \mathcal{D} \). Note that \( X = \Psi X_0 \) implies \( FA = I + X \otimes A \cong I + \coprod_{a \in A} A \), so \( F \)-algebras \( F \)-algebras \( (A, \delta) \) can be represented as triples \( (A, \delta_a, i) \) with \( \delta_a : A \to A \ (a \in X_0) \) and \( i : I \to A \). They correspond to automata in \( \mathcal{D} \) with inputs from the alphabet \( X_0 \) and without final states.

**Proposition B.5** (see [2]). (a) The categories of finite \( T \)-coalgebras and finite \( F \)-algebras are dually equivalent. The equivalence maps any finite \( T \)-coalgebra \( Q = (Q, \gamma_a, f) \) to its dual \( F \)-algebra \( \hat{Q} = (\hat{Q}, \hat{\gamma_a}, \hat{f}) \):

\[
(\mathcal{Y} \xleftarrow{L} Q \xrightarrow{\gamma_a} Q) \mapsto (I \xrightarrow{I} \hat{Q} \xleftarrow{\hat{\gamma}_a} \hat{Q}).
\]

(b) A finite \( T \)-coalgebra \( Q \) is a subcoalgebra of \( gT \) if its dual \( F \)-algebra \( \hat{Q} \) is a quotient of the initial \( F \)-algebra \( X^* \).

**Example B.6.** For a finite \( T \)-coalgebra \( Q = (Q, \gamma_a) \) in \( \text{BA} \) the dual \( F \)-algebra \( \hat{Q} \) has as states the atoms of \( Q \), and the initial state is the unique atomic final state of \( Q \). Moreover, there is a transition \( z \xrightarrow{a} z' \) if \( a \in X_0 \) in \( \hat{Q} \) iff \( z' \) is the unique atom with \( \gamma_a(z') \geq z \) in \( Q \).

By a local variety of languages over \( X_0 \) in \( \mathcal{C} \) we mean a subcoalgebra \( V \) of \( gT \) closed under right derivatives (i.e., \( L \in |V| \) implies \( La^{-1} \in |V| \) for all \( a \in X_0 \)). Note that a local variety is also closed under the \( \mathcal{C} \)-algebraic operations of \( gT \), being a subalgebra of \( gT \) in \( \mathcal{C} \), and under left derivatives, being a subcoalgebra of \( gT \).
Example B.7. A local variety of languages in $\mathbf{BA}$ is a set of regular languages over $X_0$ closed under the boolean operations (union, intersection and complement) as well as left and right derivatives. This concept was introduced by Gehrke, Grigorieff and Pin [11].

In the following proposition recall that every $X_0$-generated $\mathcal{D}$-monoid defines an $F$-algebra, see Definition 2.15.

Proposition B.8 (see [2]). A finite subcoalgebra $V$ of $g\mathcal{T}$ is a local variety iff its dual $F$-algebra $\hat{V}$ is derived from an $X_0$-generated $\mathcal{D}$-monoid.

In other words, given a finite local variety $V : \to g\mathcal{T}$ in $\mathcal{C}$, there exists a unique $\mathcal{D}$-monoid structure on $\hat{V}$ making the unique (surjective) $F$-algebra to $\mathcal{D}$-algebra homomorphism $e_V : X^* \to \hat{V}$ is $\mathcal{D}$-monoid morphism. Hence the monoid multiplication on $\hat{V}$ is well-defined by $e_V(x) \cdot e_V(y) := e_V(x \cdot y)$ for all $x, y \in X^*$, and the unit it the initial state of the $F$-algebra $\hat{V}$.

Remark B.9. A pointed $\mathcal{T}$-coalgebra is a $\mathcal{T}$-coalgebra $(Q, \gamma_a, f)$ equipped with an initial state $i : 1 \to Q$. Observe that every finite pointed $\mathcal{T}$-coalgebra $(Q, \gamma_a, f, i)$ dualizes to a finite $\mathcal{D}$-automaton $(\hat{Q}, \hat{\gamma}_a, \hat{f}, \hat{i})$. The language of $(Q, \delta_a, f, i)$ is the function

$$L_Q : X^*_0 \to |Y|, \quad a_1 \ldots a_n \mapsto f \cdot \delta_{a_1} \cdot \ldots \cdot \delta_{a_n} \cdot i.$$ 

Letting $m_Q : Q \to g\mathcal{T}$ denote the unique coalgebra homomorphism, $L_Q$ is precisely the element of $g\mathcal{T}$ determined by $I \xrightarrow{\cdot} Q \xrightarrow{m_Q} g\mathcal{T}$. Since $|Y| = |Y|$ and $X^* = \Psi X^*_0$, the function $L_Q : X^*_0 \to |Y|$ can be identified with its adjoint transpose $L_Q^T : X^* \to Y$, i.e., with a language in $\mathcal{D}$. The reversal of a language $L : X^* \to Y$ in $\mathcal{D}$ is $L^{\text{rev}} = L \cdot L^{\text{rev}} : X^* \to Y$, where $L^{\text{rev}} : X^* \to X^*$ denotes the unique morphism of $\mathcal{D}$ extending the function $X^*_0 \to X^*_0$ that reverses words.

Proposition B.10 (see [4]). The language accepted by a finite pointed $\mathcal{T}$-coalgebra is the reversal of the language accepted by its dual $\mathcal{D}$-automaton.

If a finite $X_0$-generated $\mathcal{D}$-monoid $e : X^* \to M$ recognizes a language $L : X^* \to Y$ via $f : M \to Y$, i.e., $L = (X^* \xrightarrow{e} M \xrightarrow{f} Y)$, we dually get the morphism $\mathcal{T} \xrightarrow{m} V \xrightarrow{g\mathcal{T}} g\mathcal{T}$ (where $V$ is the local variety dual to $M$, $i$ is the dual morphism of $f$ and $m$ is the unique coalgebra homomorphism) choosing the element $L^{\text{rev}}$ of $g\mathcal{T}$. Now suppose that $L$ is a regular language, and let $V_L$ be the finite local variety of languages dual to the syntactic $\mathcal{D}$ monoid $\text{Syn}(L)$, see Proposition B.8. The universal property of $\text{Syn}(L)$ in Definition 3.5 then dualizes as follows: $V_L$ is (i) a local variety containing $L^{\text{rev}}$, and (ii) for every local variety $V \to g\mathcal{T}$ containing $L^{\text{rev}}$, the local variety $V_L$ is contained in $V$. In other words, $V_L$ is the smallest local variety containing $L^{\text{rev}}$.

In summary, we have followed dual characterization of syntactic $\mathcal{D}$-monoids:

Theorem B.11. For every regular language $L$ the syntactic $\mathcal{D}$-monoid $\text{Syn}(L)$ is dual to the smallest local variety of languages over $X_0$ in $\mathcal{C}$ containing $L^{\text{rev}}$.

Example B.12. For $\mathcal{C} = \mathbf{BA}$ and $\mathcal{D} = \mathbf{Set}$ the previous theorem gives the following construction of the syntactic monoid of a regular language $L \subseteq X^*$:
1. Form the smallest local variety of languages $V_L \subseteq \varrho T$ containing $L^{rev}$. Hence $V_L$ is the closure of the (finite) set of all both-sided derivatives $u^{-1}L^{rev}v^{-1} = \{w \in X^* : uwv \in L^{rev}\}$ $(u, v \in X^*)$ under union, intersection and complement.

2. Compute the $F$-algebra $\hat{V}_L$ dual to the coalgebra $V_L$. The states of $\hat{V}_L$ are the atoms of $V_L$, and the initial state is the unique atom $i \in V_L$ containing the empty word. Given atoms $z, z' \in V_L$ and $a \in X$, there is a transition $z \xrightarrow{a} z'$ in $\hat{V}_L$ iff $z'$ is the (unique) atom with $z \subseteq a^{-1}z'$.

3. Define a monoid multiplication on $\hat{V}_L$ as follows: given states $z, z' \in \hat{V}_L$, choose words $w, w' \in X^*$ with $i \xrightarrow{w} z$ and $i \xrightarrow{w'} z'$ in $\hat{V}_L$. Then $z \bullet z'$ is the state reached on input $ww'$, i.e., $i \xrightarrow{ww'} z \bullet z'$. The resulting monoid (with multiplication $\bullet$ and unit $i$) is $\text{Syn}(L)$.

By dropping right derivatives and using the correspondence between finite subcoalgebras of $\varrho T$ and finite quotient algebras of $X^*$, one also gets the following dual characterization of minimal $D$-automata:

> **Theorem B.13.** For every regular language $L$ the minimal $D$-automaton for $L$ is dual to the smallest subcoalgebra of $\varrho T$ containing $L^{rev}$

> **Remark B.14.** Our above assumption that $Y$ and $\mathcal{Y}$ have two elements is inessential. Without this assumption, the rational coalgebra $\varrho T$ is not carried by regular languages, but more generally by regular behaviors, i.e., functions $b : X_0^* \to |Y|$ realized by finite Moore automata with output set $|Y|$. The coalgebra structure is given by the output map $b \mapsto b(\varepsilon)$, and transitions $b \xrightarrow{a} a^{-1}b$ for $a \in X_0$, where $a^{-1}b$ is the (generalized) left derivative of $b$ defined by $a^{-1}b(w) = b(aw)$. (Right derivatives are defined analogously.) A local variety of behaviors over $X_0$ in $\mathcal{C}$ is a subcoalgebra of $\varrho T$ closed under right derivatives. All results of this section hold for this more general setting, see Section 5 of [4] for details. In particular, this allows us to cover the case $\mathcal{C} = D = \text{Vec}(K)$ for an arbitrary finite field $K$. In this case Theorem 5.7 states that the syntactic associative algebra of a rational weighted language $L : X_0^* \to K$ dualizes to the smallest set of rational weighted languages that contains $L^{rev}$ and is closed under scalar multiplication, addition and left and right derivatives.